

User Manual

This is the official manual for wfview.

If you are just starting, please read the [Getting Started chapter](#) first. Most settings should be very intuitive, however, detailed explanations of each button and control are included in the links below.

For support, please visit <https://forum.wfview.org/> and sign up

Getting Started:

- [First time guide](#)
- [FAQ](#)
- [Xiegu X6100](#) (offsite link)

User Interface:

- [View Tab](#)
 - [CW Sender](#)
 - [Repeater and Split](#)
- [Band Tab](#)
- [Frequency Tab](#)
- [Settings Tab](#)
- [USB Controllers \(Icom RC-28, Contour Shuttle Express, and Contour ShuttlePRO\)](#)
- [Keyboard Shortcuts](#)
- [Appearance Customization with Examples](#)

Interaction with Other Programs:

- [Sharing control with other programs](#)
 - [Hamlib rigctld emulation](#) (recommended)
 - [Virtual Serial Port](#)
- [Audio Configuration for Windows/Linux \(audio sharing\)](#)

Built-in Server:

- [Remote Operation \(server\) with common hardware](#)
 - [Inexpensive Dedicated Server using SBCs like the Raspberry Pi and Inovato Quadra](#)

How-To:

- [Send us your logfile](#)
- [APRS packet decoding in linux over wfview's network audio](#)
- [Serial Port Management \(Linux udev rules, Device Manager, etc\)](#)
- [Inexpensive Dedicated Server using SBCs like the Raspberry Pi and Inovato Quadra](#)

Advanced:

- [Command-Line Options](#)
- [Logfile](#)
- [Preferences File](#)
- [Using older \(unsupported\) radios](#)
- [Supported Radios and Features](#)
- [Headless Server \(beta\)](#)
- [Compiling \(see the developer's corner\)](#)

Older versions of the User Manual may be found [here](#).

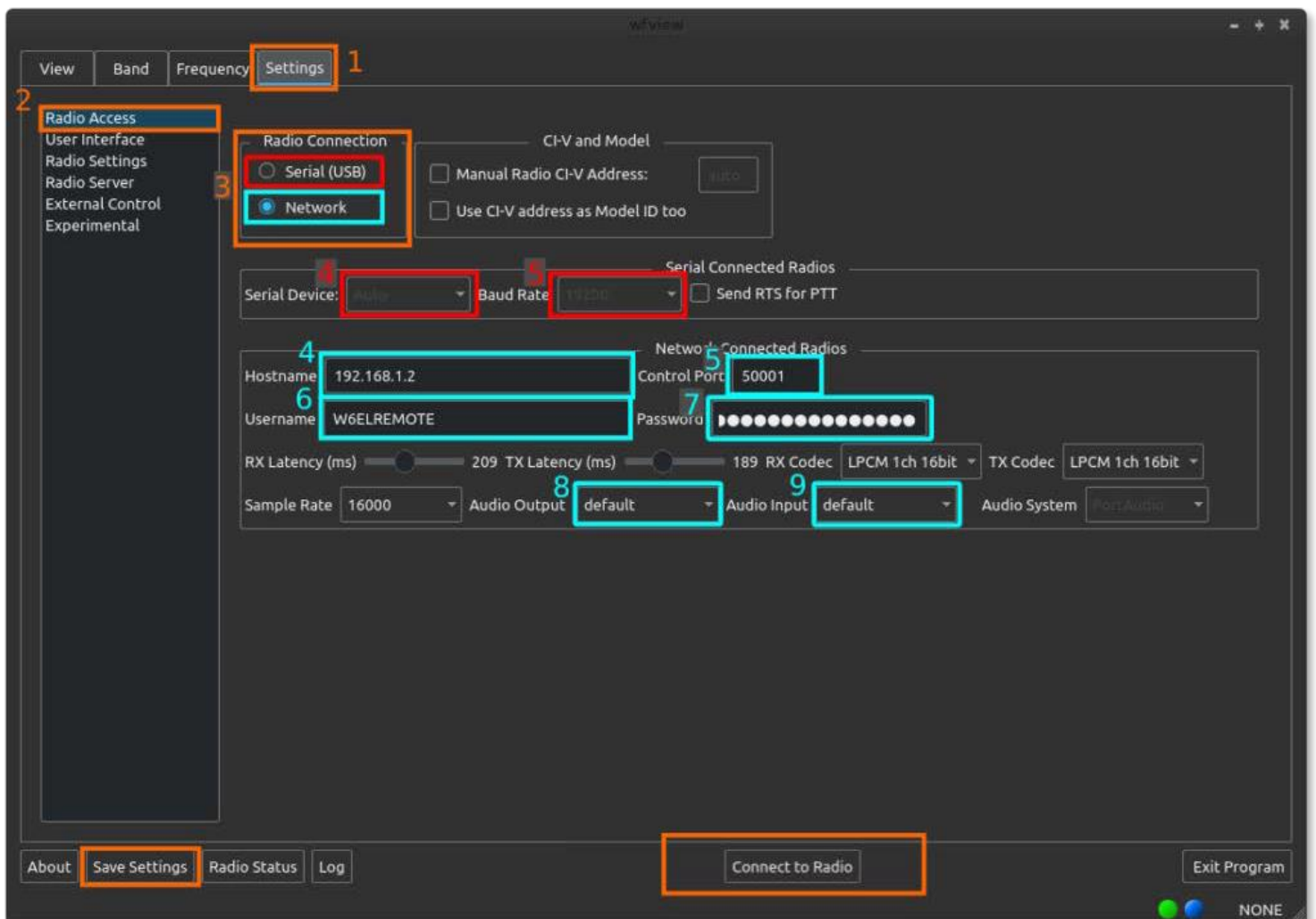
Getting Started

Once wfview has been downloaded, you will need to set up the radio and the program.

Radio Settings (via the Radio's Menu):

- Set the radio to the default CI-V address.
- address. Enable CI-V Transceive.
- For radios with ethernet or wifi: Set a known username and password, and enable Network Control. Make a note of the radio's ip address.
- For radios with Serial or USB: Set the baud rate to the highest supported rate, and make a note of the rate. [Here](#) is an in-depth guide to [IC-7300 radio settings](#).

wfview settings:



First:

Press [1] "Settings" and then press [2] "Radio Access". Select [3] either Serial (USB) or Network.

For Serial (USB) radios:

- Select the [4] Serial Device
- Select the [5] Baud Rate
- **Do not** check "Manual Radio CI-V Address"
- **Do not** check "Use CI-V address as Model ID too"

For Network (ethernet or wifi) radios:

- Type in the [4] IP address of the radio or radio server. If you know the hostname, you may supply the hostname instead.
- Enter the [5] Control Port. The default is 50001 on most radios.
- Enter the [6] username as set on the radio.
- Enter the [7] password
- Select the [8] audio output device you wish to use. Typically this is your headset or speakers. Linux users may simply select "default" for their desktop environment's default audio choice.
- Select the [9] audio input device you wish to use. Typically this is your microphone. Linux users may simply select "default" for their desktop environment's default audio choice.
- **Do not** check "Manual Radio CI-V Address"
- **Do not** check "Use CI-V address as Model ID too"

Next, for both types:

Press "Connect to Radio" (bottom of the window). If successful, the radio model number should appear in the window title bar and the lower-right corner of the window. Once working, **press "Save Settings"** (bottom left of the window).

FAQ

This is a collection of questions that we frequently see come up on the support forum. Please, have a look here before posting, it may save you a lot of time.

Section 1: Downloading, Installing, and Initial Run:

Q: I'm using Windows and I get an error that "MSVCP140.dll is not found.

A: This happens sometimes. Press "Ignore" if prompted. If there are errors running wfview, simply run the `vcredist_x86.exe` program, which is conveniently located in the wfview application directory (usually `c:\program files (x86)\wfview\`)

Q: I'm using Windows and I get an error that "VCRUNTIME140.DLL" is not found.

A: This happens sometimes. Press "Ignore" if prompted, and then simply run the `vcredist_x86.exe` program, which is conveniently located in the wfview application directory (usually `c:\program files (x86)\wfview\`)

Q: I'm using Windows and I get an error that "vcredist_x86.exe" failed to start or could not be found.

A: This happens sometimes. Press "Ignore" and then, if wfview will not start, simply run the `vcredist_x86.exe` program, which is conveniently located in the wfview application directory (usually `c:\program files (x86)\wfview\`)

Q: I'm using windows and I get an error that "Qt5Xml.dll" is not found.

A: This can happen if wfview isn't in the same folder as that DLL file, and may especially occur since that is a new dependency found only in our daily and weekly builds. The DLL file can be found in our public builds folder. Simply place it in the same folder as wfview.exe and you should be good to go.

Q: I'm trying to use the latest public build and it is not working. It complains that it cannot find portaudio_x86.dll or qcustomplot2.dll (or some other DLL).

A: Any downloaded public build exe must go in the same folder as your existing wfview.exe. Do not rename any files, just move it to the same folder. If you are missing a DLL file, please check the [public_builds](#) folder to see if there are any DLL files that you need. Those DLL files need to go in the same folder as wfview.exe. When you have that set up, double-click on the new version of wfview that you downloaded, which will have a filename like "wfview-master-202323332.exe" or similar. Keep the original wfview.exe file around in case you wish to revert to the older version.

Q: I'm trying to use the latest public build and it is not working. It says "The application was unable to start correctly (0xc000007b)".

A: Please make sure that you have all the DLLs from the `public_build` folder within the same folder as wfview.exe, which is also the same folder that you have placed the downloaded public build exe file. The error means that one (or more) DLL files is missing.

Q: I'm trying to run the linux build script and it says "permission denied" when I try to run the script.

A: Check the directions carefully. Make sure to add execute permission to the script:
`chmod 755 fullbuild- wfview.sh` . Do not ever run the build script as root.

Q: I'm trying to run the linux build script and when I run it, it says "./fullbuild-wfview.sh: line 2: syntax error near unexpected token newline" (or similar)

A: Please follow the directions carefully. Most likely, you have inadvertently downloaded some HTML instead of the actual build script. When you click on the link for the script, you must then press the download button on gitlab's web interface. This is also detailed in the [video](#) on linux install in our video section.

Q: How can I tell what version of wfview I have?

A: Press the About button.

Q: My wfview doesn't look like the ones I see in screenshots or on YouTube. Some features seem missing!

A: Please make sure you're running the latest version. You can check using the About button, and compare with what is offered on our [website](#).

Section 2: Audio

Q: I don't have any audio on wfview client

A: Check your log (press "Log") and see what audio device wfview is trying to use. Try disconnecting from the radio, changing the audio system, and then connecting again. You may also want to try other devices.

Q: Why isn't there any transmit audio?

A: Check what audio MOD source is selected on the radio. For convenience, you can also do this within wfview using the Radio Settings page under Settings. For radios with native LAN support, this should be set to "LAN", generally for Voice and Data modes. For radios where you are running your own wfview server, you'll most likely need to set it to "USB" or "ACC" — use whichever is being used by the server computer. Next, verify that the gain for the selected MOD source is at least 50%. The slider will be labeled "LAN" for LAN connected radios, and "USB" for USB connected radios, matching the selected MOD source. Finally, enable the secondary meter under Settings on the User Interface page. Set the secondary meter to "TxAudio". This will cause a second meter to appear on the main wfview page which will always display the transmit audio level as seen by wfview. This is not the same as the radio's audio level, this is the level of the audio being streamed to the radio. This level should be around -12dBfs to -6dBfs. Finally, if that audio is showing, change the secondary meter to ALC and see if the ALC meter is moving during transmit on SSB.

Q: The audio is distorted, too quiet, or too noisy

A: See if you can adjust the audio device gain in your operating system's system preferences or control panel. Use wfview's built-in "TxAudio" meter to view the audio level. Speaking should exceed -12dBfs, with some peaks over -6dBfs. Idle sound level should be below -30dBfs.

Q: The wfview log says "you may need to install libqt5multimedia5-plugins"

A: This only applies to some linux distributions, and the actual plugin name may vary. The error comes up when no audio devices are found. You may wish to try one of the other audio systems such as Port Audio or RT Audio.

Q: Why isn't there any audio in wfview for my radio connected with USB?

A: wfview does not need to do anything with the audio from a USB radio, unless you are running a [wfview server for remote access](#) to your radio. If you need to access the radio's audio, such as for a digital mode like WSJT-X or PSK31, simply select the radio's USB audio directly from the digital mode program. To use a headset with your USB-connected radio, you'll need to install some sort of audio loopback utility to direct headset audio into the radio. Again, this advice here only applies to the computer physically connected to a USB radio at the radio.

Q: (for network-accessed radios) Which audio device is my microphone? Which audio device is my speaker?

A: This is a tricky one. We present, in wfview, all the audio devices that the computer's operating system and libraries expose. You may want to experiment with each device one by one, or even use another program to help you test your computer's audio system carefully, so that you know what your devices are named. If you enable the TxAudio meter, it may help you with the microphone search.

Q: (for USB-connected radios, when running wfview as a server) Which audio device is my radio?

A: The exact name will depend a lot on the operating system. Windows often names the device “USB Audio CODEC”, and Linux may be similar or may even have “BurrBrown” in the name. If you see several, you may want to disconnect any and all USB audio devices and radios from the computer, close wfview, and then connect only the radio in question. Next, start wfview. If the list is still too cluttered, it can help to reboot with a minimum of audio devices connected.

Q: I picked up the mic on the radio and it isn't transmitting any audio! What happened?

A: These days, radios have so many audio inputs for transmit (modulation). You can change the transmit modulation source to “MIC” using the menus on the radio, or conveniently, we have a menu in wfview which also performs this task on the Settings tab under the Radio Settings page. Simply set it to “MIC” for your radio's microphone. You can read more about this [here](#).

Q: When I connect to the radio over the network, nobody can hear me and I don't see any transmit power indicated or ALC action.

A: Make sure you switch the radio's Transmit Modulation Input to “LAN” so that the radio transmits network audio. You can do this either using the radio's menus or using wfview's Radio Settings page under the Settings tab. You can read more about this [here](#). Additionally, make sure you have selected your computer's microphone correctly under the “Radio Access” page of the Settings tab.

Section 3: Waterfall Display

Q: Which radios have waterfall display?

A: wfview gets the waterfall data from the radio. Therefore, only radios which provide waterfall data over CI-V will show a waterfall display in wfview. These radios are: IC-705, IC-7300, IC-7850/7851, IC-9700, and the IC-R8600.

Q: Why is the waterfall display so slow?

A: For a variety of reasons, when you use a serial or USB port for radio connection, the waterfall rate is quite limited. This is a limitation of the radio firmware; wfview displays the data as fast as it is presented over CI-V. Radios which support network connections (Ethernet or WiFi) will have significantly faster waterfall when you connect over the network instead of USB.

Q: I have a supported radio for waterfall but I do not see any waterfall.

A: The radio *must* be set at 115200 baud in order for it to generate waterfall data. Additionally, some setup menus must not be open on the radio's screen, and the waterfall must be displayed on the radio display too. These are limitations baked into the firmware and we cannot change them. Please see our [IC-7300 guide](#) for complete setup information if you are having difficulty making this work. Additionally, if you change baud rates to 115200 (as required), you will need to cancel and/or disconnect with wfview and then re-connect when you change baud rate. This is simply a limitation of how serial ports are handled.

Q: Can I lower the CPU usage when drawing the waterfall?

A: Yes. Under Settings, on the User Interface page, disable Anti-Aliasing and disable Interpolation.

Q: Can I turn off the waterfall display:

A: Yes. There is a checkbox for this at the bottom of the waterfall display, “Enable WF”, which you can uncheck.

Q: Why is the waterfall so fuzzy?

A: Make sure to turn off Averaging and also set the Video Bandwidth Filter to “Narrow”. This will make a dramatic improvement in image sharpness.

Section 4: Sharing Control with Other Programs at the same time

Q: How do I run another program that controls the radio at the same time?

A: See [this overview](#) on the topic first. For programs supporting hamlib's rigctld, use wfview's [built-in rigctld-compatible server](#). For programs without this support, use the built-in [virtual serial port](#).

Q: How do I connect two programs at once to the virtual serial port?

A: That isn't supported. The Virtual serial port can only connect to one program at a time. On the other hand, the rigctld server built-in to wfview can handle multiple connections at once.

Q: How do I get wfview's client audio stream into another program?

A: You will need to create a loopback audio device. See [this page](#) for audio configuration.

Section 5: Display Issues

Q: The fonts all look too large

A: Check your operating system and see if you have any kind of Interface or Font Scaling enabled.

Q: How do I change the color of something in the plot? How do I make something disappear?

A: Go to [Settings](#), and then click on [the "User Interface" page](#). Use the color editor to select the desired colors. Setting a color to "#00000000" will make a given element invisible. More appearance notes are [here](#).

Q: Where is the SWR/ALC/Audio meter? I see it in videos but I don't see it in my wfview.

A: Look under Settings, and then click on [the "User Interface" page](#). Make a "Secondary Meter Selection" to choose one of the other meters offered.

Section 6: Networking

Q: What is my ip address?

A: We don't know. Every computer has a different method of showing your computer's ip address. You may want to look in the control panels or system preferences. Remember, your computer's ip address is not the same as your overall public ip address, which is assigned to your home router.

Q: What is my radio's ip address?

A: We don't know it. You can use the menus on the radio to find the ip address.

Q: I can't connect to my radio or radio server

A: Please see if you can [ping](#) from your client computer to the radio or radio server. Please check that there isn't a firewall on the client or server side that is preventing the connection (it may help to temporarily turn off any firewalls, just to eliminate this potential issue). Make sure the port numbers are correct. Make sure the username and password are correct. For Icom radios, make sure that network control is enabled in the radio's menu. Check the [log](#) on both the server and the client side for potential error messages that could be helpful.

Section 7: Protocol

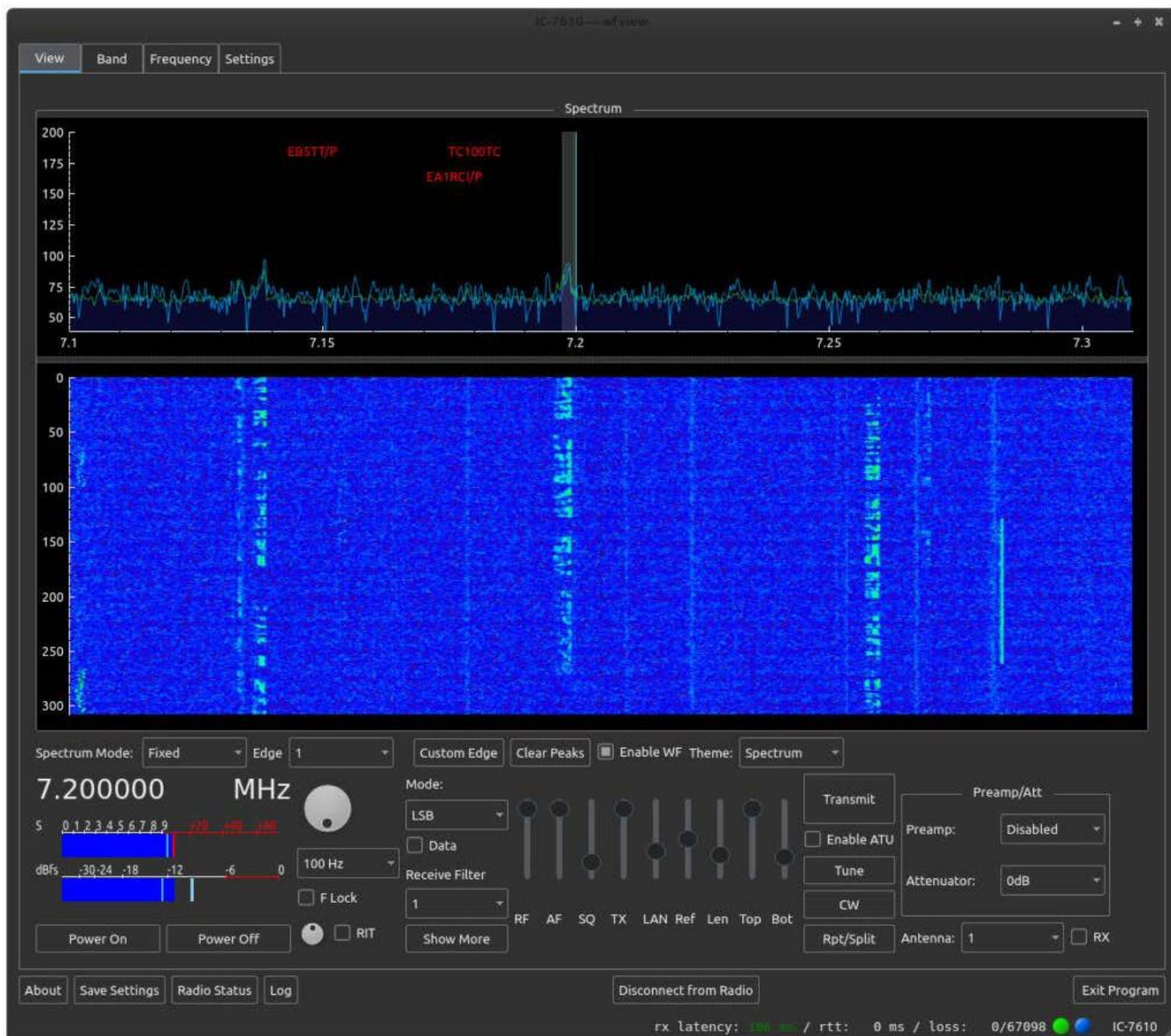
Q: What is the OEM networking protocol?

A: The networking protocol is quite complicated and it is not documented anywhere. wfview is open source software, and you can examine [the source code](#) if you want to see how it works. The protocol was first discovered by the [kappanhag](#) project. Additionally, [this open source project](#), written in swift, has some good notes on the protocol. The wfview team regularly makes improvements to our implementation of the networking protocol, which is compatible with every available Icom radio with an ethernet or wifi connection to date. The protocol is complex and filled with subtleties which are difficult master, let alone describe. We would welcome help in this area though, since a proper description of the protocol would be helpful to everyone.

View Tab

The View tab of wfview, also often called the “Main” tab, contains commonly-used controls for the radio and visualization.

Scroll down for a description of each element.



Spectrum plot and waterfall display:

These two items show the spectrum data passed from the radio to wfview. The vertical axis of the upper plot is likely proportionate to a value in dB, although the exact scale conversion isn't known. Left to right is frequency. The bottom plot's vertical axis is time, with newer values going in at the top. Many adjustments for these plots are available, read on to learn more.

Double-click anywhere in the plot to immediately jump frequency. A single click will simply show the frequency in the lower-left status bar area. Use the scroll wheel to quickly jog around the band.

The vertical tuning line in the upper plot indicates the *carrier point* of the radio. This is the frequency the dial indicates as well. The tuning line color can be customized (and even disabled via a transparent color) under the

Settings tab on the User Interface page.

There is a divider between the upper plot and the lower waterfall section. You can drag the divider up and down to change the division of area between the plot and the waterfall.

Spectrum Mode and Edge controls:

- **Spectrum Mode:** This control selects between the radio's built-in four modes: Center, Fixed, Scroll-C, and Scroll-F.
- **Span:** This control sets the total span for the spectrum, and is only available in Center and Scroll-C modes.
- **Edge:** In Fixed and Scroll-F modes, this control selects the current edge (1 through 4).
- **Custom Edge:** This button lets you define the scope edges in Fixed mode (in Scroll-F mode, the edge you define is translated over to the current frequency). The Custom Edge dialog box makes it very easy to jump to a desired frequency range.
- **ToFixed:** This button will take the current center span range and convert it into a fixed range. It is a quick way to keep the same displayed range and allow for tuning without changing the range displayed.
- **Clear Peaks:** Clears the spectrum underlay, which can be either Peak Hold, Buffered Peak Hold, or Average (select the desired mode under Settings).
- **Enable WF:** This checkbox can be used to turn off the waterfall data. This has the effect of freezing the waterfall display (good for taking a screenshot) and also makes the overall level of CI-V traffic a lot lower.
- **Theme:** Selects the data to color mapping scheme. The color themes come from QCustomPlot's built-in themes, and can be seen [here](#). Using the Top and Bot controls, the color scheme can be zoomed in around the data for very beautiful results.

Frequency and Tuning controls:

- **Frequency:** The current frequency is listed in large font (7.250 MHz). Tip: Press "*" (the star or asterisk key) to quickly jump to the Frequency tab and enter a new frequency).
- **Knob:** This is just a traditional tuning knob. You can use the scroll wheel of the mouse to turn it. Hold Shift or Control to increase the tuning step size.
- **Tuning Step:** This combo box (currently set at 100 Hz) sets the desired tuning step size. This tuning step is used for the tuning knob as well as scroll-wheel action over the spectrum and waterfall area.
- **F Lock:** This control locks the user interface tuning controls. It does not activate the radio's tuning lock function, and will not stop external programs from tuning. Use this control if you want to prevent accidentally clicking or tuning around the spectrum.

Meters:

wfview can display two meters at a time. The first meter is always power output during transmit, and receive strength ("S") during receive. The second meter can be selected under the User Interface area of the Settings tab. Meters include: SWR, ALC, Compression, Voltage, Current, Center (IC-R8600 only), TxRxAudio, RxAudio, and TxAudio. Please keep in mind that the audio meters will only function for the client side of network radios. The audio meters do not do anything with a locally-connected USB radio.

The meter displays three numbers at a time.

- **Instantaneous Level:** This is the long solid bar which shows you the same value as the radio's meter.
- **Average:** A thinner bright bar showing the average of the last ten or so readings.
- **Peak:** A thicker bright bar showing the maximum of the last ten or so readings.



The Meter Polling time can be adjusted under the User Interface page of the Settings tab. Shorter polling periods produce more responsive meters, but possibly at the expense of available CI-V bandwidth. wfview calculates a “safe” polling value on startup based on the type of radio connection, but faster values can be used if other programs are not polling as well.

Meter colors can be customized under the User Interface page of the Settings tab.

Power Controls:

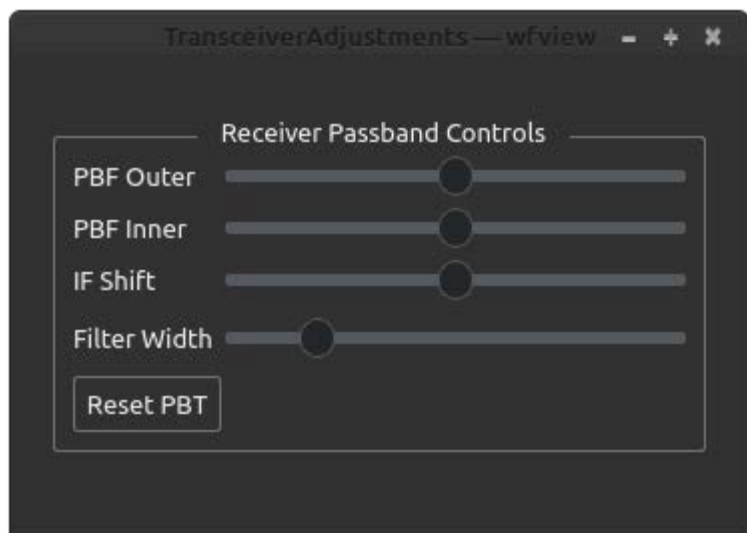
On radios supporting this feature, the radio power may be turned on and off. Supported radios usually maintain some standby power which allows the radio to be remotely turned back on. For these radios, wfview may appear unresponsive on initial connection if the radio is off; pressing Power On will alleviate this condition. wfview will pause all communication to the radio for about three seconds during power on.

Mode and Filter Controls:

The mode may be selected from the pop-down combo menu (LSB currently in this screenshot). Some common modes may be quickly selected using [keystrokes](#). The “Data” checkbox sets the data mode on the radio, for example, USB-D. Not all modes have a data mode.

The Receive Filter combo box allows for one of the radio’s existing filter definitions to be selected.

Show More opens additional radio controls such as IF-Shift. Not all radios support these controls:

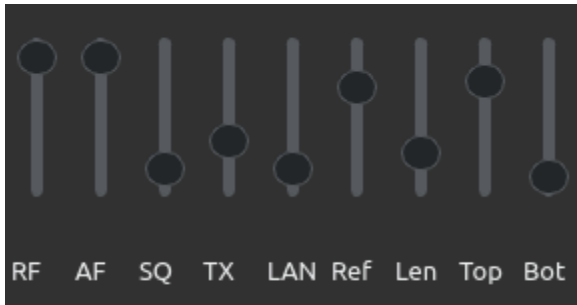


Use the PBF to shift the twin Pass Band Filter. The spectrum display will show the resulting adjustments to the PBF in red. Adjustments to the Filter Width will be in a lighter color. The IF Shift control can be used to simultaneously adjust both PBFs, simulating an IF Shift control.

Sliders:

The sliders in wfview are used as knobs for many common adjustments. The short names were chosen so as to allow the knobs to occupy less space in the interface. Each knob has useful “tooltip” text that can be seen by

hovering the mouse over the slider for a few seconds. The numerical value of each slider is read out in the status text area (lower-left corner of wfview).



- RF: This is the receive RF Gain
- AF: This is the receive audio level. For network-connected radios, this is the volume of the streaming audio. For serial (USB)-connected radios, this controls the volume of the radio's speaker. The level is saved when you save settings.
- SQ: Squelch. If you don't hear anything on FM, try lowering this slider :-).
- TX: Transmit Power Level.
- MIC/LAN/USB/ACC/UNK: This next slider will have a name based on the currently-selected transmit mod (audio) source. For network radios, this will likely be set to LAN, and for serial radios being served onto a network using wfview's built-in server, this will likely be set to "USB", indicating the use of the USB port. MIC indicates the microphone jack, ACC is the accessory input, and "UNK" indicates an unknown setting, generally one of the "dual" settings that some radios support such as MIC/ACC. You can pick the radio mod source under Settings on the Radio Settings page. If you can't transmit any audio, check here to see what the radio is set for, and adjust the level. It is a good strategy to verify audio by setting the meter to "TxAudio". Once audio levels are verified, set the meter to ALC and adjust the LAN slider until you see appropriate ALC action. Some radios may have additional level controls available in the radio menu.
- Ref: This is the so-called Scope Reference Level. It is actually just a level control, but anyway, this lets you set the level of spectrum data.
- Len: This is the length of the waterfall display. Set to short values, you can see a lot of details, and set to long values, a longer history. This value can be changed any time; the waterfall data is always buffered for the last 1024 lines of spectrum.
- Top: This control adjusts the ceiling of the spectrum plot and the top value for the color mapping of the waterfall. Generally this should be set to match the maximum value on the plot. Set lower will accentuate the lower signals.
- Bot: This control adjusts the floor of the spectrum plot and the bottom value of the color mapping of the waterfall. Generally this should be set to just a little bit below the noise floor, although higher values will create a higher contrast display of strong signals.

Transmit, Tuner, and Repeater Controls:

- Transmit: Causes the radio to transmit. Control-T and Control-R can also toggle this behavior. PTT lockout is available under Settings on the User Interface page, titled "Enable PTT Controls". While transmitting, this control will say "Receive"
- Enable ATU: This checkbox enables the radio's automatic antenna tuner. If the ATU fails to tuner, this will automatically uncheck.
- Tune: This button causes the ATU to tune. Tuning success (or failure) will be indicated in wfview's status text area at the lower-left.
- CW: Opens the CW Sender. See [here](#) for more information on the CW Sender.
- Repeater: Opens the Repeater Access Controls dialog box, where the tone, tone squelch, DPL, and repeater split direction can be set. Tip: Use the scroll wheel on the PL tone squelch combo box to quickly cycle through tones and find the repeater's output tone. See [here](#) for more information on split and repeater usage.

Preamp, Attenuator, and Antenna Controls:

- Preamp: Select from the radio's available receive preamps
- Attenuator: Select from the radio's available receive attenuators.
- Antenna: Select from the radio's available antennas.
- RX: Select the receive-only antenna (if available)

Bottom-Row Buttons (available in every tab):

- About: Brings up the about box, which displays the current version, git commit revision, and built date.
- Save Settings: Saves the current settings. Settings are otherwise discarded on exit.
- Radio Status: Shows the current radio and course audio levels. Future version of wfview may allow for multiple radios!
- Log: Displays wfview's internal message log. This is generally the same text as the logfile. If wfview is not behaving as expected, click [here](#) and view the log. Read more about this feature [here](#).
- Connect/Cancel/Disconnect: This button will connect to the radio, disconnect from the radio, or cancel an existing radio connection attempt. Look [here](#) to see if you have connected to the radio.

Bottom-Row Text:

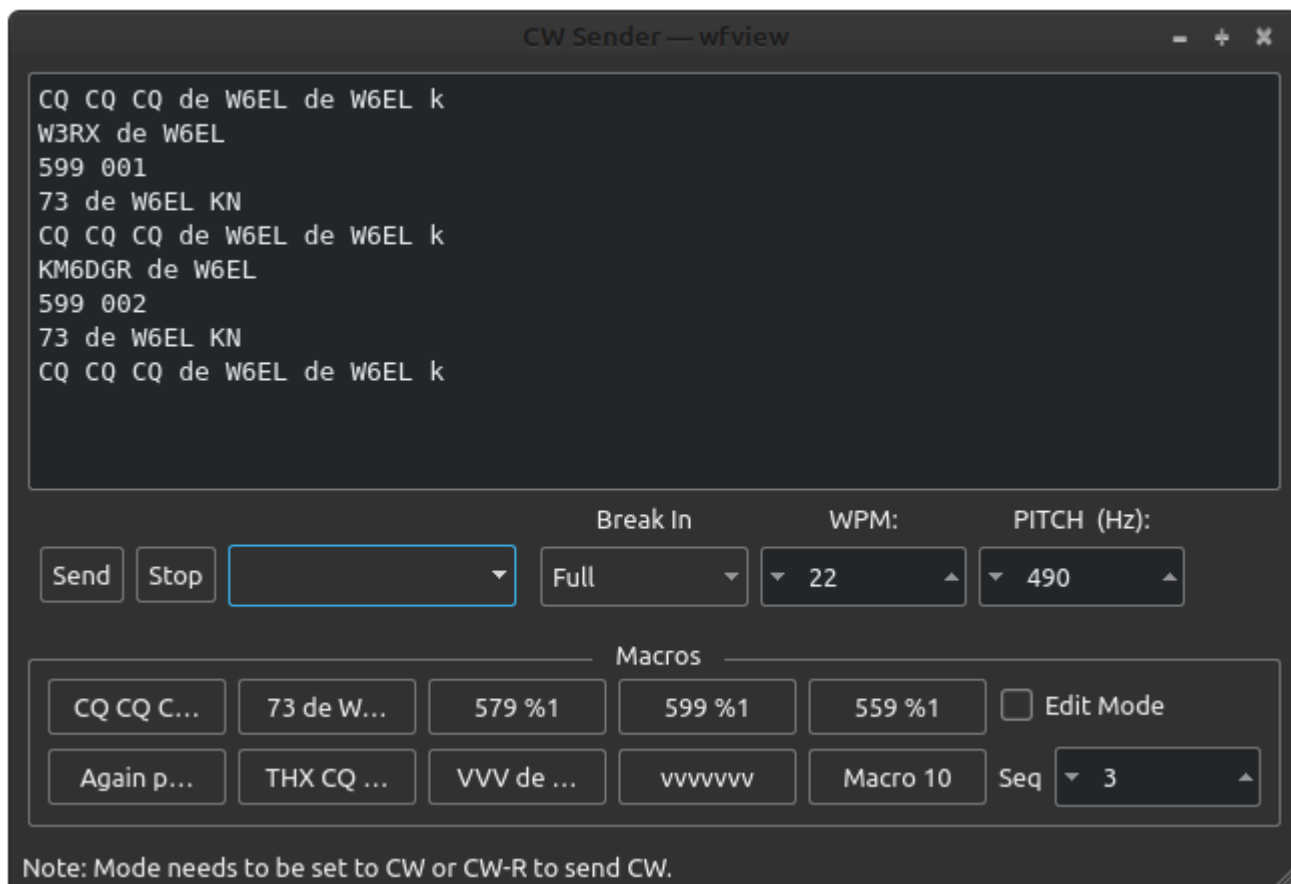
- Left side: Status text. Shows slider values during adjustment and other information
- rx latency: latency for receive audio.
- rtt: round-trip time for receive audio
- loss: lost packets / total packets. For example, 0/25332 indicates that zero packets have been lost out of 25332 sent.
- Color "LEDs": Green/Red LED indicates transmit and receive. I honestly forgot what the blue one means.
- NONE or Radio name: This text begins as "NONE" and changes to indicate the radio model number, for example "IC-718". If you see "NONE" here, then your radio model is not being identified, possibly indicating a CI-V issue such as CI-V Transceive being off (it needs to be on) or other problems.

CW Sender

wfview has a built-in CW Sender which uses the radio's built-in keyer. Text typed in by the user is sent to the radio for keying. A sidetone is not provided at this time, although it is possible to stream the sidetone from USB-connected radios (such as the IC-7300) when used with wfview as a server.

Note: The radio will not actually transmit any CW unless it is in CW or CW-R mode.

CW Sender Controls:

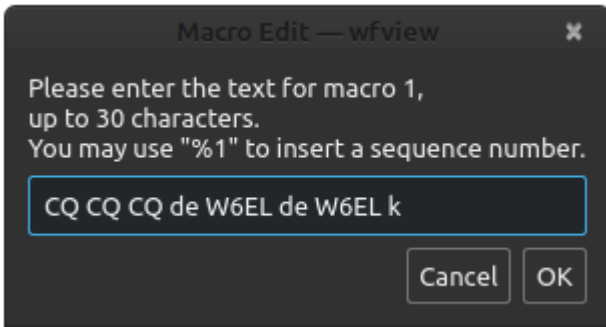


Top level controls as they appear:

- **Send:** This button sends the user text to the radio. In Full or Semi-Breakin mode, the radio will transmit and begin sending the CW.
- **Stop:** This button causes the radio to stop sending prior to actually finishing whatever text has been buffered from the user. Otherwise, stopping is automatic.
- **The text box** allows users to enter text to be sent to the radio. Up to 30 characters may be sent at once. Multiple stanzas of 30 characters may be sent to the radio, which will be sent out in the order they are received. Some characters are not allowed; check the log if you think your characters may be being sent oddly. Characters that aren't allowed get replaced with a "?".
- **Break In:** This combo box lets you pick the break In mode:
 - **Full:** The transceiver will switch T/R in between words whenever there is a gap. Some amplifiers may not be able to keep up!
 - **Semi:** The transceiver will transmit at the beginning of the CW stanza and receive at the end of all the buffered text.
 - **Manual:** You, the user, must press Transmit at the beginning and press Receive at the end. The radio starts "sending" the CW immediately, so press Transmit quickly!
- **WPM:** Words per minute. The radio will accept 6 to 48 WPM.
- **Pitch:** The desired CW pitch, referenced to the tuned carrier frequency.

Macros

wfview provides ten user-defined macros for common phrases. The macros are saved when the user presses “Save Settings”. Macros may be edited by pressing “Edit Mode” and then clicking on a macro button.



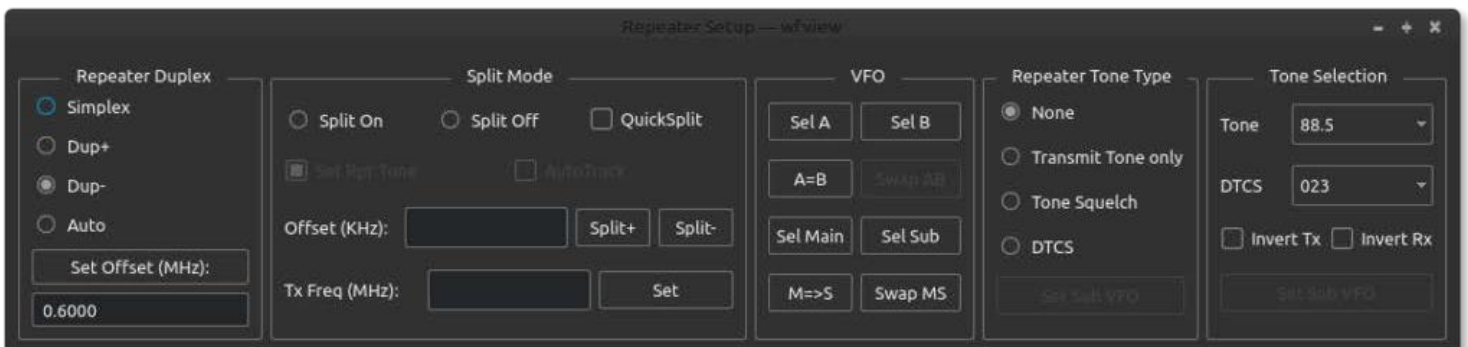
You can type in “%1” (without the quotes) to have wfview automatically replace the %1 with the current sequence number (otherwise referred to as a “serial” number). This is useful in some contests.

The sequence number may be changed using the up and down spinbox to the right labeled “Seq”.

Uncheck the “Edit Mode” checkbox to use the macros. Simply click a macro button to send it over the air.

Repeater and Split operation

This page describes how to use the SPLIT and Repeater settings of the radio. Press the “Rpt/Split” button in wfview to bring up this window:



Note: When transmitting with split enabled, the transmit indicator “LED” will have a split appearance:



Supported Features

What you see in this window will reflect the features the radio supports. For example, most HF radios do not support the “Repeater Duplex” section but do support using Split Mode. Some radios support Main and Sub VFOs, and some support A and B VFOs. Additionally, exactly how some features on the sub VFO are addressed varies quite a bit.

The IC-7610 and IC-7850 have special commands to allow the repeater tone to be set on the sub VFO; the other radios lack these commands.

Repeater Duplex

This section deals with repeater controls, primarily present on radios for accessing VHF and UHF FM repeaters.

- Simplex: transmit and receive frequencies are the same
- Dup+: The transmit frequency is above the receive frequency
- Dup-: The transmit frequency is below the receive frequency
- Auto: The radio should automatically decide which duplex mode to use (may not work correctly depending on your region)
- Set Offset (MHz): Type in the desired offset, in MHz, and press this button to set the default repeater offset for the currently selected band (ie, 2M/70CM/23CM)

Split Mode

This area handles split operation, available on most HF rigs and some VHF/UHF/SHF rigs

- Split On: Turns on the split feature, disabling any other duplex or repeater mode selected
- Split Off: Turns off the split feature. Some radios have both “split off” and “simplex” modes, but I can’t tell you what the difference is.
- Quick Split: Activate the radio’s quick split feature, when available. This feature is supposed to sync the repeater tone, mode, and frequency of the sub VFO. However, it can be “canceled” if the user makes certain adjustments to the sub VFO. The checkbox is available but you may need to experiment with it.
- Offset (KHz): Type in the desired split offset here, for example, 100 for 10M FM repeaters.
- Split+ and Split- buttons: These buttons set the radio’s secondary VFO (“B” or “Sub” depending upon the radio) to the appropriate split frequency and populate the Tx Freq box as a convenience to the operator. If the “Set Rpt Tone” checkbox is checked, and the radio supports such commands (7850 and 7610), then the secondary VFO’s repeater tone and tone mode is also set at this time.
- Tx Freq (MHz): You can manually type in a transmit frequency here and press “Set” any time. But it is not needed when using the Split+ and Split- buttons
- AutoTrack: Checking this box causes the sub VFO to automatically track the Main or “A” VFO with supplied Offset. In FM mode, the repeater tone and repeater tone mode will also track. Use this feature to discover 10M repeaters easily. As this function is similar to Quick Split, do not use both at the same time.

VFO Buttons

These buttons are for selecting, swapping, and equalizing VFOs.

- Sel A, Sel B, Sel Main, and Sel Sub: These buttons “select” a given VFO. On the radios I have, this is the same as touching the given VFO on the screen and does not provide much functionality. But it may be different on other radios.
- A=B: Sets the B VFO to the contents of the A VFO
- M=>S: Sets the Sub VFO to the contents of the Main VFO
- Swap A/B and Swap M/S: Swaps the contents of the two VFOs. Use this feature to rapidly swap between VFOs and to make adjustments to the Sub VFO when needed, for example, with radios that do not support addressing the SUB VFO directly and if you need to set a repeater tone.

I realize some of this seems incomplete or difficult to understand. I recommend playing with these features first with the radio at your side.

Repeater Tone Type

These buttons select the method of repeater tone access.

- None: There is no transmitted tone of any sort and there is not a tone squelch.
- Transmit Tone Only: A subaudable tone is transmitted. No tone squelch.
- Tone Squelch: A tone is transmitted, and the same tone frequency is used as a tone squelch.
- DTCS: Also known as DCS, a digital code is sent with your transmission and is also used as a digital tone squelch.
- Set Sub VFO: Available on the IC-7610 and IC-7850. Use this button to force the sub VFO to a specific tone mode (typically "Transmit Tone only"). This button's function is automatically run when the "Set Rpt Tone" checkbox is checked.

Tone Selection

Use these controls to select a Tone and/or D(T)CS code. For D(T)CS, the Transmit and Receive code can be inverted.

Press "Set Sub VFO" to force the sub VFO to the specified tone selected. Not available on most radios and not available for D(T)CS. This button's function is automatically called when setting split and the "Set Rpt Tone" checkbox is selected.

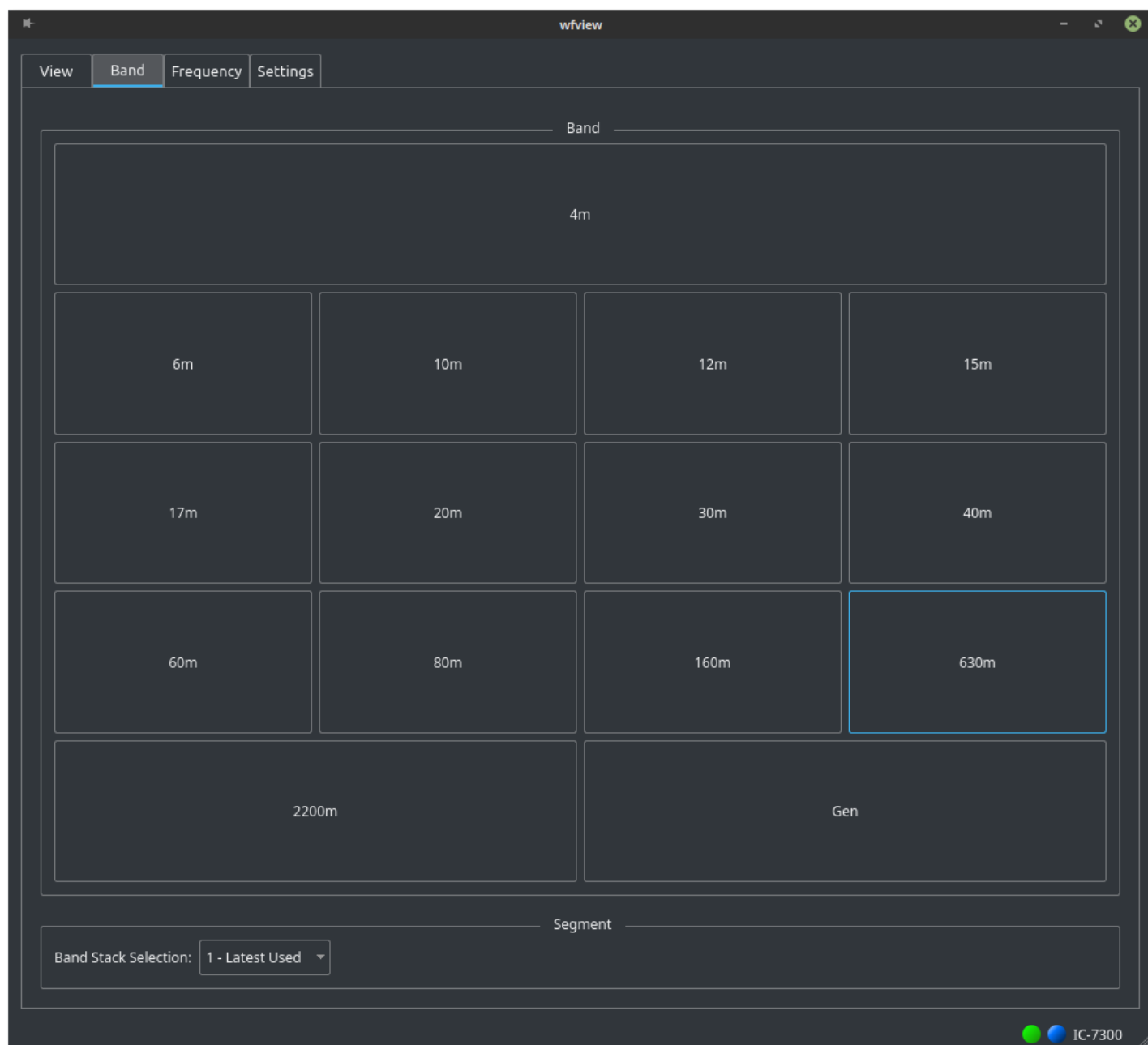
Radios and Features

For the curious, here's a short chart on what features are available on what radios:

Model	VFO A/B/M/S Operations		0x25 0x26	0x29
	Has MS 0x07	Has AB 0x07	Spe VFO Bnd/Mode	Specify M/S
IC-705	No	Yes	Yes	No
IC-706	No	Yes	No	No
IC-910H	No	Yes	No	No
IC-7000	No	Yes	No	No
IC-7100	No	Yes	Yes	No
IC-7200	No	Yes	No	No
IC-7300	No	Yes	Yes	No
IC-7410	No	Yes	No	No
IC-7600	Yes	No	No	No
IC-7610	Yes	No	Yes	Yes
IC-7700	No	Yes	Yes	No
IC-7800	Yes	No	No	No
IC-7850	Yes	No	Yes	Yes
IC-9100H	Yes	Yes	No	No
IC-9700	Yes	Yes	Yes	No
IC-R8600	undocumented	undocumented	Yes	No
9700, 9100	For radios with MS and AB: Command 07B0 is Exchange M/S			
	For radios with AB only: Command 07B0 is Exchange A/B			
	For radios with MS only: Command 07B0 is Exchange M/S			

Band Tab

This is the Band tab of wfview:



This is the band tab of wfview

From this tab, common ham radio bands can be quickly selected. These buttons utilize the radio's built-in Band Stack Register, or "BSR". Briefly, the radio remembers the last three frequencies and modes used on each band. You can use this screen to access those stored frequencies.

It's worth noting that a number of bands do not have actual BSR entries. For those bands, wfview will take the rig to a known calling frequency. The bands not in the BSR are: 60m, 630m, and 2200m.

To go to a band, simply click the band's button. Optionally, you may select which of the three band segments into the band stack to go to by using the combo box menu at the bottom.

You can get to the Band tab either by selecting the tab with the mouse, or pressing F2.

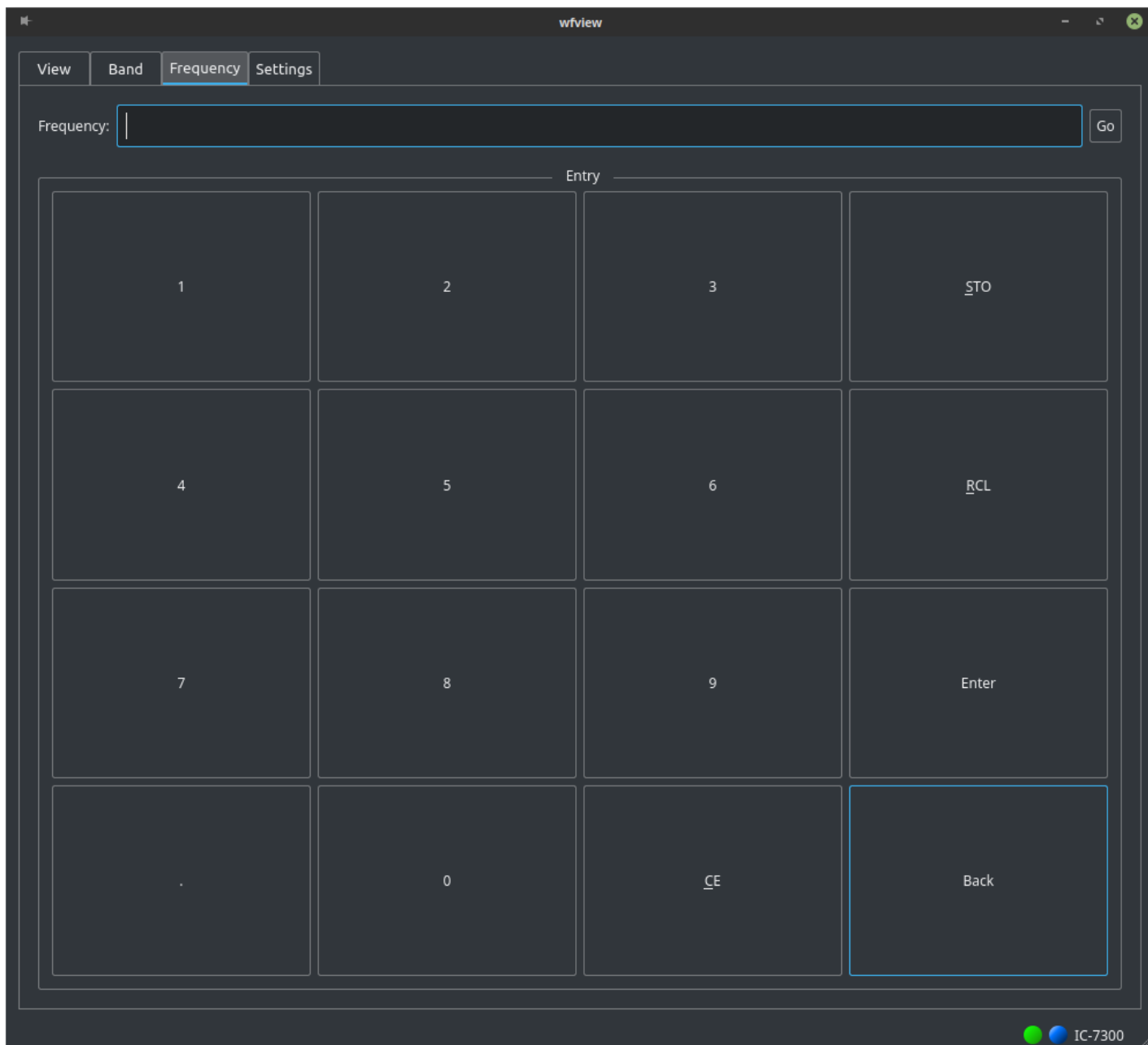
Many of these buttons have convenient preset keystrokes, enabling a skilled operator to switch between bands rapidly.

Here are the current band keystrokes:

Key	Function
6	Jump to 6 meters
1	Jump to 10 meters
t	Jump to 12 meters
5	Jump to 15 meters
7	Jump to 17 meters
2	Jump to 20 meters
3	Jump to 30 meters
4	Jump to 40 meters
s	Jump to 60 meters
8	Jump to 80 meters
x	Jump to 160 meters
g	Jump to General coverage
Shift+4	Jump to 4m
w	Jump to WFM band
a	Jump to Air band
v	Jump to 2 meters
u	Jump to 70cm
Shift+S	Jump to 23cm

Frequency Tab

This is the Frequency tab of wfview:



This is the frequency tab of wfview.

From the frequency tab, frequencies may be typed in using either the buttons on the screen or the computer keyboard. Frequency input defaults to units of MHz, unless the period is omitted, in which case the frequency will be interpreted as KHz.

To enter a frequency, simply press the buttons to “type in” the frequency. Press Enter on the keyboard or on-screen keypad to bring the rig to the desired frequency. The main View tab will be re-selected upon entry.

The Frequency tab has a memory function similar to that of a simple calculator. Frequencies and current mode may be saved and recalled. These memories are not stored on the radio, rather, they are kept in the wfview preference file.

To store a memory, use the following sequence:

1. Go to the desired frequency and mode as normal.
2. Go to the Frequency tab
3. Type in the desired memory location, from 1 to 99
4. Press “STO” (or just hit “S”).

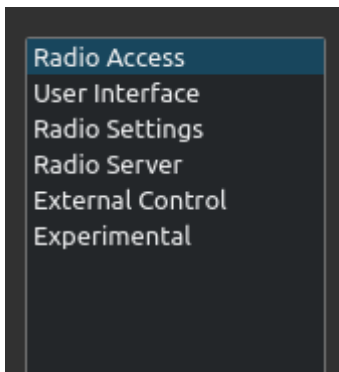
To recall a memory, use the following sequence:

1. Go to the Frequency tab
2. Type in the memory location number, from 1 to 99
3. Press “RCL” (or just hit “R”).

You can quickly get to the Frequency tab by pressing F3. You can also quickly jump to this tab and enter frequencies by using the asterisk (*) key on a numeric keypad. This will bring you to the frequency entry box. Now type in the desired frequency followed by Enter, and you will jump back to the View tab. Using this quick method, a stand-alone numeric keypad can be used to tune the radio. Other numeric keypad controls include the forward slash (/) key, used to cycle between common modes, and the + and – keys, which adjust frequency per the selected tuning step.

Settings Tab

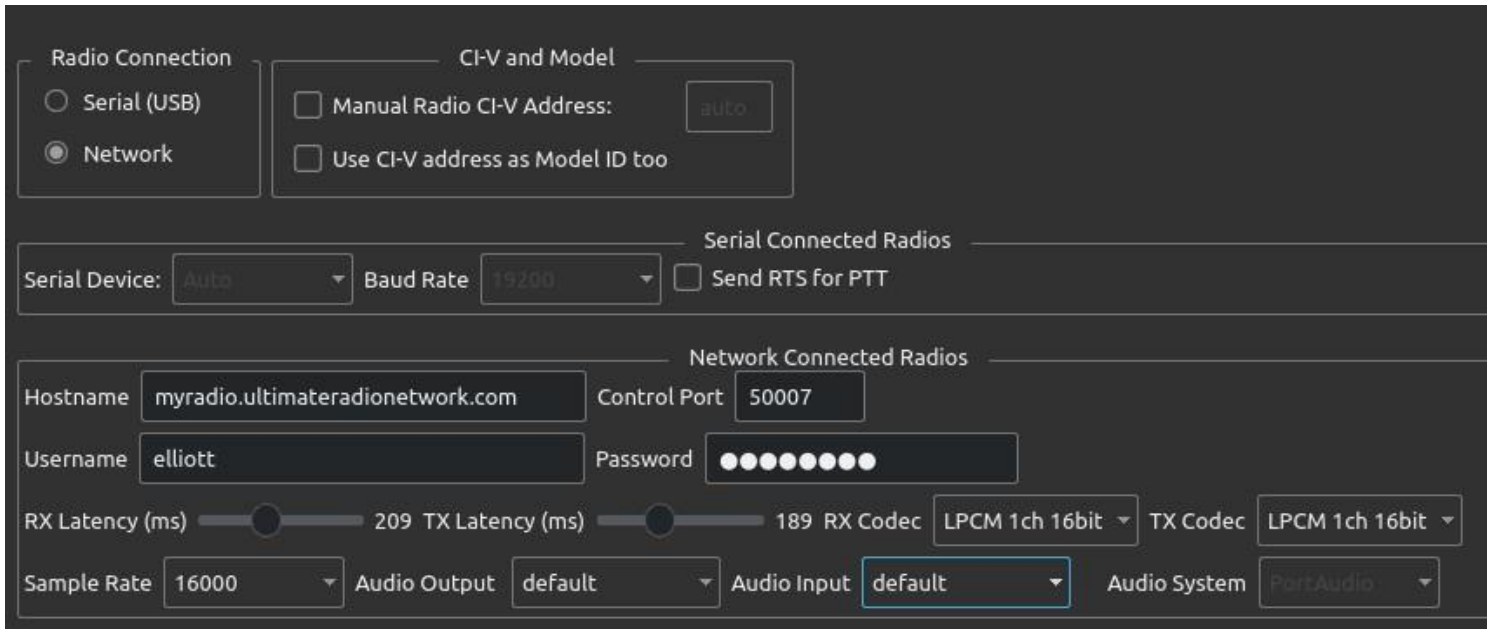
The Settings tab has several “pages” of settings:



Each page contains categories of settings. The settings are saved in a format detailed [here](#). Quickly jump to each page:

Radio Access

Under Radio Access are controls relating to how wfview access your radio:



The screenshot shows the 'Radio Access' configuration window. It is organized into several sections:

- Radio Connection:** Two radio buttons are present: 'Serial (USB)' and 'Network'. The 'Network' option is selected.
- CI-V and Model:** Two checkboxes are shown. The first is 'Manual Radio CI-V Address:' with a text input field containing 'auto'. The second is 'Use CI-V address as Model ID too'.
- Serial Connected Radios:** A 'Serial Device:' dropdown menu is set to 'Auto'. The 'Baud Rate' dropdown is set to '19200'. There is a checkbox for 'Send RTS for PTT' which is unchecked.
- Network Connected Radios:** The 'Hostname' field contains 'myradio.ultimateradionetwork.com'. The 'Control Port' field contains '50007'. The 'Username' field contains 'elliott'. The 'Password' field is masked with dots.
- Audio Settings:** 'RX Latency (ms)' is a slider set to 209. 'TX Latency (ms)' is a slider set to 189. 'RX Codec' and 'TX Codec' are both dropdown menus set to 'LPCM 1ch 16bit'. 'Sample Rate' is a dropdown set to '16000'. 'Audio Output' and 'Audio Input' are dropdown menus set to 'default'. 'Audio System' is a dropdown set to 'PortAudio'.

Radio Connection:

- Serial (USB): Select this option if your radio connects to this computer using a Serial port, USB port, or Serial CI-V USB adapter.
- Network: Select this option if you are accessing your radio over a network

CI-V and Model:

These controls should not be used except for very old radios that do not support CI-V Radio ID queries or for unusual circumstances. wfview automatically figures out the CI-V address and model number for most radios. Don't forget to turn on CI-V Transceiver, as we use that feature to automatically find your radio.

- Manual Radio CI-V Address: Check this box to manually set the radio's CI-V address. The input format is hex, which is the same as the radio's menu and documentation. For example, if you have an Icom IC-756 Pro, enter "5C" here for the default CI-V address on this model. **We recommend not checking this box** unless absolutely necessary. Enable CI-V Transceiver on your radio instead.
- Use CI-V address as Model ID too: Check this box to cause wfview to infer the radio's capabilities by assuming the provided CI-V address is the same as the model number. This is needed on older radios that do not respond to Radio ID queries (command 0x19 0x00). Set the radio to the default CI-V address and enter it in as a manual address above. This is not needed on most radios made after the IC-718 (late 90s). Just enable CI-V Transceiver on the radio and leave this unchecked.
- Usage Tip: You can also use these two boxes to cause a radio to masquerade as a different model, which is useful if wfview doesn't have support for your radio but you know that the capabilities are similar enough to another model. Simply set the radio's CI-V address on the radio to the default address of a supported model, and then enter this address in manually in wfview, and check the "Use CI-V address as Model ID too" box.

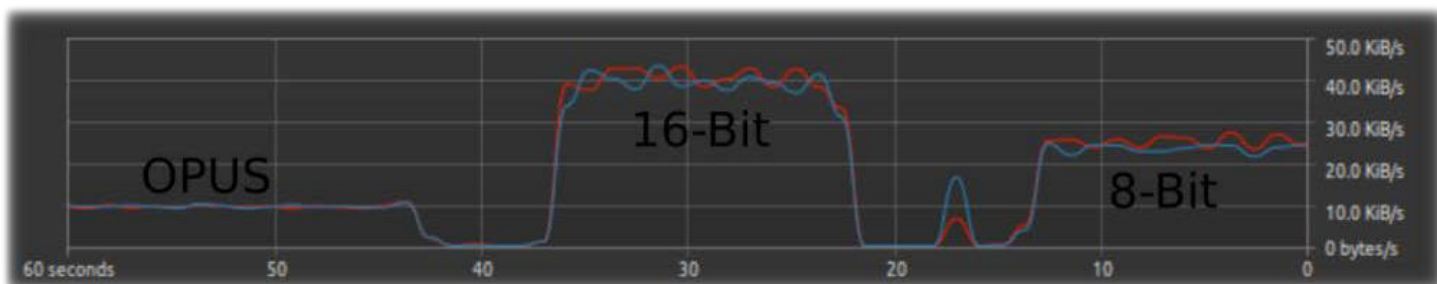
Serial Connected Radios:

- **Serial Device:** Select the serial device entry here. On Linux, "Auto" will automatically select OEM Icom radios with built-in USB ports. Use "Manual" if you need to manually enter a port that isn't on the list (for example, if you are using a mysterious serial port)
- **Baud Rate:** Select the radio's baud rate. It's recommended to set the radio to the highest-supported baud rate unless there are problems. For the IC-7300, you must set the baud rate to 115200 in order to get spectrum data.
- **Send RTS for PTT:** check this box if the radio does not have a PTT command. This will cause wfview to toggle the RTS serial line to indicate PTT, which is supported on many rig interface boxes. wfview will intercept any PTT command from external programs and toggle the RTS as well.
- **Tip:** Linux users can automatically set useful names for USB serial devices with reasonable default permissions. See [this page](#) on how to do it.

Network Connected Radios:

- **Hostname:** Type in the radio's (or radio server's) IP address or hostname here.
- **Control Port:** Type in the radio's control port here. The default value is 50001. The other ports used will be ascertained via the control channel automatically.
- **Username:** Type in the user name as is set on the remote radio here.
- **Password:** Type in the password as is set on the remote radio here.
- **RX Latency (ms):** This is the size of the receive audio buffer, in milliseconds. Larger values are needed for less-stable connections.
- **TX Latency (ms):** This is the size of the transmit audio buffer, in milliseconds. Larger values are needed for less-stable connections. **Tip:** Set the RX latency control to the lowest value and advance until there are no dropouts. Advance about 20ms further, and then set the TX Latency to the same value.
- **RX Codec and TX Codec:** These select the codec for the audio going to and from the radio:
 - **LPCM 1/2 ch 16bit:** PCM Lossless encoding, one (or two) channel, 16-bits per sample
 - **LPCM 1/2 ch 8 bit:** PCM Lossless encoding, one (or two) channel, 8-bits per sample
 - **uLaw 1/2 ch 8 bit:** Logarithmic bit space 8-bit per sample. This is a higher-quality format than LPCM 8-bit, but still has the advantage of only needing 8 bits per sample.
 - **Opus:** This is a high-quality perceptually-lossless (but technically lossy) format, which provides a 4:1 reduction in audio bandwidth. Opus is only supported when the radio server is another computer running a wfview server, or the Xieigo X6100. Opus is the recommended choice when the other side is wfview. If selected on an unsupported direct-to-radio connection, wfview will default back to LPCM 16 bit.
 - **Tip:** Some VPNs and ISPs re-order and mangle UDP packets, causing 16-bit packets to get split. Using uLaw 8 bit or Opus will solve this problem.

Here is a graph showing the network bandwidth in kilobytes per second required for the Opus codec, 16-bit LPCM, and 8-Bit LPCM (same as 8-bit uLaw). Each was taken with a 1-channel format with a sample rate of 16 KHz:



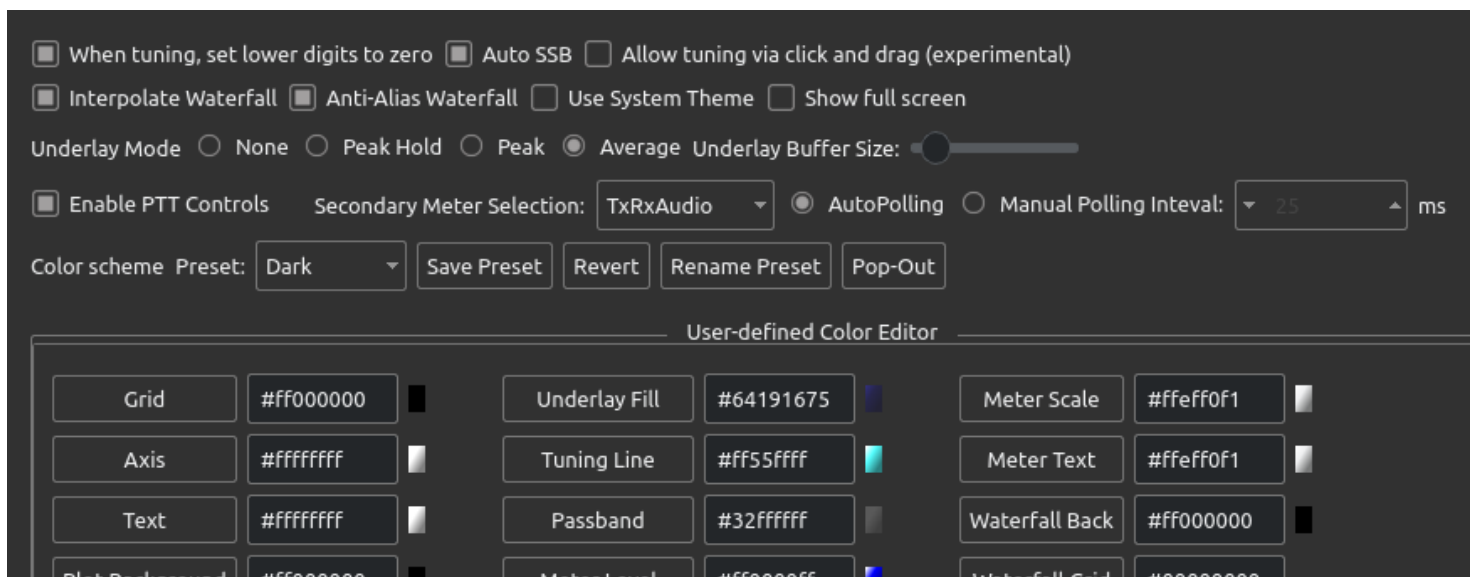
- **Sample Rate:** Select the desired sample rate here. Remember, the audio will never exceed half the sample rate. For example, using 16 KHz as a sample rate will limit the audio to just under 8 KHz, but will also require only 1/3rd the bandwidth of a 48 KHz sample rate connection. Lowering the sample rate is an excellent way to reduce network bandwidth. Values below 16 KHz may sound a bit lower in quality, but at 16 KHz and above is generally favorable. wfview takes care of converting the audio for your computer's hardware sample rates.

Audio Output: Select your computer's speakers or headset speakers here. Some platforms will provide a "default" device, which is generally acceptable.

- Audio Input: Select your computer's microphone or headset microphone here. Again, some platforms provide a "default" device.
- Audio System: This lets you select from the three available audio systems. The differences between the different systems can include latency, device sharing, and reliability. To change audio systems, first disconnect from the radio, select an audio system, and then, very important, re-visit the Audio Input and Audio Output combo boxes. Do not forget to click on these boxes and make sure to re-select the device. The devices tend to get re-ordered and renamed, so it is worth clicking and reading all the available choices each time. Once selected, you can then re-connect to the radio.
 - Qt Audio: This is the default choice and should work on all platforms.
 - Port Audio: This system may provide a more consistent stream of audio for some users.
 - RT Audio: For some people, this is a lower-latency choice
 - Tip: If your current audio system works, just leave this alone and don't worry about it :-).

User Interface

These settings determine many aspects of the interface and user experience in wfview.



- When tuning, set lower digits to zero: This control causes the tuning dial to automatically set the below-tuning-step digits to zero. This is generally desired. Uncheck this control if you want to dial in and keep an offset applied to all your tuning, for example if you want frequencies always ending in 5 KHz for some reason.
- Auto SSB: This setting will automatically select the "correct" sideband when switching frequencies. This is only done if you are already in LSB or USB mode, and only if you change frequencies using the Frequency tab (or hit the star/asterisk key).
- Allow tuning via click and drag: This checkbox made it into the master branch accidentally. If enabled, you can try clicking and dragging on the spectrum to tune. It is an experimental option and still under development.
- Interpolate Waterfall: If you have a big screen, check this box and the pixels of the waterfall will not look quite as blocky and pixelated. Checking this box can add significant CPU use, so do not do this on a humble platform like a Raspberry Pi.
- Anti-Alias Waterfall: Check this box for anti-aliasing on the waterfall display. This seems to have almost no impact on the performance or appearance! But it's an option if you want it.
- Use System Theme: Check this box to switch from wfview's built-in theme ([QDarkStyleSheet](#)) to your desktop environment's default appearance. Read more about this [here](#).
- Show full screen: Check this box for full-screen mode. You can also use F11 to toggle full screen. Tip: For a dedicated wfview display, use a raspberry pi set to launch wfview on startup under the default user. Check the full screen box and save the settings.

- Underlay Mode: These four options select an “underlay” for the spectrum plot. The underlay is an additional plot shown under the radio’s spectrum data. Two of the underlay options obtain data from a variable-size buffer of received spectra, in which one value is deleted each time a new value is received. The size can be set using the buffer size slider. The available underlay options are:
 - None: no underlay drawn
 - Peak Hold: An underlay is drawn showing the maximum value received. The peaks do not expire except when tuning.
 - Peak: A buffered peak mode. The peak value in the buffer (as set with the Buffer Size slider) is plotted for each frequency.
 - Average: The average value for each frequency is plotted. This provides a “smoothed” spectrum mode which often helps beat down broad-band noise such as RFI.
- Underlay Buffer Size: Sets the size of the underlay buffer. Lower values will be more responsive, higher values will be more “sticky”. Tip: Press “Pop-Out” to see the effects of adjusting this slider in real-time as you make the adjustments.
- Enable PTT Controls: Enables wfview’s transmit button. This checkbox has no effect on the radio’s physical PTT buttons, nor does it block any other radio control programs from sending PTT commands. It is designed to prevent inadvertent transmission, and could be useful during open radio demonstrations. wfview has a built-in 3 minute timeout on transmit.
- Secondary Meter Selection: Select a desired secondary meter here. Some radios do not support some meter types.
 - Available meters: None, SWR, ALC, Compression, Voltage, Current, Center (IC-R8600 only), TxRxAudio (switches Tx/Rx automatically), RxAudio (RX audio all the time), TxAudio (TX audio all the time)
 - See more about the meter under the [View Tab](#) manual page.
- Polling: wfview automatically determines the optimal polling rate for the radio. However, faster polling will make the meters more responsive and is generally fine so long as a secondary program isn’t polling as well. Slower polling rates may increase reliability, especially with older radios using single-wire CI-V. If you are running wfview as a server as well, it can make sense to set the polling period to a high value on the server so that the server isn’t contributing much to the overall polling activity.

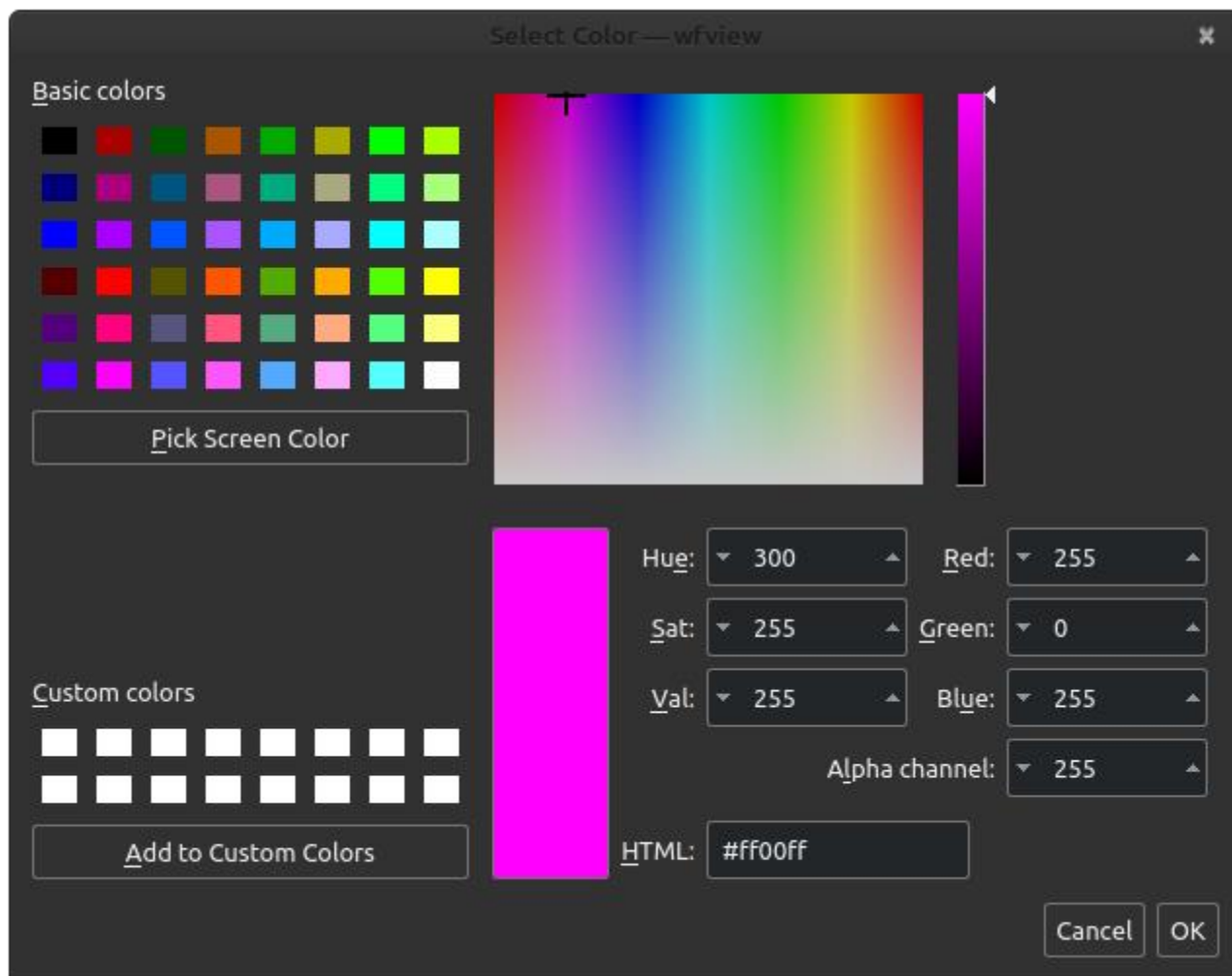
Color Scheme:

Use these controls to adjust the color scheme. Pressing “Pop-Out” will let you see the color scheme changes as you are working on them. More information on customizing the appearance is available [here](#).

- Preset: This control is for recalling the five available color scheme presets. Presets are saved with the settings and may be renamed and reverted.
- Save Preset: This button will save the selected preset’s colors and name to the settings file.
- Revert: Press this button to revert the selected preset to the default values.
- Rename Preset: Press here to rename the selected preset. Names may be up to ten characters long and may contain spaces and probably even unicode characters.
- Pop-Out: Read carefully, Jumangi players. Pressing this button pops the Settings tab out into a separate window. This will let you see the changes you are making on the View tab. When you are done making changes, you must press the button a second time to re-dock the Settings window back into a tab. The button will say “re-attach” in this state, and the pop’d out Settings window *cannot be closed until you press this button*. The first player to reach the golden city and yell Jumangi wins.

User-Defined Color Editor

Use these buttons to change the colors of many of the user-interface elements of wfview. The color editor allows a custom color to be selected. Simply press the button for a given attribute (such as “Grid”, “Axis”, “Text”, etc) and the color picker will be shown:

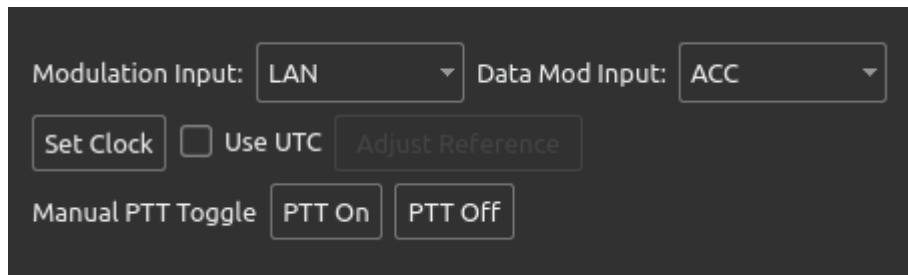


The color can then be previewed. Important: It is easy to accidentally set the Alpha channel to a low (transparent) value and not realize it, since the picker does not demonstrate the transparency. So be mindful. On the other hand, you can set an element to fully transparent (value 0) to completely hide it. wfview will warn the user if a color is set to fully transparent, but it is fine to do so.

You can also simply type in an HTML-style color in the text boxes. The format is #AARRGGBB, where AA is the transparency (00 for fully transparent, ff for fully opaque), RR is red, GG is green, and BB is blue. The “#” is optional and will be added automatically. Don’t forget to save the preset you just worked on, otherwise it will be gone next time you open wfview.

Radio Settings

These settings are convenience functions to help with radio operation.



- Modulation Input: Select from the available transmit audio choices:
 - LAN: Network audio for radios with built-in network support (this is what most network users will want)
 - USB: USB port for radios with a USB port (this is what IC-7300 users will need for remote IC-7300 access using wfview's built-in server)
 - MIC: Microphone jack on the radio
 - ACC: Accessory Port
- Data Mod Input: This is the transmit audio source for data modes. Generally not set to MIC. Usually you want this set to LAN or USB.
- Set Clock: Sets the radio's clock to the same as the computer's clock. wfview will send the time command at the next change-of-minute. Watch the status bar carefully and you may catch it.
- Use UTC: Sets the radio's clock to the current UTC time, again using the computer's clock as a reference.
- Adjust Reference (IC-9700 only): Brings up the reference adjustment window. It is very useful to switch to the view tab and zoom in on a reliable signal while adjusting these controls.
- Manual PTT Toggle: Use these buttons to manually toggle the PTT on or off. Useful if the View tab's Transmit/Receive button becomes out-of-sync with the world.

Radio Server

This page contains settings for the [built-in radio server](#). This server is designed to take a serial or USB port radio, with audio, and allow it to be accessed over a network by another copy of wfview running on another computer. The radio server handles full-duplex streaming audio and radio control. The server is largely compatible with the Icom OEM protocol but has been extended to allow for some additional functionality.

The wfview team has tested this feature extensively, with radios as old as the IC-736 and as modern as the IC-7300. Even across the ocean, audio and control can work quite well.

Enable

Server Setup

Control Port Civ Port Audio Port

RX Audio Input TX Audio Output Audio System

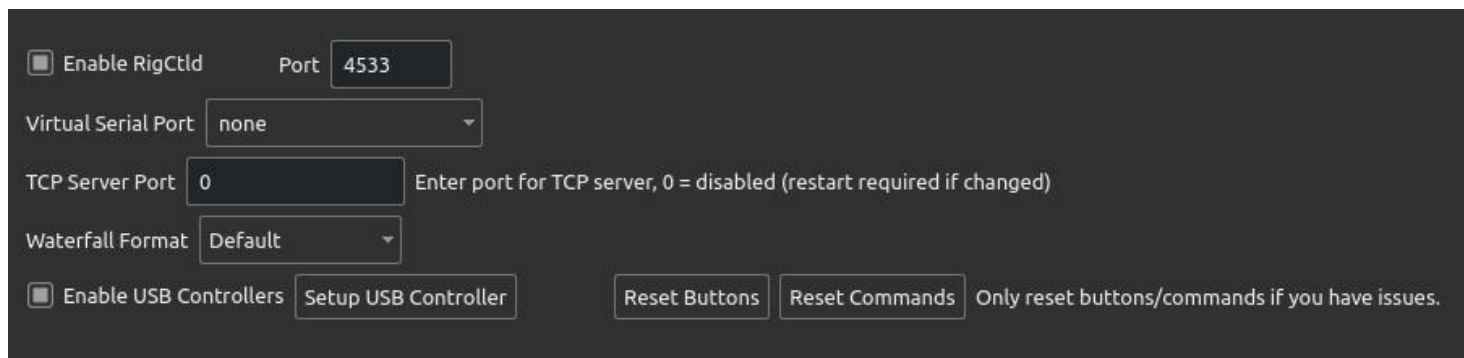
Username	Password	Admin	Delete?
John	●●●●●●●●	Full with no TX ▾	Delete
Paul	●●●●●●●●	Full User ▾	Delete
George	●●●●●●●●	Full User ▾	Delete
Ringo	●●●●●●●●	Monitor only ▾	Delete

Brief description of controls:

- Enable: Check this box to enable the built-in server
- Control Port: This is the primary port for the server, used for server-related control tasks. The default is 50001.
- CI-V Port: This is the port through which CI-V traffic is passed. The default is 50002
- Audio Port: This is the port through which audio is passed. The default is 50003.
- Note: All traffic is UDP

External Control

These settings help other programs control the radio while wfview is open. This is useful if you like to use digital modes or logging programs. The best and most simple-to-use option is wfview's excellent Hamlib Rigctl server, which emulates the hamlib rigctl program for TCP/IP control. Almost any program written to support Hamlib can utilize this mode of control. When TCP/IP isn't an option, the built-in Virtual Serial Port (pseudo-terminal) support will create a fake serial port which other programs can connect to as though connecting to a genuine radio's serial port directly. The Virtual Serial Port can only accept one program at a time, so if you need more than one, try and use the rigctl option, as it can allow for multiple clients at once.



Enable RigCtl Port

Virtual Serial Port

TCP Server Port Enter port for TCP server, 0 = disabled (restart required if changed)

Waterfall Format

Enable USB Controllers Only reset buttons/commands if you have issues.

Tip: The Hamlib model number for network control is #2, called "Hamlib NET rigctl (stable)". The device string is "127.0.0.1:4532" (substitute whatever port is in wfview). fldigi can connect to wfview using this method if it is compiled with hamlib support. The port most programs expect is 4532.

- Enable Rigctl: Enables the built-in rigctl server.
- Port: This is the port that wfview's built-in rigctl server will listen on. The port most programs expect is 4532.
- Virtual Serial Port: Select a virtual serial port here through which other programs designed for a serial port can connect. See [here](#) for more details.
- TCP Server Port: This is a raw TCP/IP port for CI-V traffic and is compatible with N1MM+ and may work with RUMLogNG. To disable this feature, set it to zero.
- Waterfall Format: wfview can arrange the waterfall spectrum data so that the data are arranged as multiple separate chunks (serial connection style) or as single long messages (network radio style). This is useful if you are using a program on the Virtual Serial Port that expects serial connection style data, for example.
- Enable USB Controllers: This checkbox enables the use of external hardware USB controllers, such as the Icom RC-28, Contour Shuttle Express, Contour ShuttlePRO V2, and XBox. Please see [this page](#) for more information on how to setup and use external USB controllers.
- Setup USB Controller: Press here to open the USB Controller Setup page
- Reset Buttons: Press here to clear all button assignments
- Reset Commands: Press here to clear all button assignments and to reset your controller preferences. Use this option if the assigned button commands are malfunctioning. Once clicked, Save Settings and exit wfview.

See [this page](#) for more information on [USB Controllers](#).

See [this page of the manual](#) for more information about sharing radio control. Also see this page on [sharing audio](#) with other programs.

Using the RC-28, ShuttleXpress, and ShuttlePRO controllers

This page describes how to use USB-connected controllers with wfview to control your radio. Hardware controllers offer an immensely-improved user experience for tuning, changing modes, PTT, and more.

Supported Controllers

wfview supports the following USB controllers:

- [Icom RC-28](#)
- [Contour Shuttle Xpress aka Shuttle Express](#)
- [Contour ShuttlePRO V2](#)
- XBox USB controller TODO: Link for supported
- device [sunSDR Pro Ecoder Plus](#) (beta)
- [Quick Keys wireless](#)
- (beta) [Stream Deck](#):
 - Mini, MiniV2, Original, Original V2, Original MK2, XL, XLV2, Pedal, Plus

You can have a look at [the top portion of usbcontrollers.cpp](#) for the complete list, if you are curious about it.

Preliminary Setup

You will need to connect the USB device to your computer. It is important that there are no other programs using the device. Generally drivers are not necessary since these devices are "HID Compliant". Therefore, **do not** install any drivers or other software. wfview needs access to the USB device without any other programs or software using it at the same time.

Windows and macOS users should not need any preliminary setup. Please read the next section and skip the rest of this section.

Linux users will need to stop their desktop from using the wheel as a scroll wheel. This can be done per the following bash shell script taken from [this reddit post](#):

```
#!/bin/bash
for DEV in `xinput --list | grep -i ShuttlePro | cut -d"=" -f2 | awk {'print $1'}`
do
    xinput disable ${DEV}
done
```

Linux users will also need the following udev rule, placed in /etc/udev/rules.d, which gives the user permission to access the shuttle device:

Contents of `/etc/udev/rules.d/99-shuttle.rules` :

```
# for newer PRO model
ATTRS{name}=="Contour Design ShuttlePro" MODE="0644"
# for older PRO model
ATTRS{name}=="Contour Design ShuttlePRO v2" MODE="0644"
# for the Xpress model
ATTRS{name}=="Contour Design ShuttleXpress" MODE="0644"
SUBSYSTEMS=="usb", ATTRS{idVendor}=="0b33", ATTRS{idProduct}=="0020", MODE="0666"
```

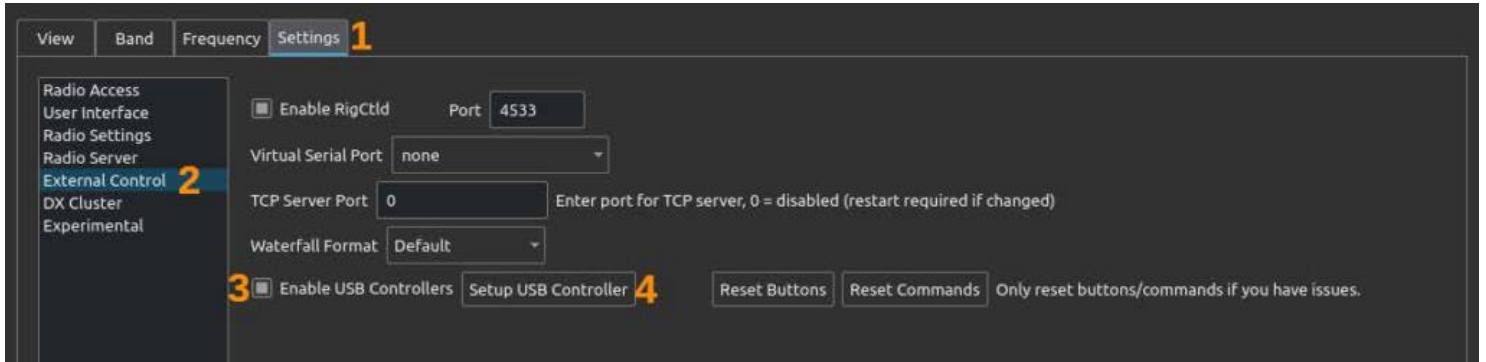
```
SUBSYSTEMS=="usb", ATTRS{idVendor}=="0b33", ATTRS{idProduct}=="0030", MODE="0666"
```

To activate the new rules, just pull the USB plug out and push it back in, that simple.

wfview Controller Setup

You will need wfview version 1.57 or greater. Currently the 1.60 release is recommended.

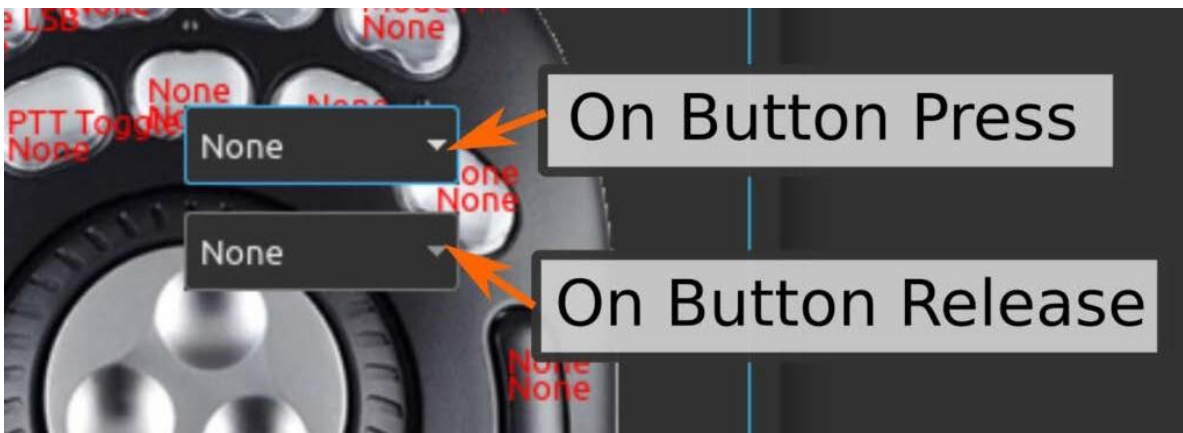
Connect the USB controller, and then, launch wfview. Click "Settings" followed by "External Control". Check the box "Enable USB Controllers". Then press "Setup USB Controller".



The controller setup window will appear. This window is different for each type of supported USB Controller. Here is how it looks for the Contour ShuttlePRO V2:



To assign a function to a button, right-click on the desired controller button. Two pop-up menus will appear. The top-most pop-up menu lets you select a function for when the selected button is pressed down. This is what most users will want for most functions. The bottom menu lets you select what will happen when the button is released. Generally, the bottom menu should be set to "None". Sometimes you will need to right-click in an empty space in order to remove the menu and reveal other buttons (we're working on a better way to do this).



To program a traditional PTT button, like a handheld microphone, where the button is held down while you are transmitting, select "PTT On" for the press (top) function and "PTT Off" for the release (bottom).

For a PTT "sticky toggle" function like the radio's chassis transmit button, select "PTT Toggle" for the top menu and select "None" for the release (bottom) menu

Press "Ok"

Important: Do not forget to press "Save Settings"!!

Keystrokes

wfview has keystrokes for almost all commonly-accessed controls one would need to have a QSO or browse the bands.

Not only is this convenient for rapidly manipulating the controls, it also may enable blind operators to use modern touch-screen radios if a computer (or even a Pi) is set up to run wfview in full-screen mode on boot.

Don't forget to check out our supported [USB Controllers](#), which offer even more hands-on controls.

Many users find that a stand-alone numeric keypad provides sufficient controls (frequency, mode, and tuning are available).

For users of the vi editor, we have added HJKL support, with modifiers for J and K being shift and control. H and L follow the selected tuning step.

And for those on the opposite end of the spectrum, we've also made sure that wfview works well on touchscreen devices. wfview looks great running full-screen on a 10" touch screen connected to a Raspberry Pi, for example.

NOTE: Mac users will need to substitute the Command key for the Control key mentioned here.

User Interface Navigation:

Key	Function
F1	Main View tab
F2	Band tab
F3	Frequency tab
F4	Settings tab

Numeric Keypad Operation:

Key	Function
/	Cycle through common modes
*	Enter frequency
+	Increase frequency by 1 KHz
-	Decrease frequency by 1 KHz
number and decimal	Used to enter frequency

Band Tab:

Key	Function
6	Jump to 6 meters
1	Jump to 10 meters
T	Jump to 12 meters
5	Jump to 15 meters
7	Jump to 17 meters
2	Jump to 20 meters
3	Jump to 30 meters
4	Jump to 40 meters
s	Jump to 60 meters
8	Jump to 80 meters
L	Jump to 160 meters
g	Jump to last general coverage frequency
A	Jump to VHF Air band
V	Jump to 2 meters
U	Jump to 70cm
W	Jump to Wideband FM band

Other:

Key	Function
F5	Activate mode LSB
F6	Activate mode USB
F7	Activate mode AM
F8	Activate mode CW
F9	Activate mode USB-D
F10	Activate mode FM
F11	Toggle full screen
F12	Send speech command to radio
F	Tell radio to speak current frequency
M	Tell radio to speak current mode
Control-T	Transmit
Control-R	Receive
Control-I	Enable/Disable the ATU
Control-U	Run ATU tuning cycle
H	Decrease frequency by selected tuning step
J	Same as -, with modifiers of shift and control too
K	Same as +, with modifiers of shift and control too
L	Increase frequency by selected tuning step
+	Increase frequency by 100 Hz
-	Decrease frequency by 100 Hz
Shift +	Increase frequency by 1KHz
Shift -	Decrease frequency by 1 KHz
Control +	Increase frequency by 1 KHz
Control -	Decrease frequency by 1 KHz
Page Up	Increase frequency by 1 MHz
Page Down	Decrease frequency by 1 MHz

Appearance Customization

The appearance of wfview can be customized fully using several methods.

1. Use the built-in color pickers in Settings, under User Interface in the “User-defined Color Editor”. In addition to assigning colors, you can also make a color fully transparent to hide the element.
2. Select an alternate waterfall theme using the “Theme” combo box at the waterfall bottom-right. Many of these themes will look dramatically different as you set the Top and Bot sliders to change the mapping of data range to color range. The color scales come from QCustomPlot and can be seen [here](#).
3. Toggle the “Use System Theme” checkbox to switch between your default desktop theme and the built-in stylesheet.
4. (Advanced users only) Specify an alternate stylesheet by manually editing the [Settings file](#) (or registry as it may be).

User-defined Colors

The built-in color picker looks like this:



Pressing any of the color buttons will let you edit the color for the respective attribute. Colors can also be manually input using the #AARRGGBB format, where "AA" is the alpha channel (ff =fully opaque, 00 =fully transparent). To help visualize the color changes, the "Pop-Out" button can be pressed to temporary pop-out the settings tab. This lets you see the changes on the View tab right away. Press pop-out again (which will say "Re-attach") to move the settings tab back into place. The settings can be saved using the "Preser' combo box. Press "Rename" to rename the setting, and press "Revert" for the default name and default colors for the selected preset. Don't forget to save the preset, otherwise the preset will revert the next time you start wfview.

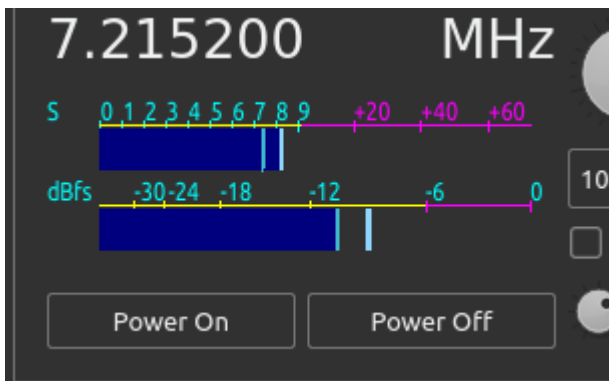
Stylesheet

wfview comes with a stylesheet built-in called [QDarkStyle](#). This stylesheet can either be edited (linux only) or, you may download your own copy of the stylesheet, edit the stylesheet, and specify the [stylesheet](#) using the Settings file.

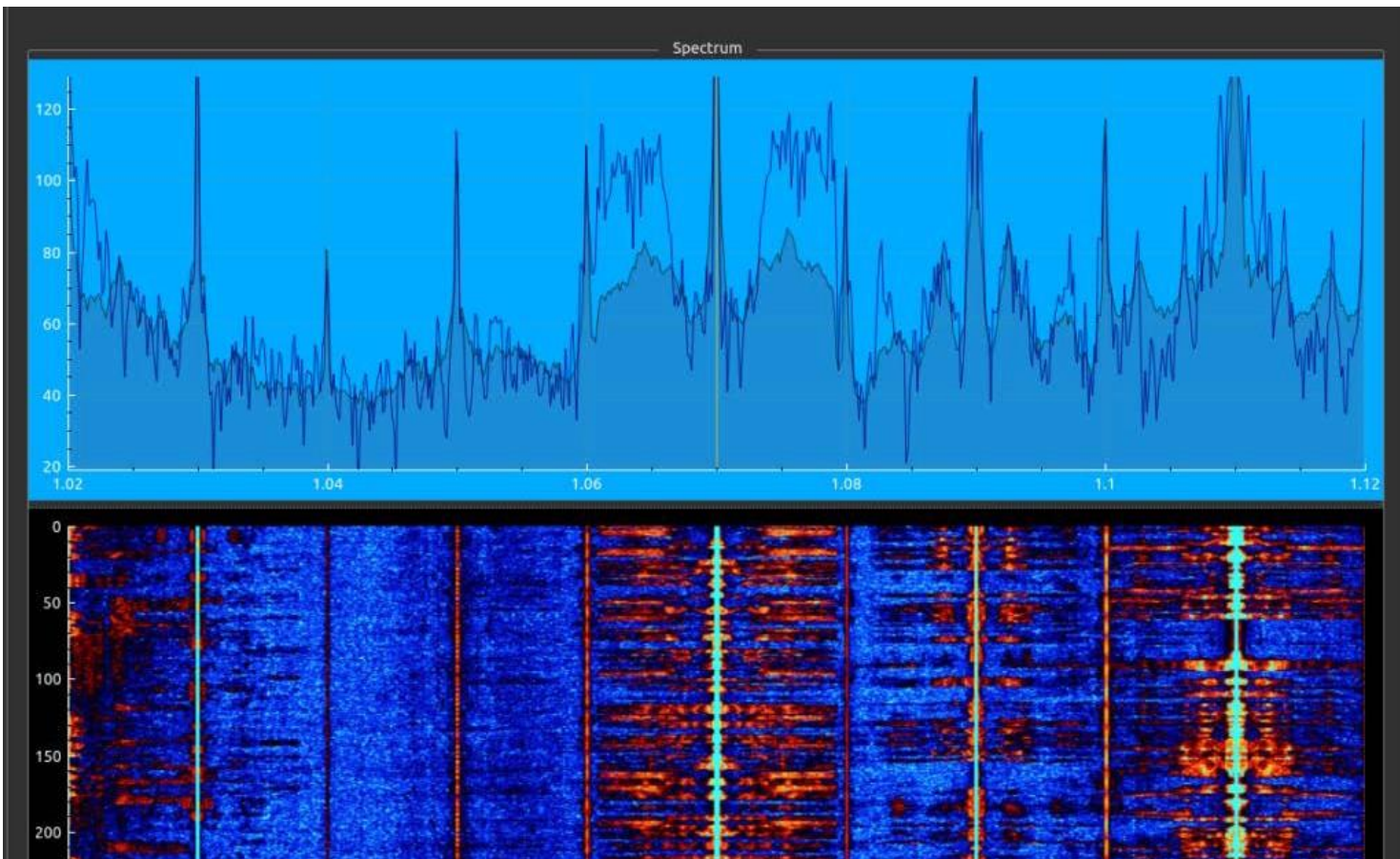
Gallery

Here are some interesting color examples.

The meters can be customized fully:

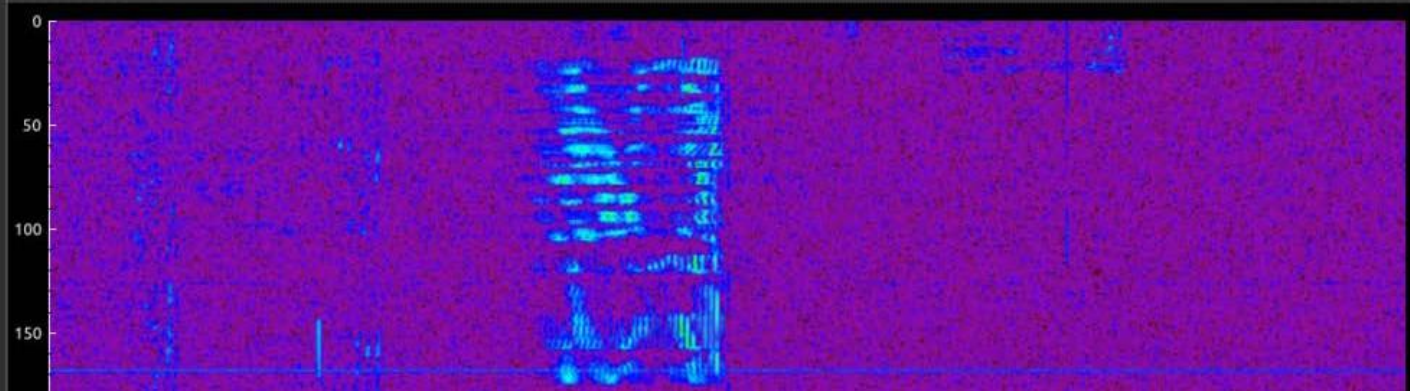


You can invert the colors on the spectrum (bright background), for example:



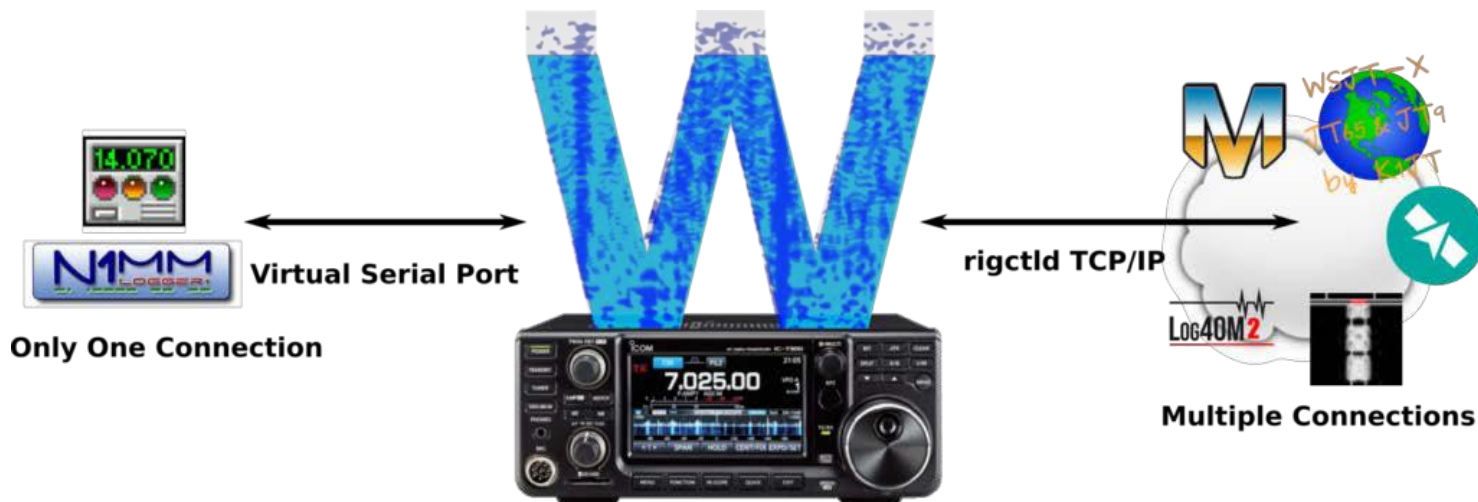
You can also change the tuning line color and the line and fill options:

Spectrum



Sharing Control Overview

wfview provides several methods of sharing the control of your radio with other programs. For example, you can connect to your radio with wfview, and then connect fldigi and MacLoggerDX to wfview. In all cases, connect wfview to the radio first. This is because wfview runs some commands that may not be compatible with other software programs, and also needs high-speed access to the waterfall data (which some programs would filter out). `rigctl`, which is built-in to wfview, is generally the best control option as it supports multiple programs at once and is generally simpler to configure. The virtual serial port option may be better for users running unusual commands that aren't in our implementation of `rigctl` yet. Don't forget to check [our guide on how to share the audio](#) with other programs, needed for network-access audio only of course.



The three principle methods of control are (click the links to learn more):

- built-in hamlib-compatible [rigctl server](#): recommended, supports multiple programs at once.
- [Virtual Serial Port interface](#): Emulates a genuine serial port in software. Only one program may use it at a time.
- Raw TCP/IP [socket](#): Works for a few programs such as N1MM+ and possibly RUMLogNG.

These options can be enabled on the Settings tab under the [External Control](#)

page. For 3rd party programs that need radio audio, such as fldigi and wsjt-x:

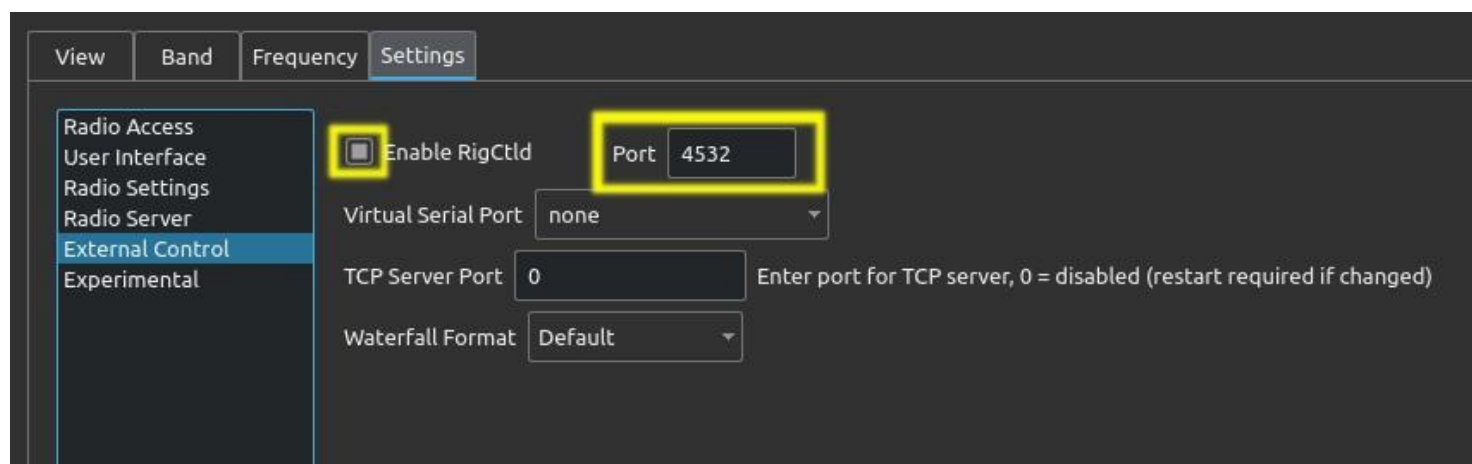
- If the computer is connected to the radio using a USB cord, simply use the audio device within the 3rd party program, as you would without wfview.
- If you are accessing the radio remotely using wfview over a network, please see [our page on Audio Configuration](#).

The following is a table of programs we have tested. For programs listing “rigctld”, you will need to select “Hamlib NET rigctl” (or similar) as the radio model. See the [rigctld page](#) for more information.

Software	Version	Platforms Tested	Working?	Connector	Comments
WSJT-X	2.3.1	Windows, Mac, Linux	YES	rigctld	
Log4OM	2.11	Windows	YES	Omni-Rig	
Minos	2.4.1	MacOS, Windows	PARTIAL	rigctld	Not Linux/MacOS
fldigi and flrig	Current	Linux	YES	rigctld (preferred) and Pseudo-Term	
MacLoggerDX	6.35b5	MacOS	YES	rigctlf (preferred)	Requires com.dogparksoftware.MacLoggerDX dp_virtual_serial_port configuration option
QLog	Current	Linux	YES	rigctld	Use model "Hamlib NET rigctl". Don't forget Equipment -> Connect Rig

Hamlib rigctld Emulation

wfview has the ability to emulate a subset of the Hamlib rigctld protocol which allows a number of applications to connect for rig control purposes. When supported, this is now the recommended/preferred method of connecting software to wfview (rather than virtual serial port/pseudo tty) as this utilizes a caching mechanism which avoids ‘spamming’ the rig with large numbers of duplicate CI-V requests.

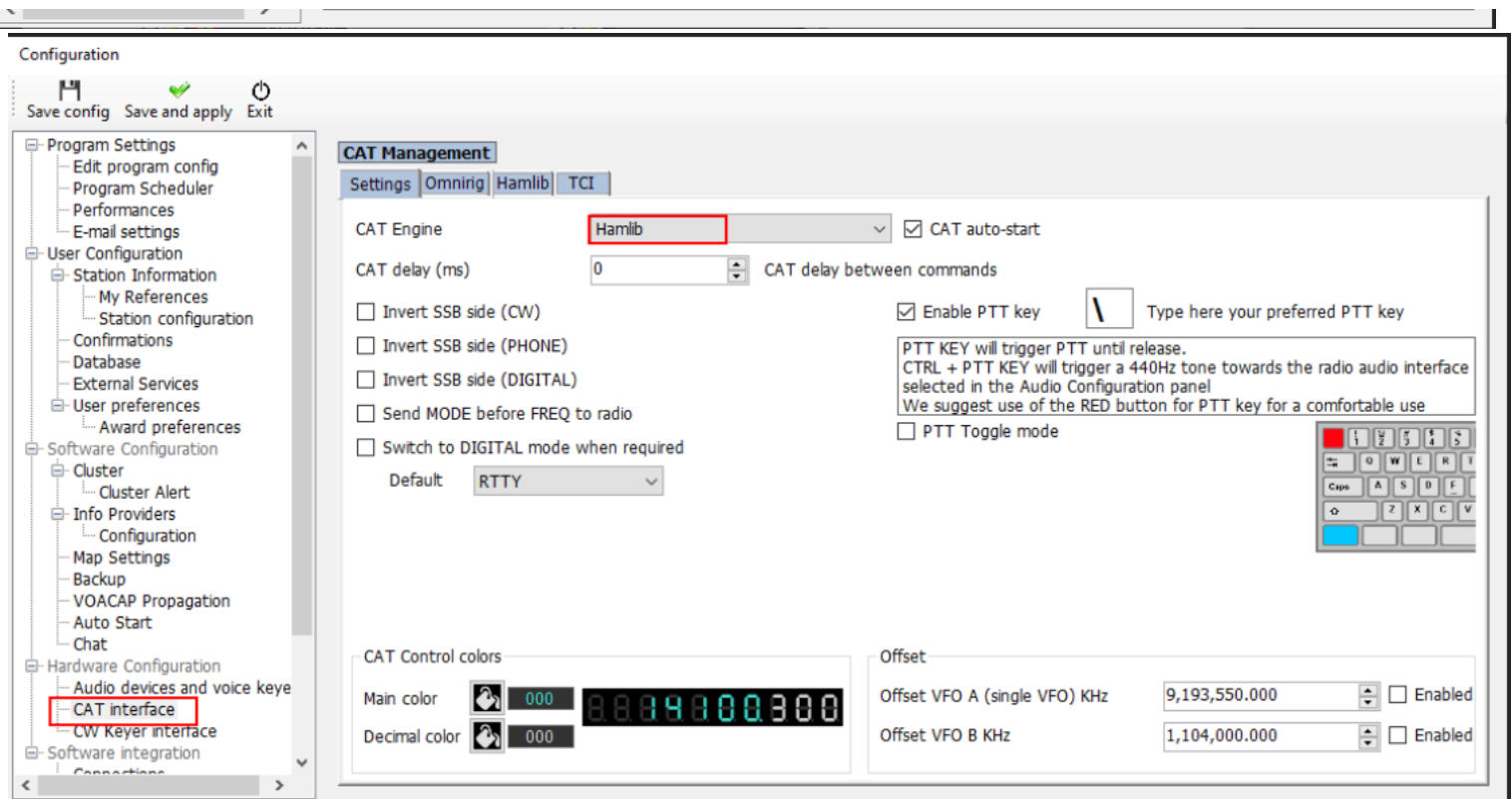


To enable rigctld emulation, simply put a check next to “Enable RigCtld” within the wfview Settings tab, on the “External Control” page. The default port is set to 4533 so as not to conflict with any existing rigctld that is running. If you are confident that no other rigctld is running, you can set this to 4532 which will be the default port on most software. The port number must be greater than 1024. The port can be changed without restarting wfview, simply uncheck/check Enable RigCtld to restart the rigctld emulation.

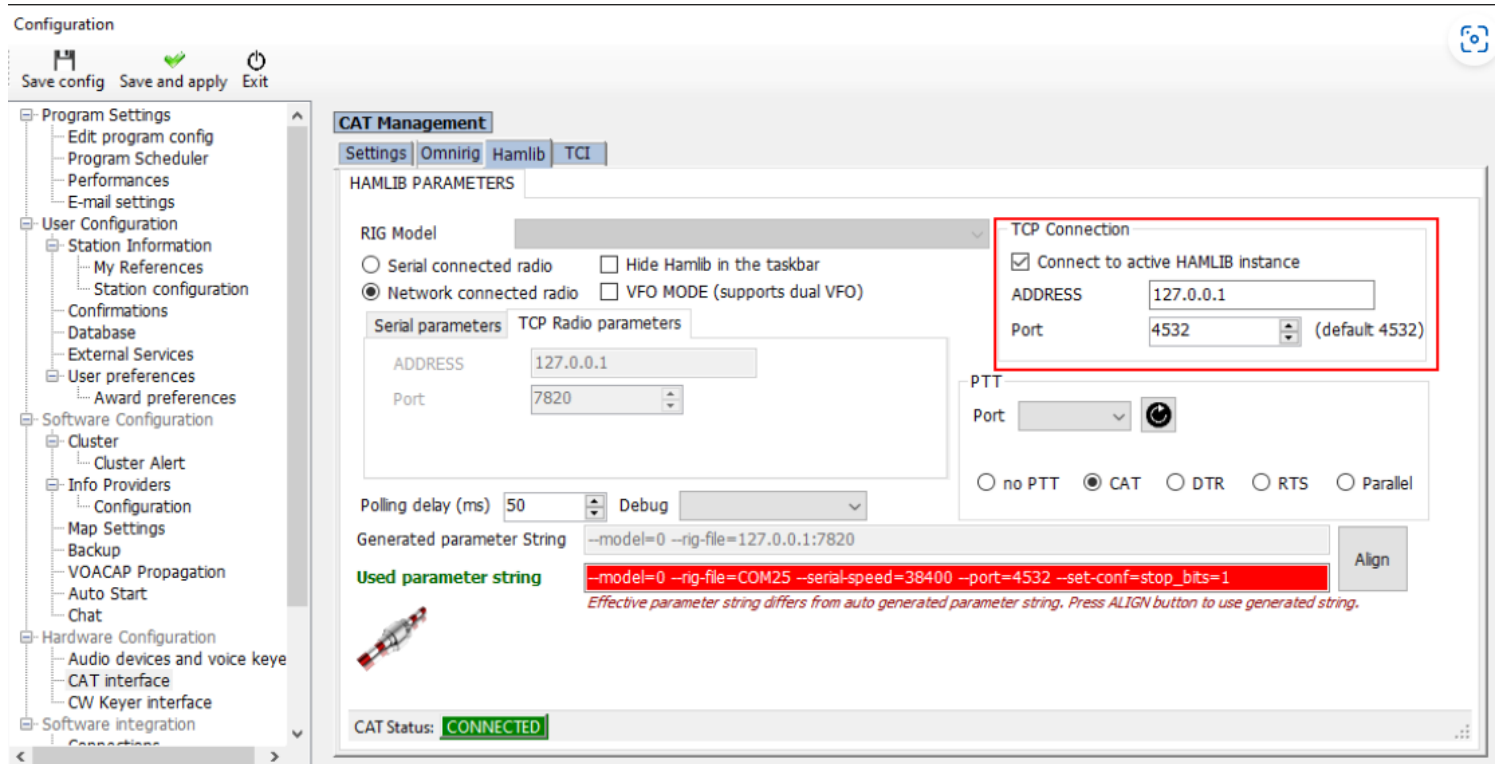
No other configuration of wfview is required, below are the required settings for various software that has been tested so far:

Log4OM 2 (NextGen) configuration

Firstly you must be running the latest version of Log4OM 2 (currently 2.17.0.0). Within “Settings/Program Configuration/CAT Interface” select Hamlib as the CAT engine.



Once done, select the Hamlib tab and enter the address and port of your wfview computer (if the same machine then 127.0.0.1). If you haven't changed the port in wfview to 4532, you will need to enter 4533 in the Port box below.



Once the above is saved, I found that I needed to restart Log4OM before it was able to connect to the wfview rigctld.

WSJT-X

Setup of WSJT-X is very straight-forward (this should be the same for the majority of software that is based on WSJT-X like JTDX and JS8-Call). Simply select the Rig: of "Hamlib NET rigctl" and enter 127.0.0.1:<port> into Network Server. Pressing Test CAT should result in the button turning green, which means it is talking to rigctld. Various other settings are optional but I would recommend setting PTT method to CAT, Mode to Data/Pkt and Split to either Fake It or Rig (rig based split support is still experimental in wfview though).

The screenshot shows the 'Settings' dialog box for WSJT-X, with the 'Radio' tab selected. The 'Rig' dropdown menu is set to 'Hamlib NET rigctl'. The 'Network Server' field is set to '127.0.0.1:4532'. The 'Poll Interval' is set to '1 s'. The 'PTT Method' is set to 'CAT' and the 'Port' is 'COM14'. The 'Transmit Audio Source' is set to 'Front/Mic'. The 'Mode' is set to 'Data/Pkt'. The 'Split Operation' is set to 'Rig'. There are two buttons at the bottom: 'Test CAT' (green) and 'Test PTT' (blue). The 'OK' and 'Cancel' buttons are at the bottom right.

Settings

General Radio Audio Tx Macros Reporting Frequencies Colors Advanced

Rig: Hamlib NET rigctl Poll Interval: 1 s

CAT Control

Network Server: 127.0.0.1:4532

Serial Port Parameters

Baud Rate: 115200

Data Bits

Default Seven Eight

Stop Bits

Default One Two

Handshake

Default None
 XON/XOFF Hardware

Force Control Lines

DTR: RTS:

PTT Method

VOX DTR
 CAT RTS

Port: COM14

Transmit Audio Source

Rear/Data Front/Mic

Mode

None USB Data/Pkt

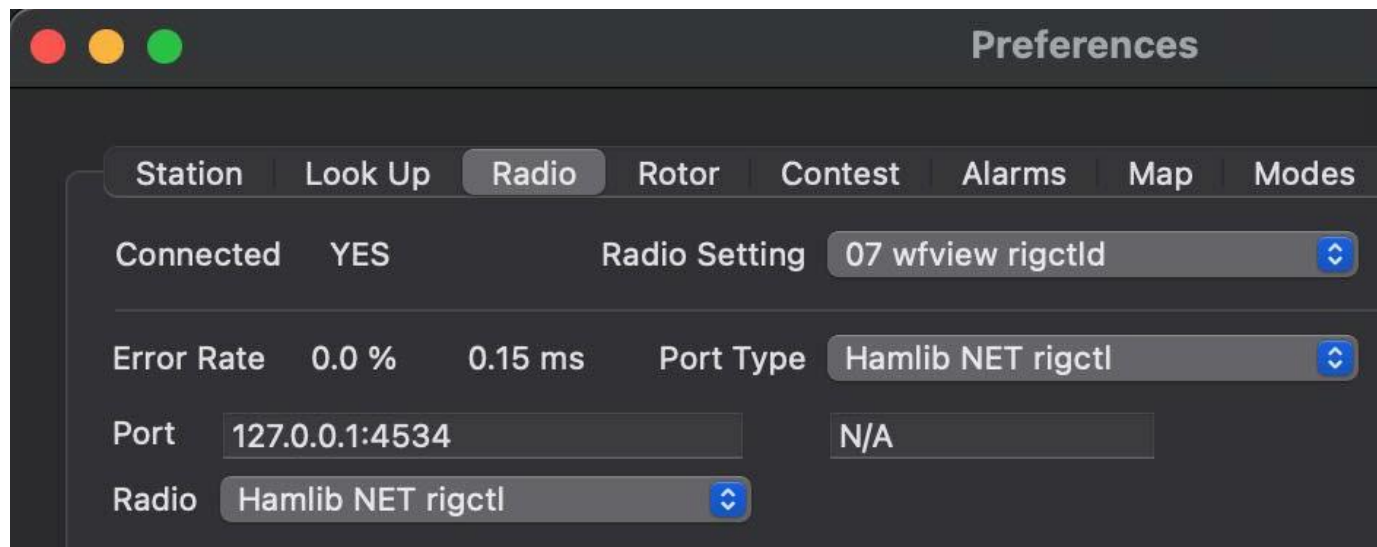
Split Operation

None Rig Fake It

Test CAT Test PTT

OK Cancel

MacLoggerDX

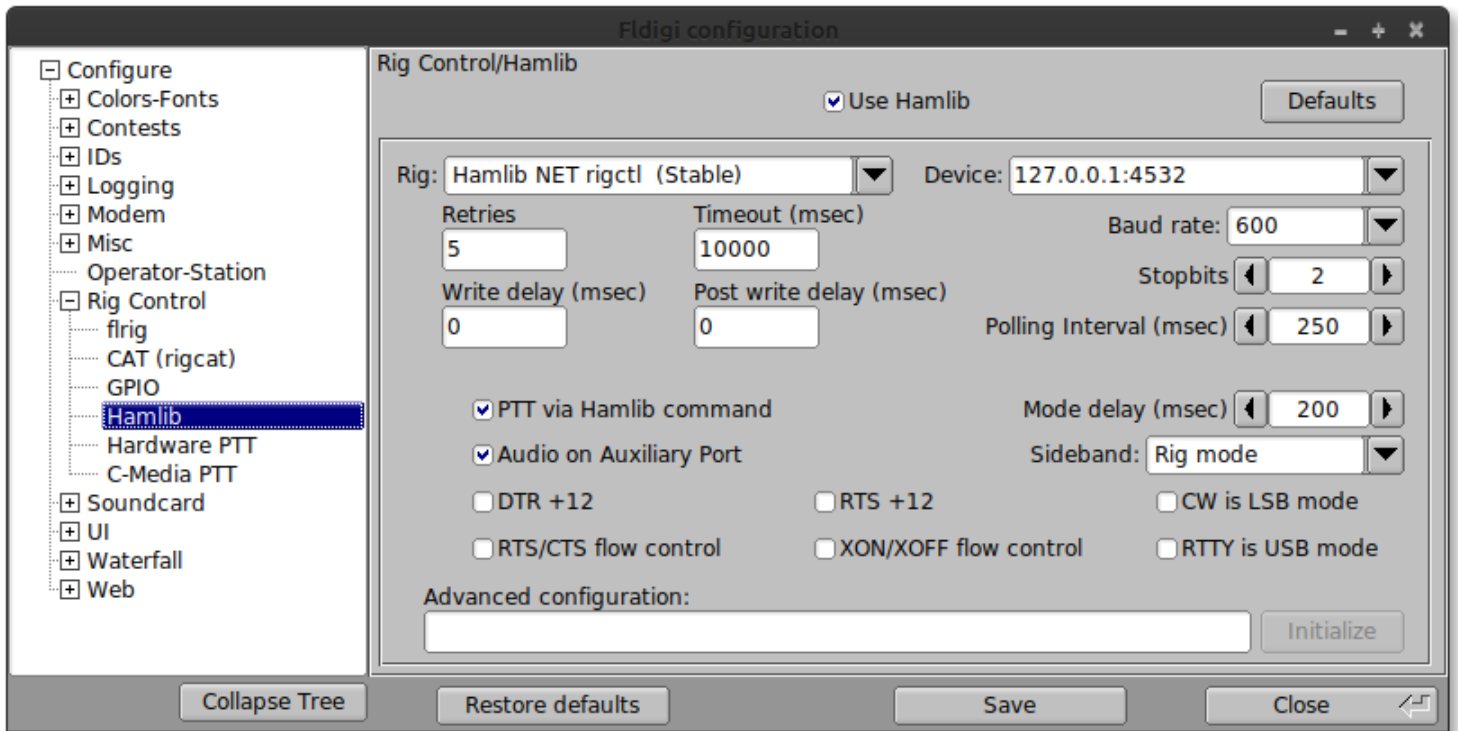


The Radio Setting name is user-defined, set it as you wish. For port type, select “Hamlib NET rigctl”. For the port, use “127.0.0.1:4534”, replacing “4534” with your port as specified in wfview.

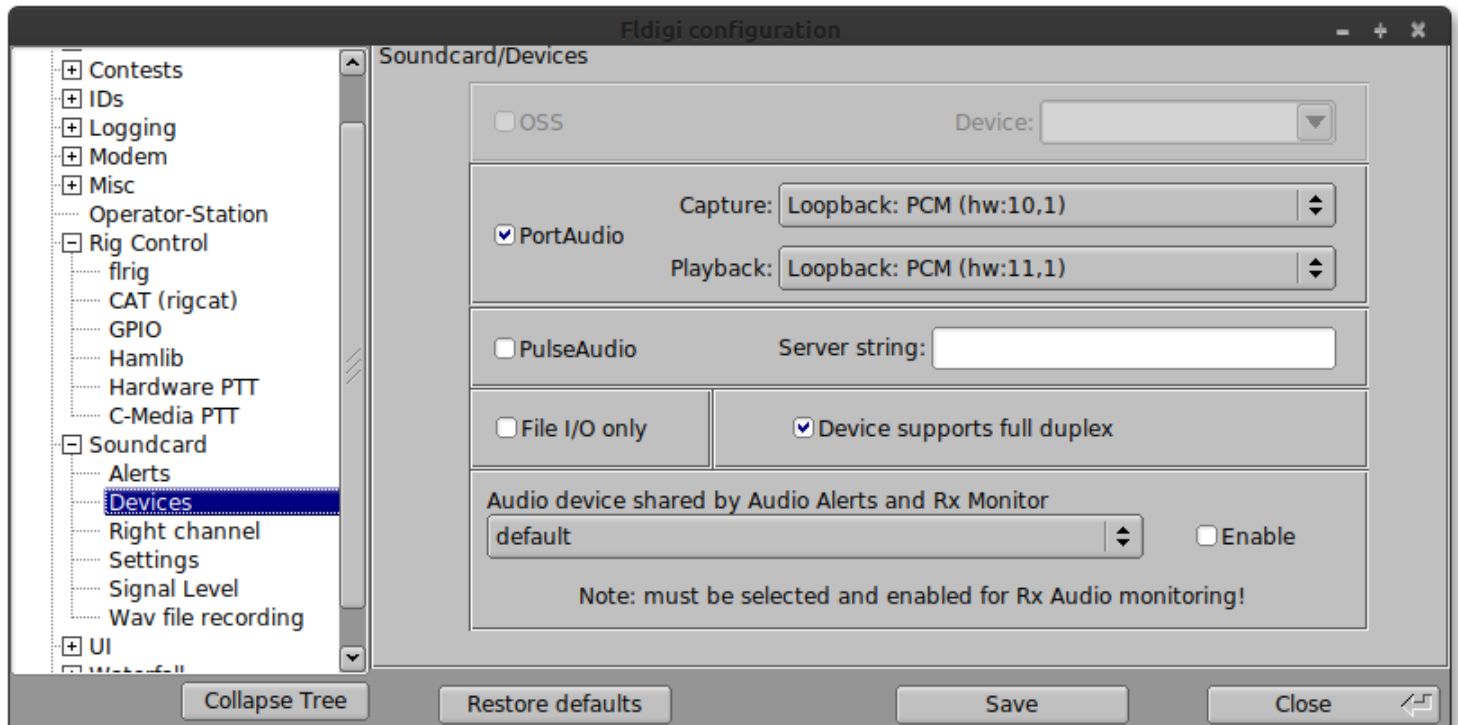
fldigi

Fldigi can easily connect to wfview using wfview's built-in rigctl emulation. Note that fldigi needs to have been compiled with Hamlib support, which is commonly done.

To configure the radio control, turn off or disable any other rig control configured (flrig, RigCAT, etc), and then configure the Hamlib interface as shown. Be sure to check the "Use Hamlib" box:



For audio, enable a loopback audio device per our [Audio Configuration guide](#), and then set as shown (this will vary some with each platform, the screenshot is from Linux):



Virtual Serial Ports

The built-in virtual serial port option of wfview lets a second program control the radio over a pseudo-terminal device. Using this method, many programs will work “out of the box” without knowledge that they are connected to a virtual serial device. Some programs will need additional configuration. Under Windows, a virtual serial port program is required to make the “loopback” connection between the wfview virtual serial port and the second program. Linux and Mac create genuine pseudo-terminal devices.

When possible (supported), it is generally better to use [the built-in rigctld](#) server. Jump to: [Windows](#) [Linux](#) [Mac](#)

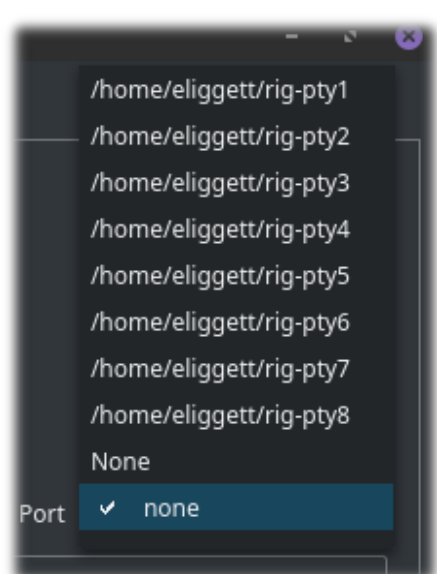
Windows

Virtual Serial Ports

If you wish to connect other software such as digital mode interfaces or logging programs to wfview, this can be accomplished by creating a serial port pair. There are a number of Windows programs that can achieve this, here are a small selection:

- Eterlogic VSPE <http://www.eterlogic.com/Products.VSPE.html>
- Eltima Virtual Serial Port Driver <https://www.eltima.com/products/vspdxp/>
- com0com This is a FOSS driver but a Windows signed binary can be downloaded from <https://pete.akeo.ie/2011/07/com0com-signed-drivers.html>

Each solution operates in (broadly) the same way, you create a “pair” of virtual serial ports which are interconnected in software so any data sent to one port, automatically appears at the other. wfview is therefore configured with one “end” of the virtual serial port pair in the “Virtual Serial Port” combobox within the settings page. Your digital mode software/logging software is configured with the other “end” of the pair. Your software can then communicate with the rig as if it was directly connected.



Linux and Mac

On Linux and macOS, wfview provides a pseudo-terminal device out of the box, which may be used as a normal serial port by other applications. Previous versions created the device at `/tmp/rig` but newer releases of wfview allow the user to specify the device under the Settings tab, using the Virtual Serial Port combo menu:

wfview's pseudo-terminal selection

Users can either select one of the pre-defined names, or type within the box to define a custom name (for example, `/home/user/ic-7300`). Don't forget to press Save Settings after changing this box. To use the pseudo-terminal in a "client" program (such as fldigi, Ham Radio Deluxe, etc), you will need to either enter the device into the settings for the program, or, for programs that don't allow free-form entry, close the program and edit the program's settings files, manually specifying the same pseudo-terminal port name.

Audio Configuration

To be able to use wfview with digital modes like FT8, PSK etc. It is necessary to create "virtual audio" connections. These are effectively loopback sound cards that the transmit and receive audio are sent via wfview and allow your digital mode software (WSJT-X, Fldigi etc) to connect to it.

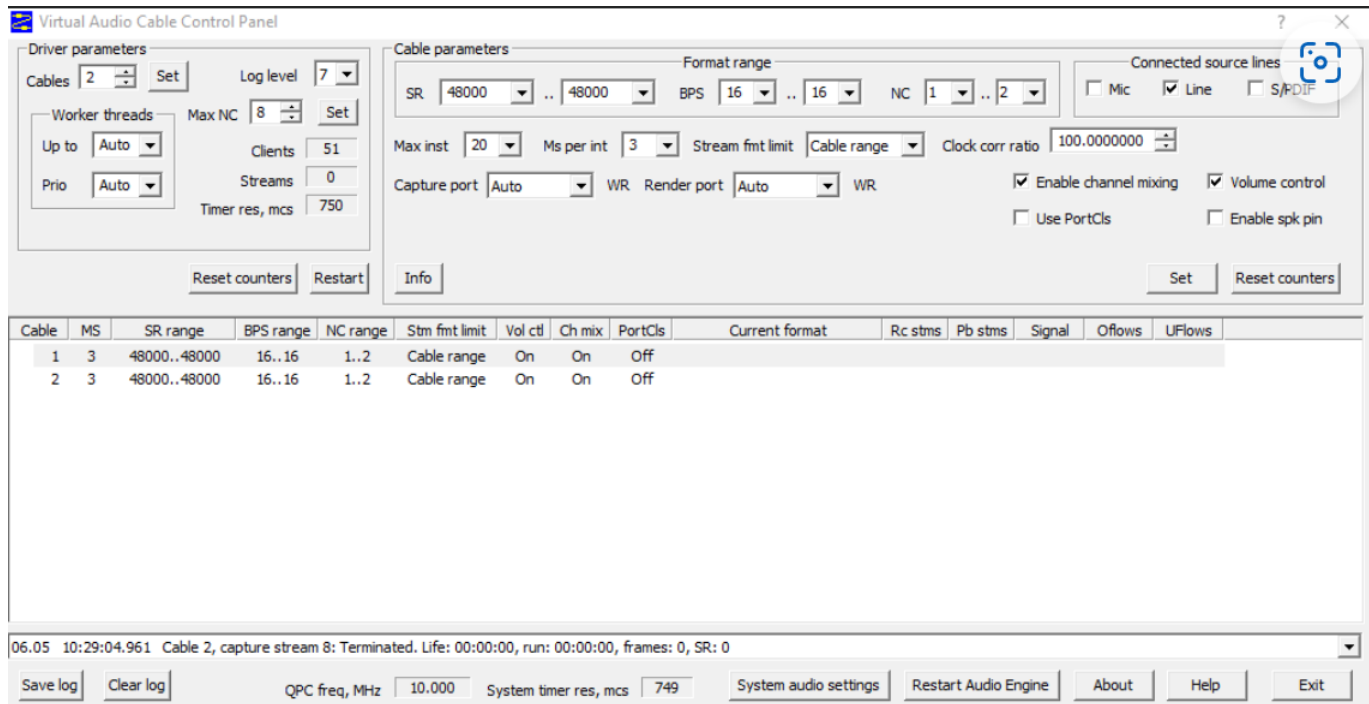
We have tested configurations in both Windows and Linux but similar functionality can be achieved in MacOS using a tool called SoundFlower <https://github.com/mattingalls/Soundflower>

The following are only required for users who are connected to their rig via LAN/WiFi. For USB connected rigs, you can simply use the USB audio devices provided by your rig.

Windows Virtual Audio Cable

For Microsoft Windows users, we have tested Virtual Audio Cable <https://vac.muzychenko.net/en/index.htm> which must be registered and costs \$30 for non-commercial use. VB-Cable is a free alternative that can be downloaded from <https://vb-audio.com/Cable/index.htm> it should work and we will hopefully get a chance to test it at some point.

In VAC You will need to create two “Cables” as per the image below:



The key settings are the Sample Rate (SR) which should be 48000 and BPS Range which should be 16bit. I only enabled Line in the “Connected Source Lines”.

Once you have this running, it is simply a case of selecting the Audio Output and Input devices in wfview and your other software. In wfview, Audio Output should be “Line 1 (Virtual Audio Cable)” and Audio Input “Line 2 (Virtual Audio Cable)”. In your software, for example WSJT-X, under the Audio tab, Input wants to be “Line 1 (Virtual Audio Cable)” and Output “Line 2 (Virtual Audio Cable)”. Basically, the output of wfview connects to the input of WSJT-X and the output of WSJT-X connects to the input of wfview.

Loopback Audio Devices in Linux

With Linux, no additional software is required however it will need some configuration completing. The following was completed on Ubuntu 18.04 but should be similar for any Debian derived version of Linux.

The basic concept is similar to Windows, you create two loopback devices (virtual audio cables).

If it doesn't exist already, create a file called `/etc/rc.local` which will contain various commands that must be run at startup.

```
sudo nano /etc/rc.local
```

Add the following into the file (if the file already exists then just add the sudo line.)

```
#!/bin/sh
#
sudo modprobe snd-aloop

exit 0
```

Newer versions of Ubuntu do not run `rc.local` by default so this must be enabled.

```
sudo systemctl enable rc-local.service
```

If you receive an error about the unit files having no `[Install]` section, you will need to do the following. If you don't receive an error, skip to "**Configuration for snd-aloop device**"

```
sudo nano /etc/systemd/system/rc-local.service
```

Then add the following to this file:

```
[Unit]
Description=/etc/rc.local Compatibility
ConditionPathExists=/etc/rc.local

[Service] Type=forking
ExecStart=/etc/rc.local start
TimeoutSec=0
StandardOutput=tty
RemainAfterExit=yes
SysVStartPriority=99

[Install]
WantedBy=multi-user.target
```

You will also need to set the executable permission on the `rc.local` file and enable the service

```
sudo chmod +x /etc/rc.local
sudo systemctl enable rc-local.service
```

If all worked, you will not receive an error on either of these commands!

Configuration for snd-aloop device

The snd-aloop device also needs configuration, this is done in /etc/modprobe.d/virtual-sound.conf (the file can actually have any name as long as it has the .conf extension)

```
sudo nano /etc/modprobe.d/virtual-sound.conf
```

Add the following into this file. This creates 2 loopback devices (10 and 11) You can add more devices here if needed:

```
alias snd-card-0 snd-aloop
options snd-aloop enable=1,1 index=10,11
```

Now Reboot your computer!

Once rebooted, you should be able to run wfview and see a number of new device.

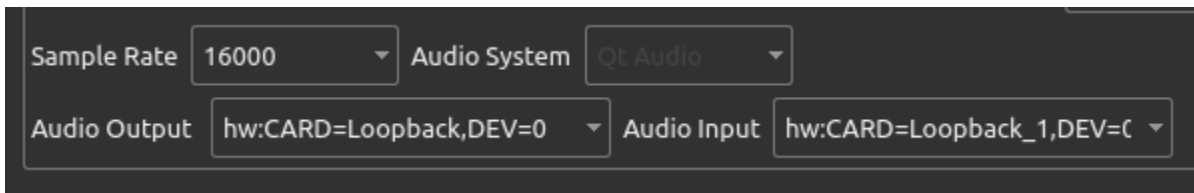
In the Audio Output combobox, select: Be careful, the wfview side of both loopback devices is “DEV=0”

```
hw:CARD=Loopback,DEV=0
```

Then in the Audio Input combobox, select:

```
hw:CARD=Loopback_1,DEV=0
```

Here is a screenshot of wfview properly configured for loopback audio:



For most applications, select the following for Input: Be careful, the application side has “DEV=1”.

```
hw:CARD=Loopback,DEV=1
```

For Output:

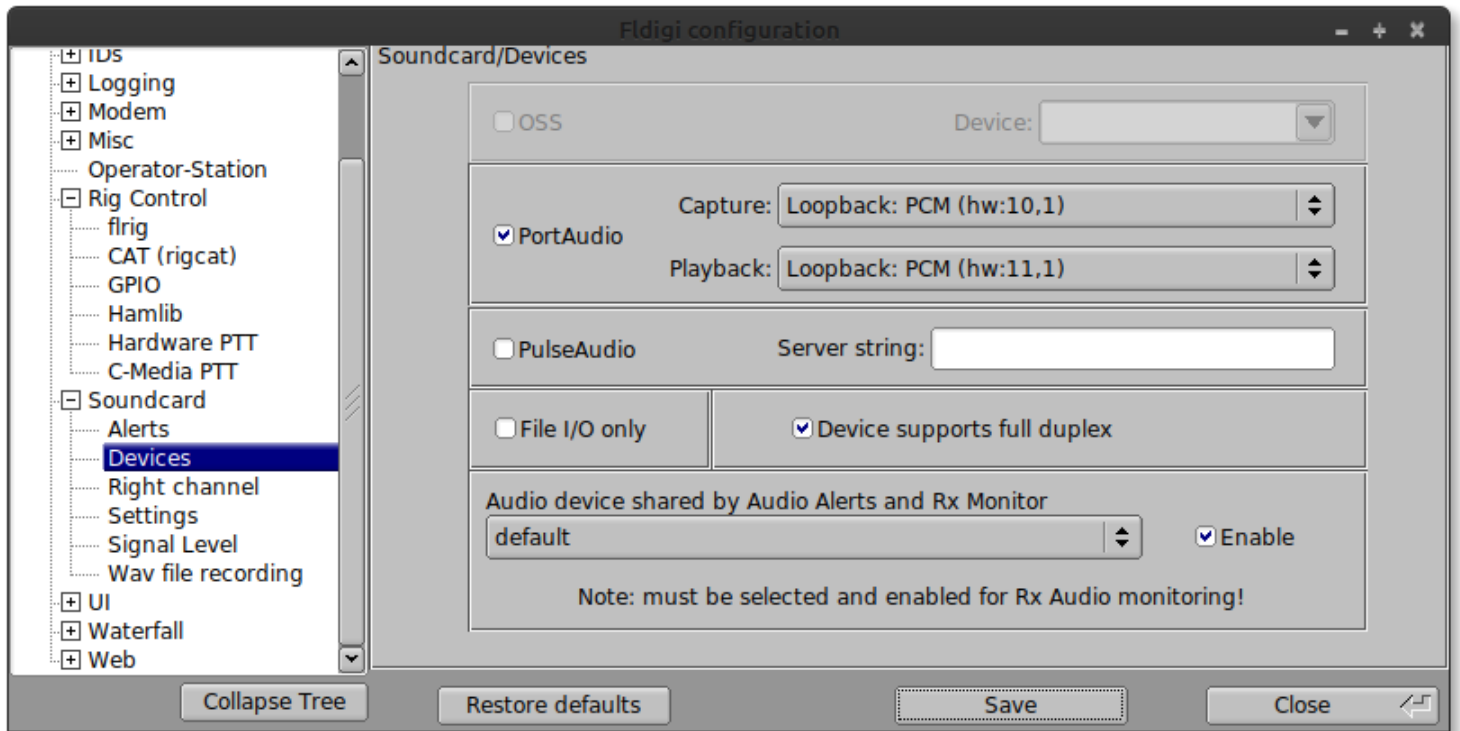
```
hw:CARD=Loopback_1,DEV=1
```

The WSJT-X Audio tab should look similar to this:

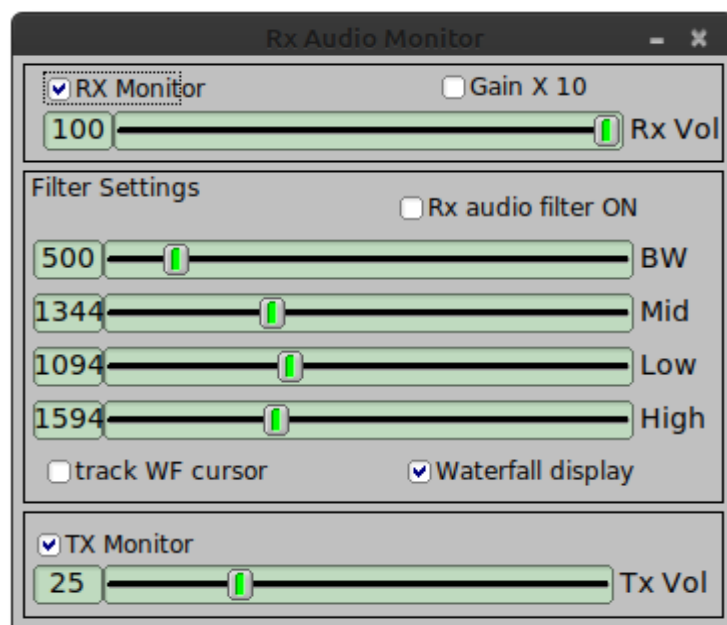


fldigi Audio configuration

After following the above guide for creating a loopback device, configure fldigi's audio as shown (there will be some differences depending upon the platform). If you enable the "RX Monitor" feature, then you can select your speakers and hear the receive audio from the radio *and* keep the audio going into fldigi for decoding! Be sure to select the correct instance of each loopback device. The ",1" part means you are on the second side of each loopback device (DEV=1 basically). Note that the capture device is connected to the wfview Audio Output device, and the Playback device is connected to the wfview Audio Input device.



If you'd like to hear the audio in the loopback, use the RX Audio Monitor inside fldigi. It can be accessed under the View menu and is called "Rx Audio Dialog". From there, you may enable the "RX Monitor" and "TX Monitor" (TX Monitor is available in versions 4.12.24 and greater). The TX Monitor is a great way to hear your CW sidetone if you do CW in fldigi.



Remote Operation (server)

wfview has a built-in radio server. This guide will help you configure the built-in server, which you can use to remotely access radios which lack a built-in server.

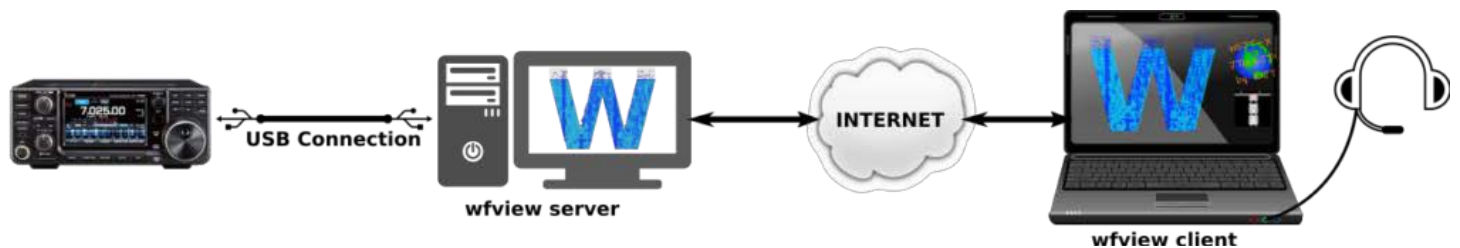
Scope

This article **only** applies to USB and Serial radios. For radios with built-in ethernet or wifi, simply connect wfview to the radio's ip address. See the [Getting Started](#) chapter. So, to be clear, this article and the entire "Radio Server" page are not needed if your radio already has a network connection.

The most commonly used radio with this setup is the IC-7300, but many older rigs will work great with the wfview built-in server, which handles getting full-duplex audio and control over the network.

wfview's built-in server handles the radio's controls as well as full-duplex audio. Simply run one copy of wfview on a server computer, which can be anything from a humble Raspberry Pi to a full desktop PC, and run a second copy of wfview on your remote computer or laptop. The controls are very responsive and audio latency is generally under 75ms (LAN) or under 200ms (WAN). The normal graphical wfview program you're familiar with has the server capability already.

Concept



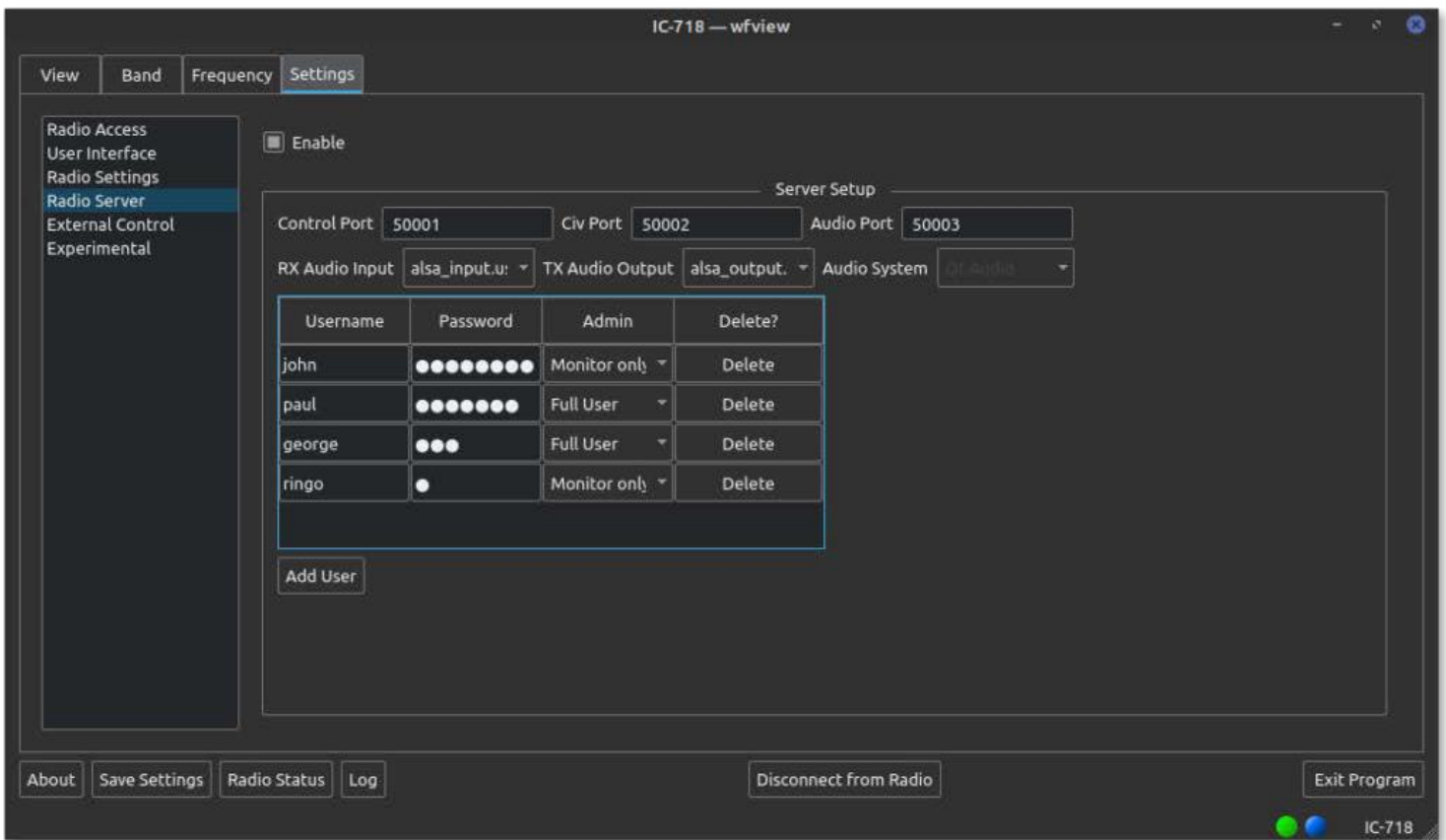
To remotely operate a radio, you will need to run one copy of wfview on a computer connected to the radio with a USB cable (or other appropriate cables). It can be a small, single-board computer like a Raspberry Pi, or a normal desktop computer. Currently, a desktop environment is required, but a command-line server version will eventually be made available. Once connected and configured, client computers running wfview can connect over the network (internet or LAN) to the computer hosting the radio.

Server Setup

For the server computer, begin by installing wfview. Connect the radio, and then launch wfview.

Follow the [Getting Started guide](#) for serial port radios. Make sure, on the radio, that CI-V Transceive is turned on.

Once wfview is running and properly controlling the rig, head over to the Settings tab of wfview, and click on the "Radio Server" page.



The standard ports, as shown in the screenshot, are 50001, 50002, and 50003. You should use these ports unless you have a reason to use something else.

Select the radio's receive audio device under "RX Audio Input", and select the radio's transmit audio device under "TX Audio Output". If these drop-down menus are grayed out, it means that wfview has not connected to the radio yet; head over to Radio Access and make sure the Serial port has been set up correctly and then press "Connect to Radio".

Under Linux, these two names (or similar names) should work for the server audio on the IC-7300:

- Audio Input: `alsa_input.usb-Burr-Brown_from_TI_USB_Audio_CODEC-00.analog-stereo`
- Audio Output: `alsa_output.usb-Burr-Brown_from_TI_USB_Audio_CODEC-00.analog-stereo`

Generally speaking, on Windows, the audio device will say "USB Audio CODEC" in the name. Remember, you are looking for a USB device. If you see many USB Audio CODEC devices, try unplugging other USB audio devices and any other radios, and then re-launch wfview and see if you see fewer, it may help.

The Audio System should generally be set to "Qt Audio" unless there are problems with this audio system. Note that you cannot change the audio system once a client is connected; therefore, it is necessary to make sure all clients are disconnected prior to changing the audio system.

Next, define a username and password for a "Full User" account. As of wfview 1.5, user privileges are not enforced, but it is something we may do in the near future. Since the passwords are not well-encrypted, do not use a sensitive password.

If you are satisfied with the settings, press "Save Settings" and then re-check the "Enable" checkbox to turn on the server. You should see "udp.server: starting udp server" in the [Log window](#). You should also see a message about the Audio Input and Audio Output devices; make sure they match your settings.

At this point, server setup is complete.

Client Setup

setting up a second computer to connect to your wfview server is simple. Launch wfview on the second computer, and follow the [Getting Started guide](#). For the radio connection type, specify “Network”. Supply the ip address of your wfview server computer for the Hostname, along with the same control port (50001 by default), username, and password.

For the TX and RX Codec, “Opus 1ch” is the recommended choice, as it uses about 4 to 10 times less bandwidth than the 16-bit PCM formats. Note that Opus will only work if you are running a wfview server; it is a special benefit of this software. For VPN users, you will likely need to use an 8-bit format such as “8-Bit uLaw” due to typical VPN packet mangling. The uLaw codec uses logarithmic bit spacing and provides surprisingly good quality.

The audio devices should be chosen for your client computer, typically a headset or speakers. Use the Qt Audio system unless you have problems with it. If you need to loop audio into a second program such as wsjt-x or fldigi, read our [audio configuration](#) page.

Most of the details for these controls and options are covered in the [Settings](#) manual page.

Radio Settings

Your radio needs to be told to source audio from the USB or ACC connector, as applicable to your setup, so head over to the Radio Settings page of the Settings Tab, and select the audio interface that the radio should use for transmit audio. IC-7300 users should select “USB”. Older radios, lacking a USB port, will likely be connected to the computer using an ACC adapter. Some older radios do not have a command for changing the input, so the input must be changed using buttons on the radio.

See [this page](#) for details.

Router Setup

To access wfview from outside your home network, you will need to port forward UDP traffic on the three ports mentioned above. Important: DO NOT map the ports to any other port numbers. They must be the same port on the router and the computer running wfview server. This is because the port numbers are a part of the communication protocol itself.

Troubleshooting

Running a server is an inherently tricky task, and it is even more difficult when streaming audio is involved! You will need to be familiar with concepts such as TCP/IP and be comfortable with topics like finding your ip address.

Audio: There are three places to look: Radio audio settings (input selection, AF gain, Mic gain, etc), Server (TX and RX audio source/sink selection), and Client (mic and speaker connection). The best strategy is to begin with a simple

USB headset on the client. You can enable TxAudio as a type of secondary meter and you should be able to see the raw headset audio on the meter even when receiving. If you don't see it, adjust the gain for the headset using your operating system's audio settings control panel. On the server, make sure the right audio devices are not only selected in the GUI but also being ultimately used by the server. The server audio modules will load upon the first client connection, and you should see this in the log (press “Log”). Check what device is being used. Radio devices are generally easy to spot.

Connecting: First, try connecting from one computer to another within your own LAN. That is, not using any external IP address. Make sure you can ping from one computer to the other. Verify the IP address of the server. Check the log on the server (press the Log button). See if you can see the remote login activity.

Inexpensive Remote Server

This page tells how to set up a simple remote server so that radios without network access can be used remotely. A popular radio for this purpose is the IC-7300, but other radios from Icom will work just as well.

For this how-to, I will be using the \$35 [Inovato Quadra](#). A Raspberry Pi 3, Orange Pi, or any modern SBC (“single-board computer”) should work fine as well. The Quadra comes with built-in storage and a case, so you do not need to buy those things. The included power supply is not appropriate for ham radio as it makes hash all throughout the HF bands (ask me how I know...). However, in testing, I observed that the maximum current on the 5V supply was about 812mA, briefly during booting, and that at idle the current is only 235mA (with the desktop open and wfview running). There are many low-noise supplies which can support this SBC at these low currents. A 7805 regulator may even be sufficient (with heatsink and capacitors of course).



You should start by reading our [server setup page](#) to get the idea of what we’re about to do. The process is very similar, but we will be using a small and dedicated computer instead of a normal desktop. You will need to be comfortable editing files and running commands in the terminal. Familiarity with network setup is a must.

You will need cables to connect your radio using the USB port. Many radios, like the IC-7300, have a built-in USB port, and for that radio you just need a USB cable. For older radios, you will need something like a RigBlaster or XGG Comms adapter. Adapters with transformer isolation are recommended.

Operating System Configuration

The Quadra comes with armbian, which is a Linux operating system based on Debian Bullseye. The Raspberry Pi also runs a debian-based distribution. Any Debian-like system should be fine.

Start with the usual. Change the default password (change it for *both* the root user and the quadra user — make sure to do both!) and make sure the network access is solid. I used ethernet for my setup as it is definitely ideal for streaming service such as this. Also, on some platforms the WiFi over USB causes issues with how we stream over USB. Therefore, use Ethernet if at all possible. Optionally run a system update and reboot. ssh is enabled out of the box on port 22.

With armbian, I had to remove `brltty` to stop it from stealing the serial port of the radio. If you don’t do this, you won’t see the serial CI-V device!

```
sudo apt-get remove brltty
```

Reboot once that is complete. I also added x11vnc so that I could easily remotely view the display. You can do this entire tutorial using ssh and ssh with x11vnc if you wish, or you can sit at the SBC and do it that way. Either way though, this utility is quite handy. You can ssh to the box and run x11vnc, and then connect over VNC to see the local display. If the machine is stuck at the login manager, then you need to run it as sudo with the “`-auth guess`”

argument. Install `x11vnc` like this:

```
sudo apt-get install x11vnc
```

I configured the machine to automatically log in the primary user. If you switch later to the non-graphical wfserver version, you won't need to do this, but for now it's quite simple and effective.

I followed [this guide](#) to configure the machine for automatic login. Here's the change to make:

Add the following two lines to `/etc/lightdm/lightdm.conf.d/11-ambian.conf` :

```
autologin-user=quadra
autologin-user-timeout=0
```

Installing wfview

Installing wfview is simple. You can use [our installer script](#) to automate the entire process, including installing dependencies, compiling, and installing the binary. Follow the directions carefully, and [watch the video](#) if you have questions. One thing you will need to change (before you run the script) in the script is to cause the build process to use only one core. This prevents smaller computers like the Raspberry Pi and Quadra from overheating or over-currenting. You can actually skip this step, however, the computer will reach 64C, so maybe it's worth doing. Edit the script using your favorite text editor, and change this line (approximately line 159):

```
time make -j2
```

to this (removing the '-j2'):

```
time make
```

Then run the script. The build process took about 15 minutes on my Quadra. Once installed, you can find wfview under the Applications menu in the "Internet" category.

Configure wfview

This part is simple. Connect the radio. Open wfview. Configure the access to the radio under Radio Access. Be sure to select the serial port (it should be `/dev/ttyUSB0`). Enable "Use RTS for PTT" if your radio needs it (many older ones do. The 7300 does not.) Configure the baud rate. Turn on CI-V Transceive on the radio, and make a note of the radio's CI-V address. Newer radios, like the IC-7100, IC-718, etc, respond to our CI-V bus query and identify themselves. Older radios will need the address manually specified. Press "Connect" once you have the settings in. Be sure to press "Save Settings" once you get it configured successfully.

Under "Radio Server", check the box to turn on the radio server, and then press "Disconnect" to disconnect from the radio. Define a user account. Select "Qt Audio" as the audio system. Select your RX and TX audio sources. Make a note of the three ports listed above (by default, 50001, 50002, and 50003).

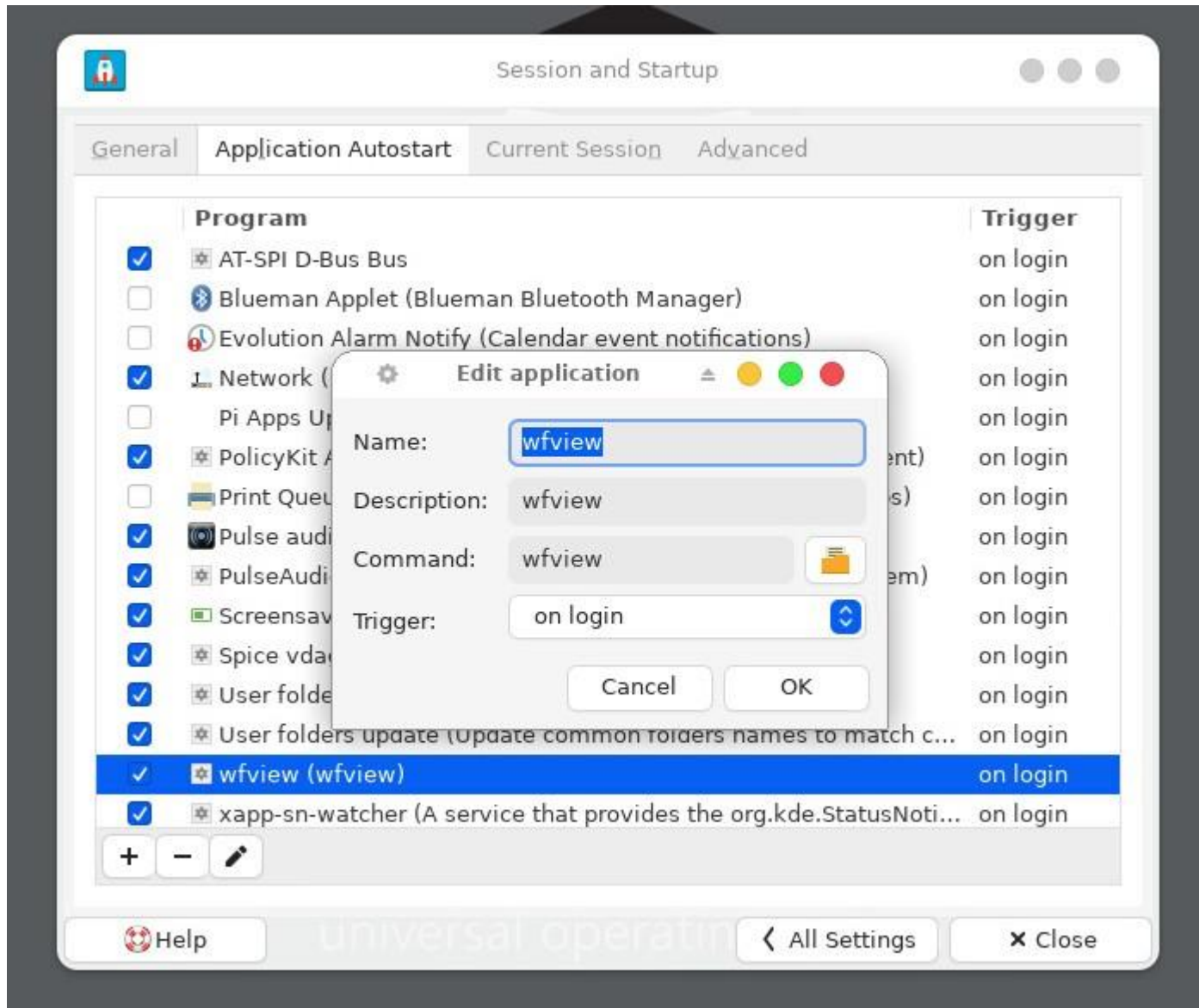
Under "User Interface", set the polling to manual and 250ms interval. This will keep the server side from polling the radio to death. Since the server and client do not sync their polling, sometimes you can have issues with older radios that have slower internal CPUs. The client side polling can remain automatic (20-75ms).

Press "Save Settings", and then press "Connect to radio". Check the Log to make sure the audio devices were used properly.

Additional System Configuration

Open the "Settings Manager" of xfce. It looks like a set of switches in the dock widget at the bottom of the screen. Scroll down to "Session and Startup" (look for the rocket ship icon). Then go under "Application Autostart" and add

wfview as a startup application. You can also uncheck Bluetooth and Evolution Alarm Notifier if you don't need those things.



Tune the radio to an active station. Connect to your wfview server using a client-side wfview instance. On the client, set the secondary meter to "TxRxAudio" (Settings, User Interface, Secondary Meter).

Set the audio level from the radio's receiver into the computer by using the Volume widget in the dock on the server. Click the speaker icon and then select "Audio mixer...". Now click the tab "Input Devices". If you do not click this tab, then you will be scaling the audio and not adjusting the analog level at the converter. So don't forget! Under input devices, you should see something like "USB Audio Device". Use the slider to adjust the level until there isn't any distortion. I like to see levels peaking at -6dB fs. I don't know how to configure this to save the gain setting, sometimes it seems to on its own and other times it doesn't.



Final Thoughts

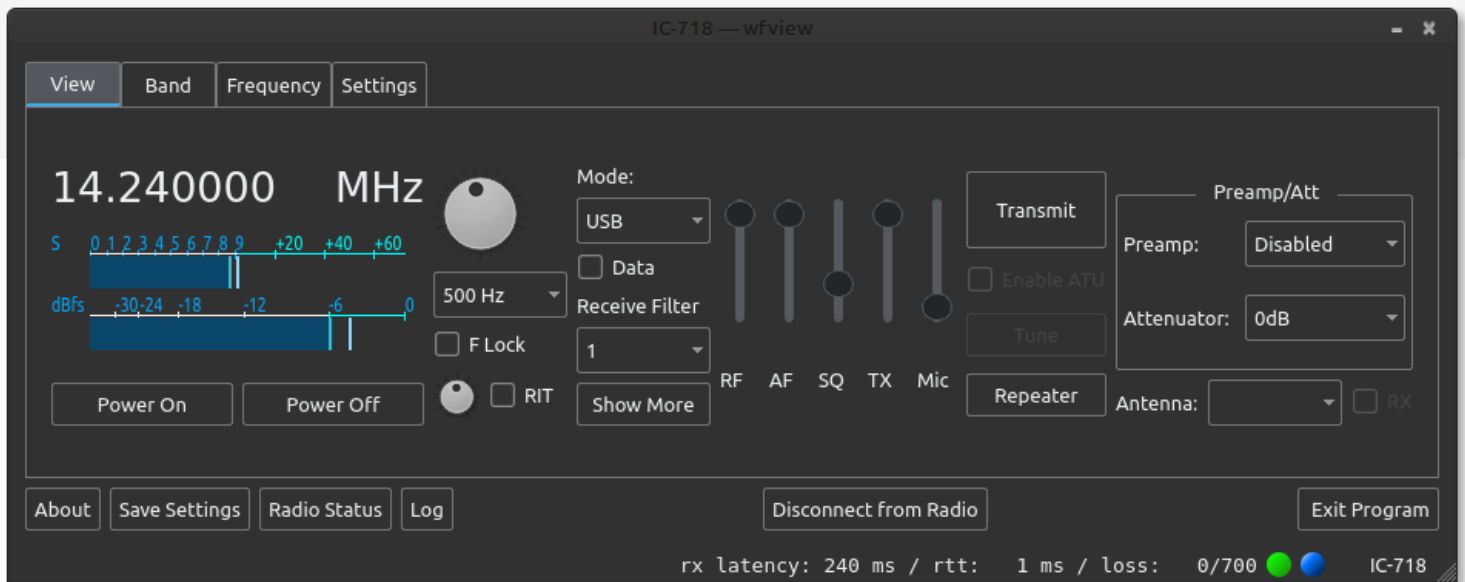
On my system, wfview uses less than 1% when there isn't an active connection. This is with my IC-718, mind you. If you are doing this with a 7300, it will be higher due to the server drawing the spectrum. It's worth a shot to configure wfserver once you have a working configuration file. wfserver does not have any tool to help you configure anything, it's purely command-line and without any user interaction. But, it understands the same configuration file.

When a user connects, I see that the total CPU usage jumps to 11% (and I'm using the opus codec, which should be higher than the PCM and uLaw options). The CPU temperature is generally around 46C, and I have not added a fan or anything to help with the thermal environment. I am seeing that the CPU idles at 480 MHz, and it will jump to 1704 MHz and then down to 720 MHz when a client connects to wfview. You can check using `sudo armbianmonitor -m`.

On storage, the quadra arrived with 8.8 GB available and it's down to 8.7 GB available with wfview installed. This is nice, the default image included a lot of the usual tools and libraries and still has plenty of space. htop shows memory usage is "490M/1.94G", and wfview is using 4.6% of that.

Overall what this means is that the computer isn't overly taxed. It can handle this task easily.

I remotely accessed my IC-718 using the Opus codec over my cellular hot spot from the park recently. Audio was smooth and control was consistently responsive. I did need to run the latency up a bit higher, around 250ms seemed to be good.



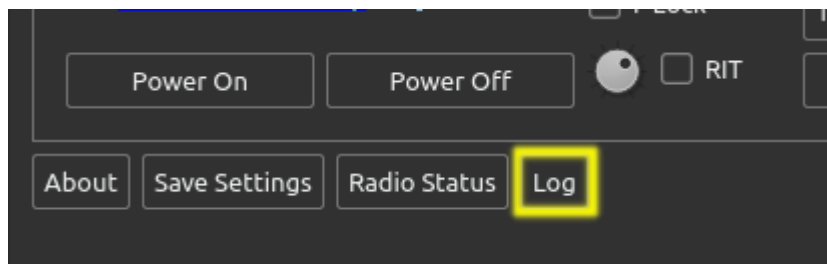
How to send a logfile

This page will tell you how to send your logfile to the wfview development team.

The steps are simple, but they must be followed carefully.

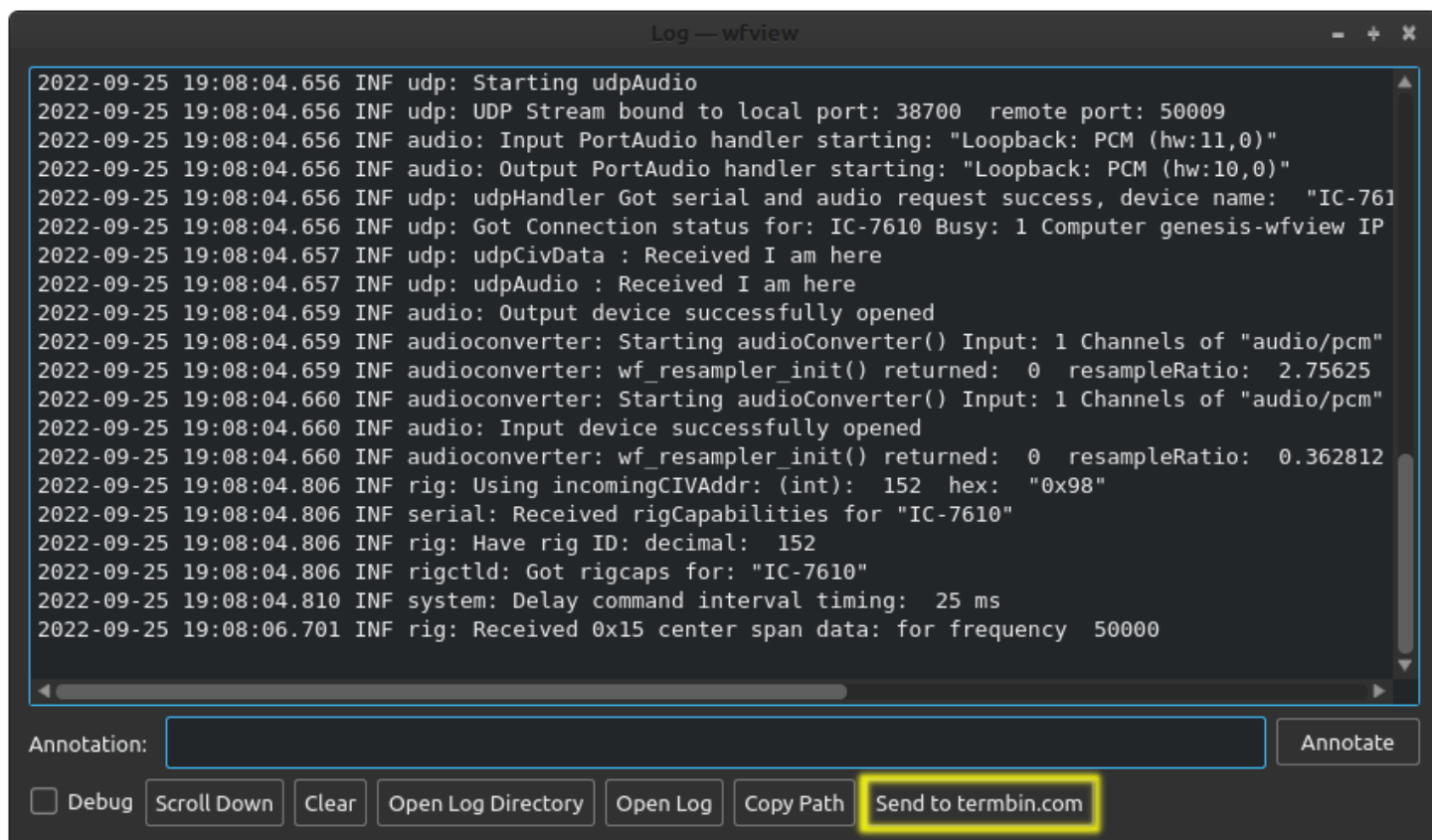
Begin by closing wfview and then re-opening it and replicating the issue as promptly as possible. This important so that you do not have a log file that is overly long or contains too much irrelevant information.

Once the issue has occurred, press the “Log” button in wfview.



This will cause the log window to open up. The log contains lots of useful information for the developers, but you may want to have a look through it to see if you see any issues you can address yourself.

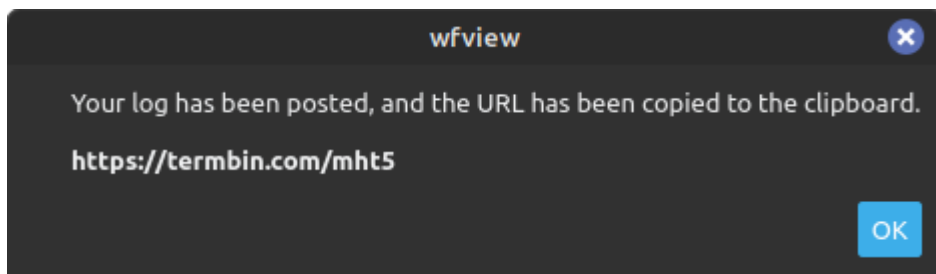
Here is an example log window:



Review the information and make sure you are comfortable sharing it. There isn't much information that is personally-identifiable, but you should check just to be sure. If you want to annotate the log with a message, you may, simply by typing a message into the log window and then pressing “Annotate”. For example, you might type in “This is wfview running on my PC with my IC-7300.” This is very helpful if you are going to send us several log files, for example, from a server and client set or for several radios.

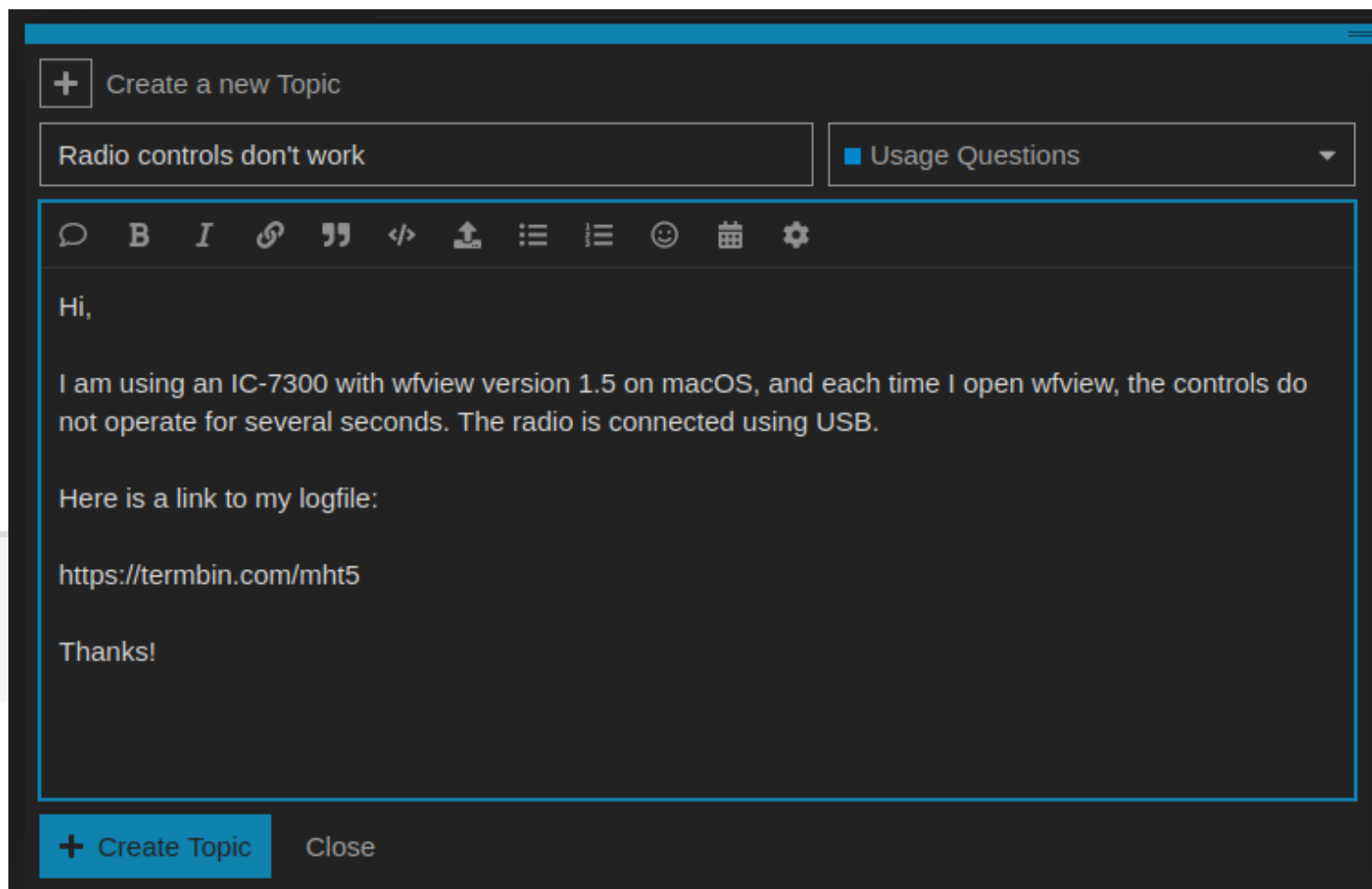
Once ready, **press** “Send to termbin.com”. This will cause the logfile to be sent to termbin. Note that *we do not operate termbin.com*. Termbin is a service that many other programs use. Thus, please understand, we cannot see

your log until you share the URL from the pop-up with us. The pop-up will appear about a second or two after you send the log file.



The resulting URL is already copied to your clipboard. What you need to do is then **send us the URL by pasting it** into a message on [our support forum](#). Once you send us that URL, we can all look at it together, which is very convenient.

For example, you might post it like this:



APRS decode from IC-9700 over Network

Objective

The objective is to be able to decode standard 1200 baud AFSK APRS packets using the network access from the IC-9700 and wfview's audio. This how-to focuses on Linux, however similar methods will work on other operating systems.

Setup

First, follow the [Audio Setup](#) guide for linux, creating two loopback devices, each with two entry points. Follow the document carefully. After you reboot, launch wfview and select the audio loopback devices. Make sure to carefully select the audio ports *per the linked page*. If you select the wrong ports, you won't have any audio. Select the ports and then Save Settings and close wfview. Optionally, use a program like Audacity to verify the audio.

Check that you can see the loopback devices via this command:

```
eliggett@genesis:~$ arecord -l
**** List of CAPTURE Hardware Devices ****
card 1: Generic_1 [HD-Audio Generic], device 0: ALC257 Analog [ALC257 Analog]
Subdevices: 1/1
Subdevice #0: subdevice #0
card 10: Loopback [Loopback], device 0: Loopback PCM [Loopback PCM]
Subdevices: 8/8
Subdevice #0: subdevice #0
Subdevice #1: subdevice #1
Subdevice #2: subdevice #2
Subdevice #3: subdevice #3
Subdevice #4: subdevice #4
Subdevice #5: subdevice #5
Subdevice #6: subdevice #6
Subdevice #7: subdevice #7
card 10: Loopback [Loopback], device 1: Loopback PCM [Loopback PCM]
Subdevices: 7/8
```

```
Subdevice #0: subdevice #0
Subdevice #1: subdevice #1
Subdevice #2: subdevice #2
Subdevice #3: subdevice #3
Subdevice #4: subdevice #4
Subdevice #5: subdevice #5
Subdevice #6: subdevice #6
Subdevice #7: subdevice #7
card 11: Loopback_1 [Loopback], device 0: Loopback PCM [Loopback PCM]
Subdevices: 7/8
Subdevice #0: subdevice #0
Subdevice #1: subdevice #1
Subdevice #2: subdevice #2
Subdevice #3: subdevice #3
Subdevice #4: subdevice #4
Subdevice #5: subdevice #5
Subdevice #6: subdevice #6
Subdevice #7: subdevice #7
card 11: Loopback_1 [Loopback], device 1: Loopback PCM [Loopback PCM]
```

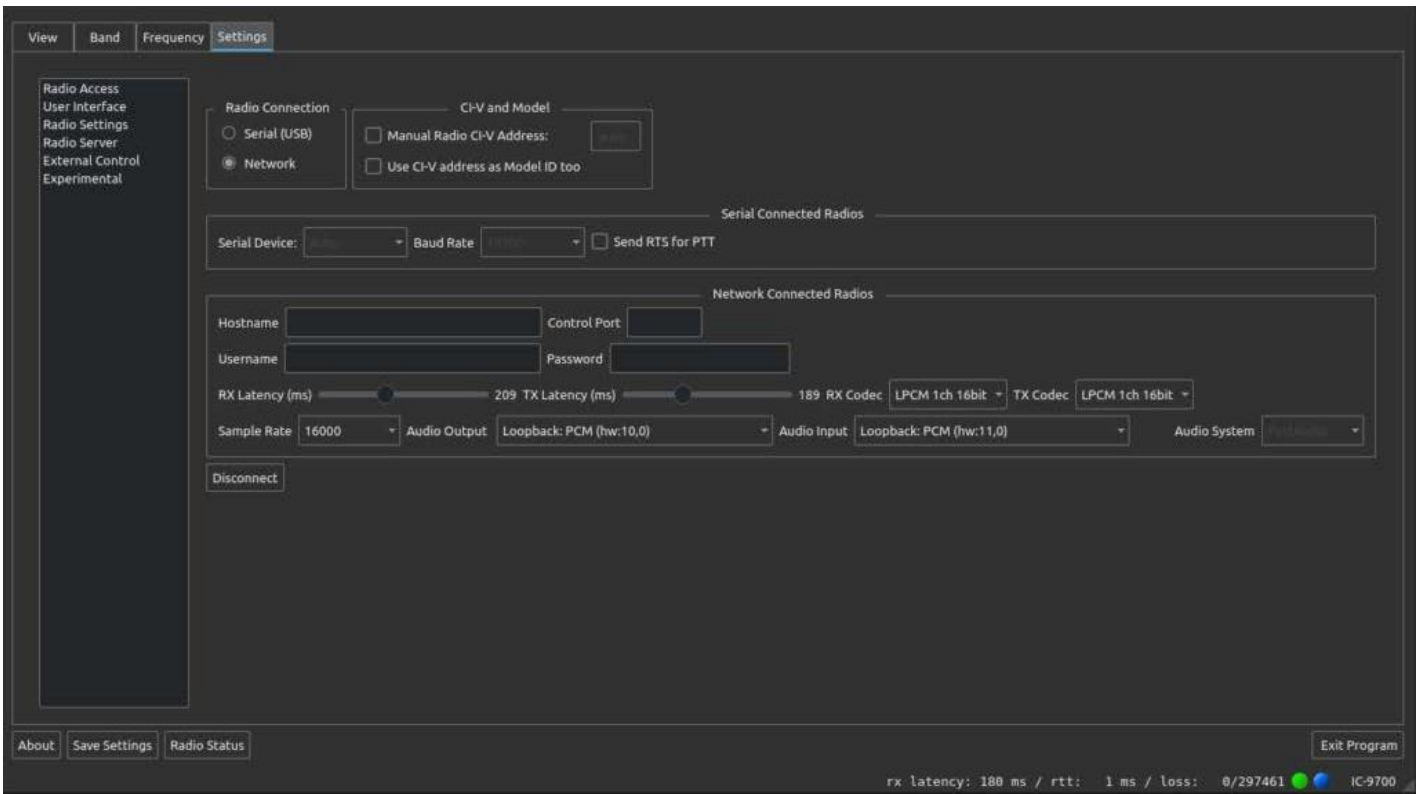
```
Subdevices: 8/8
Subdevice #0: subdevice #0
Subdevice #1: subdevice #1
Subdevice #2: subdevice #2
Subdevice #3: subdevice #3
Subdevice #4: subdevice #4
Subdevice #5: subdevice #5
Subdevice #6: subdevice #6
Subdevice #7: subdevice #7
eliggett@genesis:~$
```

You will need to install either direwolf or multimon-ng, or both, as well as sox:

```
$sudo apt-get install direwolf multimon-ng sox
```

Note that direwolf will provide a cleaner output and also allow you to contribute what you receive to the igate server, which is pretty cool. Don't be intimidated by the direwolf setup; for our purpose the setup is merely 7 lines, detailed below.

Minus the obvious hostname, username, and password, here is how my wfview Radio Access setup looks. Be sure to set the CODEC to LPCM 16bit. The sample rate should be at least 16k (you may have more success setting this to 48k with direwolf and 22050 for multimon-ng, but I did not experiment. The sox commands below will set the loopback to 48k in both cases). To be clear, the Audio Output should be "Loopback: PCM (hw:10,0)" and the Audio Input should be "Loopback: PCM (hw:11,0)". This is per the Audio Configuration linked above.



wfview audio setup for APRS

Setup the APRS program:

Note: It is far easier to start the APRS chain *prior* to opening wfview. This is because the APRS programs (and sox) are less-flexible about sample rates and audio formats in general, so we start sox and the APRS programs first, so as to define the audio format which wfview will then follow.

For direwolf, you need to follow the most basic setup first. I made the following direwolf config file, which I called "direwolf.conf":

```
ADEVICE null null
CHANNEL 0
MYCALL W6EL
IGSERVER noam.aprs2.net
IGLOGIN W6EL PASSWORDHERE
MODEM 1200

PBEACON sendto=IG delay=0:40 every=20 symbol=R& alt=475 lat=34.23386N long=118.25866W comment="iGATE
AAPRS rx-only"
```

You should use [this link](#) to get your APRS password. I don't really understand how this works but in any case, it worked for me.

For multimon-ng, there isn't any setup.

Starting the APRS programs:

For direwolf, use this command (copy-paste carefully! Be sure to enter the path for the direwolf config file correctly.):

```
sox -q -t alsa hw:10,1,0 -e float -b 32 -r 48000 -t raw -b 16 -c 1 -e signed-integer -r 48000 - |
direwolf -c ./direwolf.conf -r 48000 -D 1 -t 0 -
```

For multimon-ng:

```
sox -q -t alsa hw:10,1,0 -e float -b 32 -r 48000 -t raw -b 16 -c 1 -e signed-integer -r 22050 - |
multimon-ng -t raw -a AFSK1200 -A /dev/stdin
```

Note, if you wish, you can start sox without the "-q" flag and without the pipe to the APRS program. In this mode, you can see the audio level in a small bar-graph. There won't be any decoding, but it is fine for testing the audio pipe.

I will note here that it took me about two hours to come up with those two commands. The sample rate for multimon-ng must be 22050, and multimon-ng doesn't seem to provide any way to select another audio device. Direwolf is flexible with sample rates. sox must be setup with the loopback audio format matching any format that has already been applied to the loopback device, and it won't tell you what that is! This is why I start sox first, before I open wfview, so that sox may *define* the audio format, and wfview will *follow* whatever format is applied.

Monitoring APRS packets

With one of the above commands (sox + direwolf or sox + multimon-ng), launch wfview and confirm the loopback devices are selected. If necessary, press Disconnect, select the loopback devices per the audio configuration how-to, and then press Save Settings and hit Connect. Now tune wfview in to your local APRS frequency. 144.39 FM is used in SoCal, for example. Turn up the AF Gain slider all the way (if the signal is too loud or distorted, turning this down will just make it quiet-yet-distorted, so adjust the LAN RX level at the radio if needed). I recommend setting the squelch so that it opens with the slightest signal, or all the way down if desired. Use the "RxAudio" meter to monitor the audio coming from the radio.

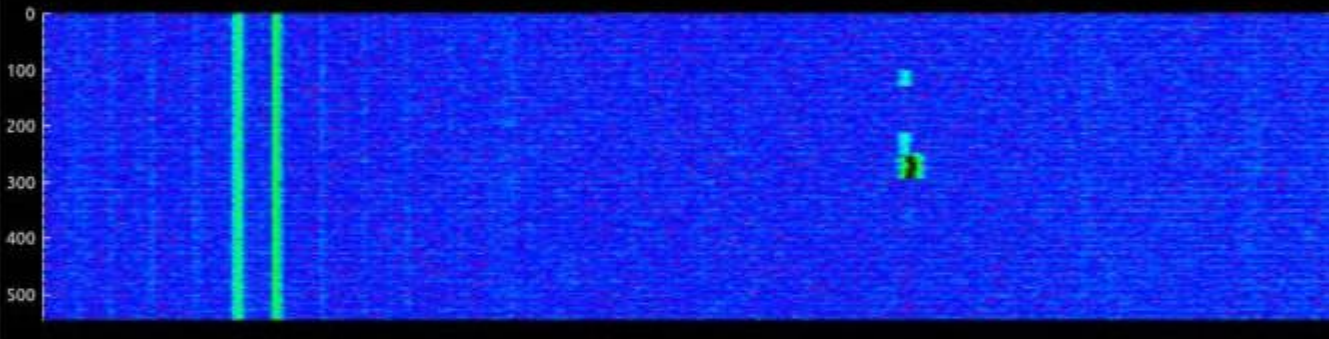
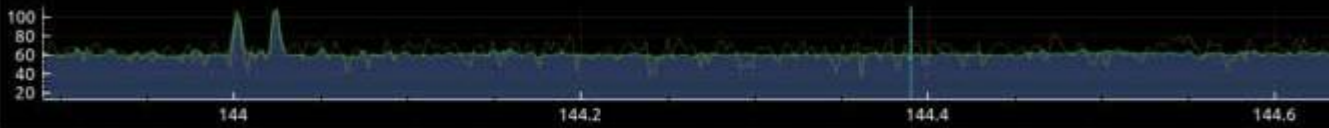
To visualize the station locations, you might wish to try xastir or YAAC. I have not tried either yet.

```
APRS: WC6Q>APWA01,N6EX-4*:@183156h3412.06N/11834.80wk090/010 /A=000650 Fore 103% (30/25) WC6Q.com
APRS: KK60Z-15>APT311,N6EX-4*:>TinyTrak3 v1.1
APRS: WC6Q>APWA01,N6EX-4*:@183216h3412.06N/11834.80wk/A=000649 STOP 102%(18) (24/20) WC6Q.com
APRS: WA6IAJ-1>APMI06,N6EX-4*:T#130,187,077,012,098,159,00000000
APRS: KC6UXQ-5>APDR16,N6EX-4*:=3346.56N\11749.96wk071/048/A=000131 APRSDroid
APRS: KF6NMZ-1>APTT4,N6EX-4*:T#310,127,154,255,041,013,00101011
APRS: KC6UXQ-5>APDR16,N6EX-4*:=3346.73N\11749.84wk356/052/A=000121 APRSDroid
APRS: KE6VRK-9>SSUQTV,N6EX-4*:.lcl"z>/^46} 1
APRS: KE6VRK-9>SSUQTV,W6SCE-10*:.lcl"z>/^46} 1
APRS: KC60YN>APOTW1,N6EX-4*:#3429.67N/11942.76W 132/004g005t092P000h30b102070TW1
APRS: W6MAF>APTW01,OAKHLS*,WIDE1*,KELLER*,N6EX-4*: 09061132c195s005g005t098r000p000P000h10b10158tU2k
APRS: W6MAF>APTW01,OAKHLS*,WIDE1*,KELLER*,N6EX-4*:#3422.10N/11724.50W PHG6150/A=003900/Oak Hills WXTrak
APRS: KM6FDI-5>APMI0A,KELLER*,WIDE1*,N6EX-4*:#3718.37N/12153.11W>APRS Voyager U=14.6V, T=??.7C.
APRS: WB0AL-9>ST10UX,W6SCE-10*,N6EX-4*:.S+lTnu/^6,}Over the road truck driver at Panther Premium_
APRS: HEAPS>APN383,N6EX-4*:#3414.09N/11708.44W PHG4390/Heaps Peak/A=006420/k6tvi@cal-net.org
APRS: WA6MHA-11>APOTW1,W6SCE-10*,N6EX-4*:#3410.50N/11828.90W_092/003g009t101P000h26b10125V1380TW1
APRS: KF6NMZ-1>APTT4,N6EX-4*:#183404h/=9r70VEN>"/TinyTrak4 Alpha
APRS: KM6FDI-5>APMI0A,KELLER*,WIDE1*,N6EX-4*:T#041,199,000,000,000,000,000,00000000
APRS: W6PWW>S4RYQS,W6SCE-10*,N6EX-4*:.B.l K\]^92}I Monitor PAPA1&DARN10=
APRS: N06X-10>S4QRRX,W6SCE-10*,N6EX-4*:.0)|6>/]^6l}=
APRS: KM6AHX-12>APOTU0,N6EX-4*:> CHG STN ON
APRS: KM6AHX-12>APOTU0,N6EX-4*:#3411.20N/11813.02W_14.2V
APRS: KM6AHX-12>APOTU0,N6EX-4*:#3411.20N/11813.02W_057/001g..t094p000h43T2WX
APRS: WB0AL-9>ST10VQ,W6SCE-10*,N6EX-4*:.S=mrBu/^6,}Over the road truck driver at Panther Premium_
APRS: KM6FDI-5>3T1QUU,KELLER*,WIDE1*,N6EX-4*:.1E3y\^9T}In conference KM6FDI-9
APRS: KM6FDI-5>3T1QUU,KELLER*,WIDE1*,W6SCE-10*:.1E3y\^9T}In conference KM6FDI-9
APRS: KK60Z-15>APT311,N6EX-4*:/061835z3355.03N/11759.58Wv269/032/A=000291
APRS: WC6Q>APWA01,W6SCE-10*,N6EX-4*:@183605h3413.02N/11834.79wk000/031 /A=000701 Fore 103% (30/25) WC6Q.com
APRS: KF6NMZ-4>APBP01,N6EX-4*:#KA6ECT>LAECTC,TCP/IP,KF6NMZ-4*:#C06113316*061836z3413.88N\11824.18W\1125-Traffic Haz
ard I5 S / Laurel Canyon Blvd
APRS: KM6FDI-5>3T1QUU,KELLER*,WIDE1*,N6EX-4*:.1E3y\^9T}In conference KM6FDI-9
APRS: KM6FDI-5>3T1QUU,KELLER*,WIDE1*,W6SCE-10*:.1E3y\^9T}In conference KM6FDI-9
APRS: WC6Q>APWA01,N6EX-4*:@183630h3413.20N/11834.79wk/A=000708 Fore 100% (31/25) WC6Q.com
APRS: XE2SI-10>APMI06,WA6Z5N*,N6EX-4*:#061836z3228.44N/11653.81W-PHG53304-WX3IN1
APRS: KE6VRK-9>SSUQTV,N6EX-4*:.lcl w>/^46} 1
APRS: N06X-10>S4QPVO,CAMRIO*,WIDE1*,N6EX-4*:.0f1"i>/]^6\}=
```

APRS: N06X-10>S40PV0,CAMRIO*,WIDE1*,W6SCE-10*: .0f!*;>/]6\}=
APRS: KF6NMZ-1>APTT4,N6EX-4*:T#312,127,154,255,040,012,10111011
APRS: KM6AHX-12>APOTU0,N6EX-4*:> CHG STN ON
APRS: KM6AHX-12>APOTU0,N6EX-4*:!3411.20N/11813.02W 14.2V

View Band Frequency Settings

Spectrum



Spectrum Mode: Center Span: ±500k Edge 1 ToFixed Clear Peaks Enable WF Theme: Ion

144.390000 MHz



Mode: FM

Data

Receive Filter 1

F Lock

RIT

Show More



Transmit

Preamp/At

Preamp: Pre #

Attenuator: 0dB

Repeater

Antenna:

About Save Settings Radio Status

rx latency: 188 ms

Serial Port Management

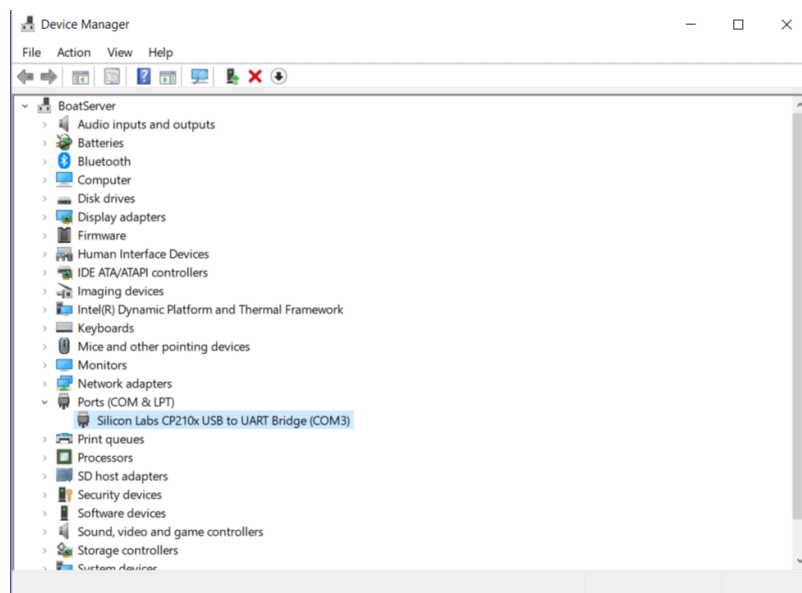
This page will help you determine which serial port is your radio, and in some instances, how to give the port a more obvious name. For [Linux](#), there are detailed directions on the use of udev.

macOS Configuration

macOS users wishing for more consistent device names may wish to check out [Discover 7610](#) and [CI-V Redirect](#), both by [Kok Chen, W7AY](#). Note that while these tools are designed for the 7610, the method outlined on the pages shows how to use the System Profiler to see which device is the radio, and that method should be equally valid for other radio models. You can access the [System Profiler](#) (also known as “System Information”) by choosing “About this Mac” from the apple menu and then pressing “System Profiler”. You can also use the [system_profiler](#) and [ioreg](#) commands, as outlined [here](#).

Windows Configuration

Windows users can use the [Device Manager](#) to determine which COM device is the radio. To connect to your rig via USB, the (USB) serial port will appear in the list of available devices in the Serial Device: combobox of wvview. Where your rig has two serial ports (A/B) you must select port A (this is usually the lower numbered of the two).



Linux Configuration

With linux, the user logged in may or may not actually have write-access to the serial port, which is needed in order to run software like wvview. There are also times where it is useful to know which serial devices are connected to radios (or perhaps to which radios).

Basic Serial Port Ownership and Group Membership:

A quick command may be run to temporarily enable full access, however, this is not a good long-term solution as the command must be run every time:

```
sudo chown $USER /dev/ttyUSB*
```

A better solution is to modify the current user’s account, adding membership into the group that the system defaults to for serial ports.

First determine the name of the group:

```
ls -l /dev/ttyUSB*
```

The group name will be the second name, after the owner's name. Here is an example output:

```
eliggett@zep:~$ ls -l /dev/ttyUSB*
crw-rw---- 1 root dialout 188, 0 May 2 01:48 /dev/ttyUSB0
crw-rw---- 1 root dialout 188, 1 Apr 3 14:38 /dev/ttyUSB1
crw-rw---- 1 root dialout 188, 2 Apr 3 14:38 /dev/ttyUSB2
```

Here, the owner is root and the group owner is "dialout". On some systems, this might be "modem".

To append group membership, issue the following command:

```
sudo usermod -aG dialout $USER
```

You may need to log out and log back in for this to take effect.

On some systems, you really will need to be the owner of the port, and this depends upon how the system security is configured. In this next section, you'll have the opportunity to become the default serial port owner.

Using UDEV for nice port names and default ownership:

Linux provides a subsystem called "udev" which, among other things, helps name entries in /dev when devices are connected to the computer. Using a custom udev entry, we can create symlinks to nice device names. For example, on my system, /dev/IC9700 is my IC-9700 serial port, and /dev/IC7300 is my IC-7300 serial port. These names are much easier to remember and to use than their default ttyUSB names, which will have different numbers when you reboot or re-connect a radio.

To do this, we'll first need to determine the USB device serial number for each radio.

Issue the following command to show the device ids:

```
ls -l /dev/serial/by-id
```

You will see output similar to what follows:

```
lrwxrwxrwx 1 root root 13 Feb 3 19:25 usb-Silicon_Labs_CP2102N_USB_to_UART_Bridge_Controller_IC-970
_013001202_A-if00-port0 -> ../../ttyUSB3
lrwxrwxrwx 1 root root 13 Feb 3 19:25 usb-Silicon_Labs_CP2102N_USB_to_UART_Bridge_Controller_IC-970
_013001202_B-if00-port0 -> ../../ttyUSB2
lrwxrwxrwx 1 root root 13 Feb 3 19:25 usb-Silicon_Labs_CP2102_USB_to_UART_Bridge_Controller_IC-7300
_03001507-if00-port0 -> ../../ttyUSB4
lrwxrwxrwx 1 root root 13 Feb 3 19:25 usb-Silicon_Labs_CP2102_USB_to_UART_Bridge_Controller_IC-7851
_03001140_A-if00-port0 -> ../../ttyUSB0
lrwxrwxrwx 1 root root 13 Feb 3 19:25 usb-Silicon_Labs_CP2102_USB_to_UART_Bridge_Controller_IC-7851
_03001140_B-if00-port0 -> ../../ttyUSB1
```

The part we're concerned with here isn't how many nice radios this person has, but rather, the name and serial numbers, for example "IC-7851_03001140_A". Here, the model is IC-7851, and the serial number is 03001140. Some Icom radios present more than one serial port, and in most cases, we are interested in the "A" port for rig control. The "B" port is for other uses.

Now we need to make a new udev rule. Open up a text document and prepare the text that we will put in. Start with this:

```
SUBSYSTEM=="tty", ATTRS{idVendor}=="10c4", ATTRS{idProduct}=="ea60", ATTRS{serial}=="IC-7851 030011
40 A", SYMLINK+="IC7851A", OWNER="eliggett"
```



```
SUBSYSTEM=="tty", ATTRS{idVendor}=="10c4", ATTRS{idProduct}=="ea60", ATRS{serial}=="IC-7851 030011
40 B", SYMLINK+="IC7851B", OWNER="eliggett"
SUBSYSTEM=="tty", ATTRS{idVendor}=="10c4", ATTRS{idProduct}=="ea60", ATTRS{serial}=="IC-7300 030015
07", SYMLINK+="IC7300", OWNER="eliggett"
SUBSYSTEM=="tty", ATTRS{idVendor}=="10c4", ATTRS{idProduct}=="ea60", ATTRS{serial}=="IC-9700 130012
02 A", SYMLINK+="IC9700A", OWNER="eliggett"
SUBSYSTEM=="tty", ATTRS{idVendor}=="10c4", ATTRS{idProduct}=="ea60", ATTRS{serial}=="IC-9700 130012
02 B", SYMLINK+="IC9700B", OWNER="eliggett"
```

Edit the file so that your serial numbers, model numbers, and username are reflected. Remove any lines for radios you don't own (yet).

Save the file to /tmp/rules.txt (or any other location you can easily get to)

Now from the terminal, copy the file in to the udev rule location for your system (commonly as shown):

```
sudo cp /tmp/rules.txt /etc/udev/rules.d/90-persistent-usb.rules
```

To test it out, disconnect the USB cables from your rigs, and then reconnect. You should now see nice entries in /dev for your radios:

```
eliggett@zep45:~$ ls -l /dev/IC*
lrwxrwxrwx 1 root root 7 Apr 3 14:38 /dev/IC7300 -> ttyUSB0
lrwxrwxrwx 1 root root 7 Apr 3 14:38 /dev/IC9700 -> ttyUSB1
lrwxrwxrwx 1 root root 7 Apr 3 14:38 /dev/IC9700-data -> ttyUSB2
```

(Note that I named my "B" port "-data". You can select whatever you like for yours.)

You can make individual shortcuts to wfview which use settings files that specify different serial ports. See the [Preferences File](#) manual page as well as the [Command-Line Arguments](#) page for more information.

Command-Line Arguments

This page will describe various ways to start wfview from a command-line with command-line arguments. These arguments are most useful to start wfview from a specific settings file or to enable debug-level logging.

How to supply arguments

To supply arguments, one must generally open a terminal and then enter in the name of the program followed by the arguments. For example, from linux, assuming wfview is within your current \$PATH:

```
wfview --argument
```

From macOS, there are two ways. One way is to change directory to the inside of the application bundle, and then to issue commands like so:

```
/Applications/wfview.app/Contents/MacOS/wfview --argument
```

You can drag the wfview.app icon into the terminal and edit the text. However, it may be easier within macOS do it like this:

```
open /Applications/wfview.app --args --argument
```

With Microsoft Windows, you will need to open a `cmd.exe` window. From there, you can drag the wfview.exe icon into the cmd window. This should enter in the full path of the executable. Once entered, you may add arguments to the text.

Most commonly used arguments

The most commonly used arguments are as follows:

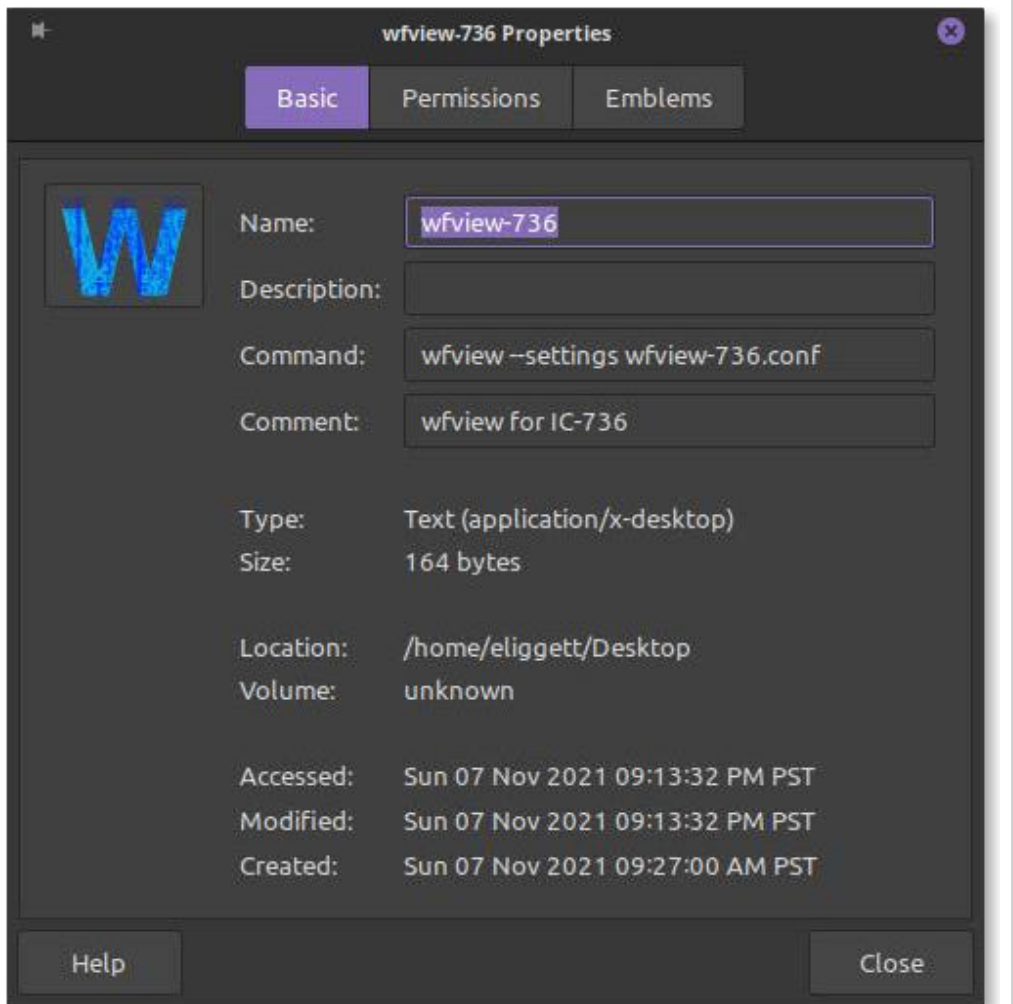
Settings file

To specify an alternate settings file, the `--settings` argument may be supplied followed by either an absolute path (generally starting with a slash), a relative path (starting from the current working directory), or simply a filename. Specifying only a filename is the preferred and simplest method; wfview will automatically generate a full filename to the operating system's user-writable application settings folder. On linux, this would be `~/.local/wfview/wfview`. The file doesn't need to initially exist, wfview will create the specified file when you press "Save Settings".

You can read more about the settings file format [here](#). A general strategy is to simply supply `--settings` followed by your radio's name:

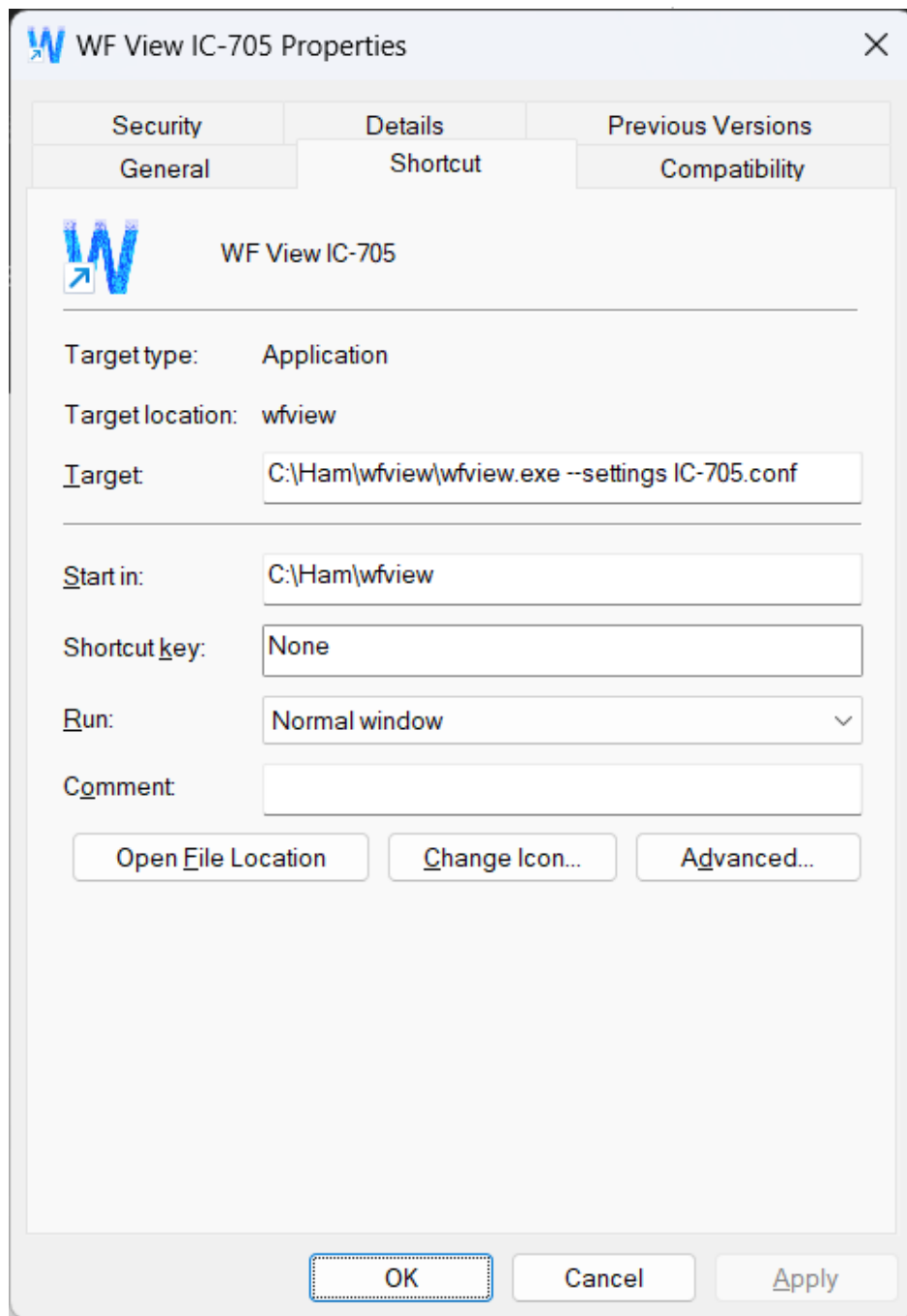
```
wfview --settings ic-718.conf
```

Once wfview is set up (under the Settings tab), do not forget to press "Save Settings". wfview will then write the settings file to disk, and you will from then on be able to immediately open wfview to your various radios simply by making desktop shortcuts that include the needed settings file within the arguments. Here is how one of mine looks for linux. I simply made it by right-clicking on the wfview icon within the applications menu, and choosing "add to desktop". From there, I customized the launcher to include the needed command-line arguments:



Desktop launcher for Linux with specific settings file specified.





This is an example wfview shortcut under windows which specifies a `--settings` argument.

Debug Mode

To enter debug mode, simply pass `--debug` to wfview. This will cause wfview to log much more information to the [log file](#), including the contents of most commands and most radio replies (except for periodic polling commands and spectrum data).

```
wfview --debug
```

Full List of Supported Command-Line Arguments

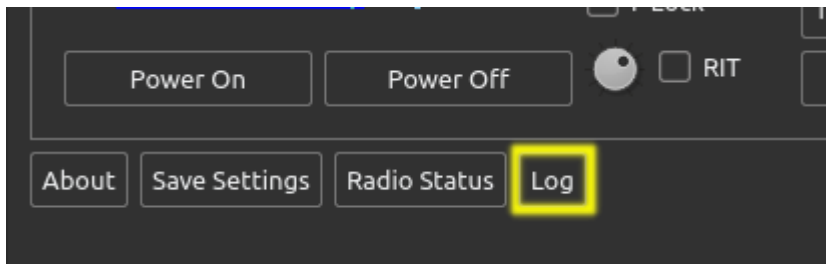
The following is a list of all supported command-line arguments. Many are intended for development use only and are not advisable to be used. The commands are listed in their short and long formats. The parsing of the commands takes place in [main.cpp](#).

- `-d --debug`
 - Enter debug mode. Causes more verbose logging and additional features to become visible. Many of those features will be incomplete.
- `-l --logfile`
 - Specify alternate location for the [log file](#).
- `-s --settings`
 - Specify settings file. Path may be absolute, starting with a slash, may be relative from the current working directory, or may be a simple filename, to which wfview will determine the correct path for user-accessible application data for your operating system. We recommend not specifying a path, merely specify a filename. A file that doesn't exist yet will be automatically created when you Save Settings. **DO NOT** specify path names using “`~`”, as this will create a directory within the application data folder named “`~`”. This is a qt bug.
- `-h --help`
 - Show help text and exit

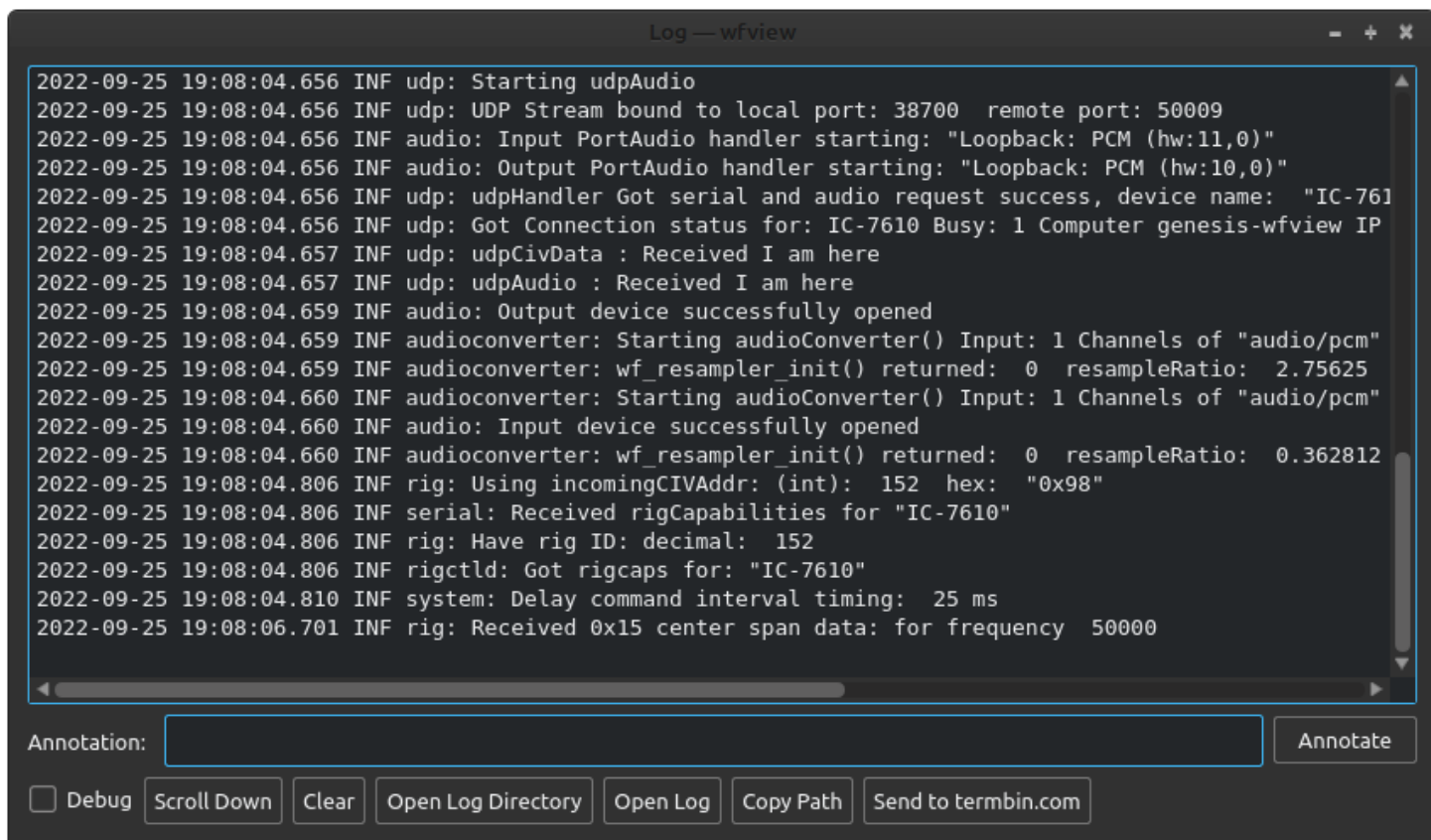
Log file

wfview writes a log file, which can be useful to look at when things don't work. When you witness unexpected program behavior, click the Log button. This is like your gauges in a car — look here first when there is trouble. When you post to [our support forum](#), it'll save everyone a step if you check the log first. If you see anything interesting, use the “send to termbin” function and paste a link to the log into your support forum post. It helps a lot!

Since wfview version 1.5, the program has a button, “Log” in the bottom row.



Press this button to open the log window, which will display the log and allows for several useful actions:



Here is what each button does:

- Annotation: Type any text that you would like to enter into the program's log file. This is useful to take notes. You might want to take notes about malfunctions you experience, or note what you are trying to do just before an action takes place.
- Debug: Checking here enables more verbose logging
- Scroll Down: Scrolls the contents of the window down, useful if you have scrolled up and just want to jump to the bottom
- Clear: Clears the display. Does not clear the contents of the log file.
- Open Log Directory: Opens the directory where the log is stored.
-
-

Open Log: Opens the log file in your operating system's text editor.

Copy Path: Copies the path of the log file to the clip board

- Send to termbin.com: Sends the displayed log entries to termbin.com, and provides you with a URL from which you can then access and share the log file.

About Send to termbin.com: When you press this button, the log (as displayed) will be sent to termbin.com. This is the same website used by many linux commands to help people share debugging information. Keep in mind that the log file may contain information such as your user name and ip address. When you are looking for help with wfview, if you press this button, a URL will be copied to the clipboard automatically, which you can then paste into a post on our support forum. This is by far the easiest way to share a log file.

Default Log File Location

Remember, you can copy the log file path using the "Copy Path" button in wfview.

- Linux: /tmp
- Windows: %TEMP%/wfview
- macOS: /private/\$TMPDIR/org.wfview.wfview (use the "Copy Path" button in wfview's log GUI to obtain the path quickly)

The log file name is, by default, in for form of "`wfview-yyyymmddhhmmss.log`". The time stamp refers to the moment the program was started. You can override this behavior by specifying the `--logfile` argument when you start wfview, for example:

```
wfview --logfile /tmp/IC-7300.log
```

would cause the log file to be named "/tmp/IC-7300.log". Paths are relative to the current working directory if an absolute path isn't given.

Debug Logging

To aid in debugging, wfview may be started with a debugging flag, which causes much more data to be logged to the log file. To start wfview with debug logging, run wfview with either a `-d` flag or `--debug`. For example:

```
wfview --debug
```

Once done, the log file will contain a much greater amount of information. For macOS, run the following terminal commands:

```
cd /Applications
open wfview.app --args --debug
```

Note that you can turn on debug logging from within wfview, by checking the "Debug" checkbox in the Log window. However, starting wfview with debug logging will generate some more messages concerning the startup sequence.

Logging Features

When wfview starts, it will begin with "INF system: Starting wfview". Each line has the current date, time, logging category, and subsystem label. Logging categories include:

- INF (informative)
- DBG (debug)
- WRN (warning)

Subsystems (logical units of code basically) include:

- system (wfmain.cpp)
- audio (audio handler)
- audioconverter (sample rate conversion, channel count, bit depth, etc)
- udpaudio (network audio)
- udp (communication to and from a radio over network)
- rig (rigCommander)
- serial (pseudo-term and serial-connected rigs)
- gui (gui-specific parts of wfmain.cpp)
- user (user annotations)

Useful Log Messages:

Briefly, here are some useful things to look for:

Rig-ID: This is how the radio is identified:

```
2021-05-15 16:38:41.230 DBG rig: ---Rig FOUND from broadcast query:
2021-05-15 16:38:41.230 INF rig: Using incomingCIVAddr: (int): 148 hex: 94
2021-05-15 16:38:41.230 INF rig: Have rig ID: decimal: 148
2021-05-15 16:38:41.230 DBG system: Rig name: "IC-7300"
2021-05-15 16:38:41.230 DBG system: Has LAN capabilities: false
2021-05-15 16:38:41.230 DBG system: Rig ID received into wfmain: spectLenMax: 475
2021-05-15 16:38:41.230 DBG system: Rig ID received into wfmain: spectAmpMax: 160
2021-05-15 16:38:41.230 DBG system: Rig ID received into wfmain: spectSeqMax: 11
2021-05-15 16:38:41.230 DBG system: Rig ID received into wfmain: hasSpectrum: true
```

In the above case, the rig was identified by means of a broadcast request (to CI-V address 0x00).

With debug logging enabled, the following additional data are available:

Traffic to the radio: wfview logs data to the radio, except for regularly polled queries (such as the s-meter). The format is as follows:

```
2021-05-15 16:38:41.267 DBG rig: Final payload in rig commander to be sent to rig:
2021-05-15 16:38:41.267 DBG rig: ---- Begin hex dump -----:
2021-05-15 16:38:41.267 DBG rig: "INDEX: 00 01 02 03 04 05 "
2021-05-15 16:38:41.267 DBG rig: "DATA: fe fe 94 e1 03 fd "
2021-05-15 16:38:41.267 DBG rig: ----- End hex dump -----
```

Here, we are requesting the current frequency (command 0x03) from the radio at address 0x94.

Traffic from radio: wfview logs any data received from the radio, except for spectrum and metering data, since these requests happen so frequently that logging them takes a lot of resources. The format is as follows:

```
2021-05-15 16:38:41.270 DBG rig: ---- Begin hex dump -----:
```

```
2021-05-15 16:38:41.271 DBG rig: "INDEX: 00 01 02 03 04 05 06 "  
2021-05-15 16:38:41.271 DBG rig: "DATA: 03 00 20 28 07 00 fd "  
2021-05-15 16:38:41.271 DBG rig: ----- End hex dump -----
```

You can see that the CI-V preamble and to/from address portion is already stripped out. In this case, the rig is responding to a frequency query request (command 0x03). The indicated frequency is 7.282000 MHz.

Preferences File

The preferences file contains data that are read upon opening wfview. The contents are saved upon pressing the Save Settings button under Settings.

These data include settings for the program's appearance, radio connection, window position, and more. Editing this file is not recommended; do so at your own risk, and only if you need to do so.

With each release of wfview, we do retire and add settings, so you may see some attributes from [prior versions](#), which are ignored.

Accessing:

The preference file may be accessed as follows:

macOS: Use the defaults command, specifying the org as org.wfview.org

Linux: Read the plain text file from `~/.config/wfview/wfview.conf`

Windows: Configuration is stored in the Windows Registry under **"HKEY_CURRENT_USER\SOFTWARE\wfview\wfview"**. Please be careful when editing the Windows Registry as incorrect usage of the regedit.exe utility can seriously damage your Windows installation. DO NOT change anything outside the registry key above unless you know what you are doing!

Note: You may specify an alternate settings file using the `--settings` command-line argument. See [here](#) for a list of [arguments](#).

Preference File Sections and Attributes:

Each section is as follows (note that the qt library does not allow wfview to determine the exact order these will appear in the file, and thus, they may be re-ordered by the user for clarity):

[Controls]

These options are from options under the Settings tab:

- EnablePTT: Set to 'true' or 'false' to enable or disable the PTT controls of wfview. Does not change the radio's own PTT lock controls.
- NiceTS: Set to 'true' or 'false' to enable or disable the frequency rounding that can occur when tuning the frequency or double-clicking in the spectrum.

[Interface]

Most of these options are from checkboxes under the Settings tab:

- ConfirmPowerOff: Boolean value to disable the "Are you sure you want to power off" dialog box.
- ConfirmExit: Boolean value to disable the "Are you sure you want to quit?" dialog box
-
-

- StylesheetPath: Set to the path of a custom stylesheet, relative to /usr/share/wfview/stylesheet
- UseFullScreen: Set to 'true' or 'false' to enable full screen on startup.
- UseSystemTheme: Set to 'true' or 'false' to enable or disable using the system's theme (versus a stylesheet — included or user-added)
- splitter: The position of the draggable plot-to-waterfall vertical space splitter. Probably best not to edit this by hand.
- windowState: The state of the window (position and maximized status)
- windowGeometry: The location and size of the window. Remove this variable and the one above if you have difficulty seeing the wfview window, this will restore the default.
- WFAntiAlias: boolean for waterfall anti-alias
- WFInterpolate: boolean for interpolation between the waterfall data points and the pixels on the screen
- WFLength: length of waterfall display. Values should be from 160 to 1023.
- Meter2Type: Selected type of meter for the secondary meter. Integers represent the meter number per the enumerated types in the source code.
- plotCeiling: value of the "top" slider
- plotFloor: value of the "bot" slider
- WFTheme: Selected color scheme theme for the waterfall
- underlayMode: The selected mode of plot underlay:
 - 0: none
 - 1: Peak Hold
 - 2: Buffered Peaks
 - 3: Average
- underlayBufferSize: The size of the buffer. Valid numbers are 2 to 128.

[Radio]

These options are specific to serial port access:

- RigCIVulnt: Use this parameter to manually set the CI-V address of the radio, in unsigned integer format. If set, wfview will not broadcast a request for all radios to identify, instead, it will send a rig ID request only to the specified CI-V address. This is useful if you have a radio that does not respond to broadcast queries on the CI-V bus, or, if you have more than one radio on the same physical CI-V bus (very rare in the post-USB-adaptor decades). If unset or set to 0, wfview will automatically find any connected radio (regardless of if that radio is a known model) and attempt to control it. The default value is 0, and is recommended for most users.
- CIVisRadioModel: Boolean value. When set to "true" causes wfview to assume the radio model is the same as the manual set CI-V above. For older radios that do not answer Rig ID queries, supply the default CI-V address above and then set this parameter to true.
- SerialPortBaud: Use this attribute to specify a custom baud rate, often needed on older radios. The default is 115200, required by the newer radios that provide spectrum data over serial.
- SerialPortRadio: Use this attribute to specify a custom path to the serial port. This can be used, for example, to specify a pseudo-terminal if you are using socat to forward serial traffic from a remote host. If unset, or set to "auto", wfview will automatically find directly-connected supported Icom radios on the linux platform. On other platforms, it will attempt to use the first available serial port.
- localAFgain: Unsigned char (0-255) used to store the user's local AF gain (volume). Does not change the volume at the radio, just how loud it sounds at the computer.
- VirtualSerialPort: String path to the virtual pseudo-terminal created for other programs to use as a serial port.

[Server]

These options are for the built-in OEM-compatible server, useful for placing non-ethernet rigs onto a network.

- ServerEnabled: true/false whether to enable wfview's internal server which emulates an Icom rig and allows multiple clients to connect. This is still quite experimental
- ServerControlPort: The UDP port that clients will make an initial connection to.
-

ServerCivPort: The UDP port that the server announces to clients, 50002 is the default.

ServerAudioPort: The UDP port that the server announces to clients, 50003 is the default.

ServerNumUsers: The number of users currently configured.

- ServerUsername_x, ServerPassword_x, serverUserType_x: contain the user information.

[LAN]

These options are for configuring the network access to radios that either have built-in network, are connected to a PC running Icom's software, or are connected to a host running wfview with the above [Server] enabled.

- EnableLAN: true or false to enable LAN connection to the radio (rather than USB)
- AudioInput: Name of audio input device, this is best set via the Settings tab as it must be spelled exactly
- AudioOutput: Name of audio output device, this is best set via the Settings tab as it must be spelled exactly
- AudioRXCodec: This is the receive codec that is sent to the radio, 0x01 = single channel uLaw, 0x02 = single channel 8bit PCM, 0x04 = single channel 16bit PCM, 0x08 is two channel 8bit PCM, 0x10 is two channel 16bit PCM and 0x20 is two channel uLaw. It is anticipated that other modes, potentially including compression will be added at some point for wfview-wfview communications.
- AudioTXCodec: This is the receive codec that is sent to the radio, 0x01 = single channel uLaw, 0x02 = single channel 8bit PCM, 0x04 = single channel 16bit PCM. It is anticipated that other modes, potentially including compression will be added at some point for wfview-wfview communications.
- AudioRXLatency: Use this to specify the number of milliseconds of latency to be allowed in RX.
- AudioTXLatency: Use this to specify the number of milliseconds of latency to be allowed in TX.
- AudioRXSampleRate: This is the sample rate to tell the radio to use for RX audio. 48000 is the default but 8000, 16000 and 24000 are also available. Any other rate will likely be rejected by the radio. Smaller sample rates will reduce the required network bandwidth but may reduce quality.
- AudioTXSampleRate: This is the sample rate to tell the radio to use for TX audio. 48000 is the default but 8000, 16000 and 24000 are also available. Any other rate will likely be rejected by the radio. Smaller sample rates will reduce the required network bandwidth but may reduce quality.
- ClientName: This will usually be the "name" of your computer but can be replaced with any other name. This cannot be changed from within wfview, so must be changed by manually editing the file/registry entry.
- ControlLANPort: This is the "control" port that wfview will use to communicate with the radio and defaults to 50001. Any other required ports are configured by the radio.
- EnableRigCtlID: true or false to enable an emulation of rigctlid within wfview. This is highly experimental and currently only supports a small subset of rigctlid commands. This is disabled by default and cannot be enabled from within the wfview GUI.
- RigCtlPort: Port that the internal rigctlid emulation should listen on. Default is 4533 so as to not conflict with any other rigctlid running on the machine. IPAddress: IP Address of the radio.
- Username: Username configured within Network menu of the radio.
- Password: Password configured for the above user in the radio.
- ResampleQuality: This is a number from 0-10 which determines the quality of wfview's internal audio resampler, with 0 being the lowest quality. This is only used if a sample rate of anything other than 48000 is used and the default of 4 has been chosen to be middle-of-the-road in terms of quality and speed.
- WaterfallFormat: The format of the waterfall data to be provided.
 - 0: do nothing
 - 1: serial-port style (multiple segments)
 - 2: LAN style (single segment)
- tcpServerPort: Port for raw CI-V access over TCP/IP, such as is used by N1MM+. A value of zero disables this feature.

[Memory]

This section contains the 99 memories managed by wfview. Don't bother making a parser for this format, we will change to CSV in future versions.

[Color Presets]

(in progress)

These lines define the color presets a user has entered. The data are read in as strings, and the expected format is #AARRGGBB, where AA is the alpha channel (0xff = opaque, 0x00 = transparent), and RRGGBB is your standard red green blue HTML color. Colors that are fully transparent (AA = 00) will basically hide an element of the interface. PresetName is used for the preset combo box and must be 10 characters or less. Unicode characters should work although this hasn't been tested. The presetNum attribute is automatically assigned and should not be changed.

Any color that is not read in properly is reverted to the default wfview colors.

You can see more about these colors in [this section](#) of the manual.

Example for the first color preset:

```
ColorPreset\1\axisColor=#ffffffff
ColorPreset\1\gridColor=#ff000000
ColorPreset\1\meterAverage=#ff3fb7cd
ColorPreset\1\meterLevel=#ff0000ff
ColorPreset\1\meterLowText=#ffe0f0f1
ColorPreset\1\meterLowerLine=#ffe0f0f1
ColorPreset\1\meterPeakLevel=#ffaaffff
ColorPreset\1\meterPeakScale=#ffff0000
ColorPreset\1\passband=#80006fa7
ColorPreset\1\plotBackground=#ff000000
ColorPreset\1\presetName=Dark
ColorPreset\1\presetNum=0
ColorPreset\1\spectrumFill=#00000000
ColorPreset\1\spectrumLine=#ffffff00
ColorPreset\1\textColor=#ffffffff
ColorPreset\1\tuningLine=#ff55ffff
ColorPreset\1\underlayFill=#96465e96
```

```
ColorPreset\1\underlayLine=#9633ff55
ColorPreset\1\wfAxis=#ffffffff
ColorPreset\1\wfBackground=#ff000000
ColorPreset\1\wfGrid=#00000000
ColorPreset\1\wfText=#ffffffff
```

Using Older Radios

Scope

This guide applies to radios that are not in the core supported radios.

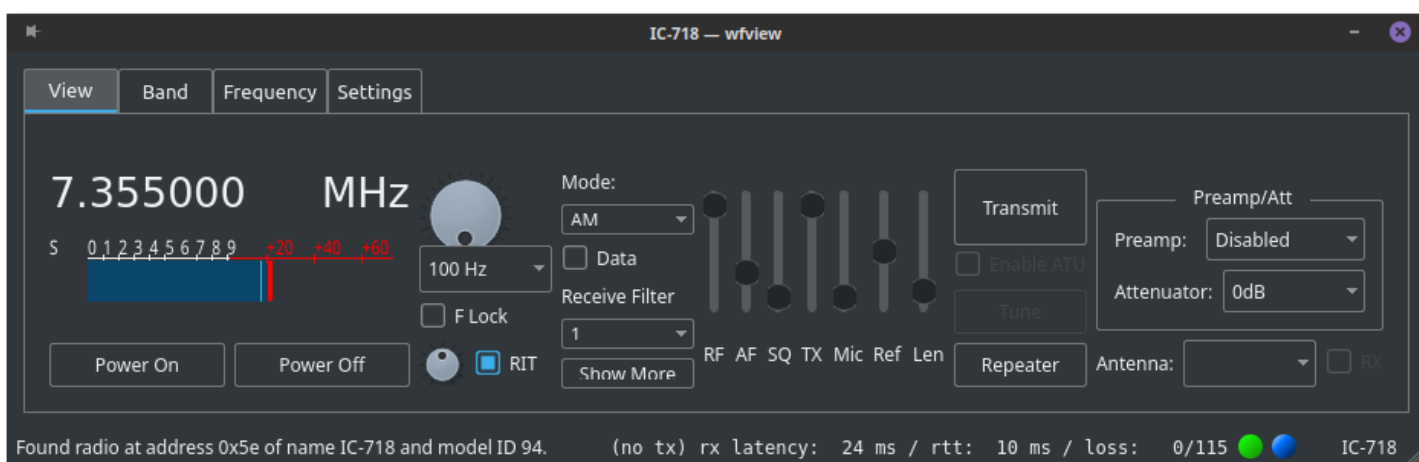
Considerations

Older radios may have several limitations:

- Lack waterfall data
 - wfview gets waterfall data from the rig, so if your rig doesn't have it, or doesn't provide it over CI-V (IC-7700, IC-7600), then this part of the interface will be blank
- Difficult physical connection constraints
 - May need a DIY level converter circuit
 - May need a custom audio interface
 - Some rigs have integrated USB, and this is generally preferred. May require manual baud rate assignment in wfview
- May require manual CI-V address assignment in wfview
- May not answer our rig ID broadcast (discovery) request (Ic-706 and older)
 - You can manually assign the CI-V address in the Settings tab, and then select "Use as model too". Only works if the radio is using the default CI-V address.
- May have unusual metering scale, or no metering at all
- May lack PTT command (IC-780, IC-736, IC-718, IC-706)
 - Note, we have added the capability to use **RTS** (serial port signal "Ready To Send") as a way to PTT
 - For these radios, the usual approach is to use a USB to CI-V converter that includes the ability to toggle the PTT pin on the ACC port by means of the internal serial adapter's RTS pin.
 - These two devices include both audio to USB as well as CI-V and
 - PTT: [XGG Comms Digimode-4-Icom](#)
 - [West Mountain Radio's Rig Blaster-series boxes](#)
- May use non-standard frequency format (IC-713, currently not supported)

Despite these limitations, if you are willing to do some work, you can operate your rig remotely using wfview for the CI-V data and the audio.

Look how beautiful wfview is with a radio such as the Icom IC-718! I am accessing this radio *remotely* too!



Setup

First, it will be wise to launch wfview and set some basic settings:

- Set the serial port to the correct port for the radio.
- Set the baud rate to the highest baud rate the radio reliably works under
- Manually enter the CI-V address of the radio. For example, for the IC-7100, the default CI-V address is “88”. Some radios don’t require this step, in particular newer radios from the past two decades. But it doesn’t hurt if you know the CI-V address to just put it in.
- If the radio does not have a “Read Transceiver ID” command (0x19 0x00), then check the “**Use as Model too**” box. This checkbox causes wfview to identify the radio by the CI-V address you provided, trusting that the radio is set to the default CI-V address, which other radios use as their default CI-V and reply to Rig ID queries. Basically, every radio made prior to 1998 lacks this command. So if you have an IC-736, check this box.

Now press “Save Settings” and then press “Connect”.

On the radio, the recommended settings, when available, are:

- CI-V Echo: off
- CI-V baud rate: maximum supported
- CI-V Transceive: ON
- CI-V address: Use the default value (and note it)

Once these parameters are set, launch wfview. Verify that, for supported radios, the bottom-right corner lists the model number (or Rig ID) and not “NONE”.

If you wish to serve the radio’s control and audio over the network, please see the [Remote Operation](#) guide.

Supported Radios and Features

The following table shows radios the wfview team is often asked about. While we would like to support all these radios, we have to target the areas of highest success, which is generally radios that have a USB connection and have a PTT command. If you want to get your radio supported, send it to us and we'll get it done in 2-3 weeks and send it back. Alternatively, let us know what radio you'd like to help test. We may be able to add better support if we can get good data from your testing.

In the table below, "wfview coded" means that there is code to support the radio within wfview. This means you can expect frequency, mode, s-meter, and where available, PTT (or RTS) and filter selection, to work. Some radios may have additional features supported such as input selection or filter tuning. For a radio like the 706, for example, only the minimum commands will function (frequency and mode and s-meter).

"wfview support candidate" means that there is enough code for this radio (or enough desire for it) that we are actively working on supporting it. Generally, these are radios that the developers own and often use.

Any radio with even the most minimum code currently written should work with the wfview built-in rig server for CI-V control and audio.

Model	USB Audio	PTT Command	Full-Duplex CI-V	S-Meter	Waterfall	wfview coded	wfview support Candidate
IC-703	No	Yes	No	Yes	No	No	No
IC-706	No	No	No	Yes	No	Yes	No
IC-718	No	No	No	Yes	No	Yes	Yes
IC-736/737/738	No	No	No	Yes	No	Yes (736)	Yes
IC-746	No	Yes	No	Yes	No	No	No
IC-746 Pro	No	Yes	No	Yes	No	No	No
IC-756	No	No	No	Yes	No	No	No
IC-756 Pro	No	Yes	No	Yes	No	Yes	No
IC-756 Pro II	No	Yes	No	Yes	No	Yes	No
IC-756 Pro III	No	Yes	No	Yes	No	Yes	No
IC-780/781	No	No	No	No	No	No	No
IC-910H	No	No	No	Yes	No	Yes	No
IC-7000	No	Yes	No	Yes	No	Yes	Yes
IC-7100	Yes	Yes	No	Yes	No	Yes	Yes
IC-7200	Yes	Yes	No	Yes	No	Yes	Yes
IC-7400	Yes	Yes	No	Yes	No	No	Yes
IC-7410	Yes	Yes	Yes	Yes	No	Yes	Yes
IC-9100	Yes	Yes	No	Yes	No	Yes	Yes

Supported Radios and Features

wfview is designed with compatibility in mind. Where possible, commands are used with a rich history dating back sometimes as far as 20 years. And while the more advanced features, such as the waterfall, are only available on newer radios, older radios can still enjoy rapid radio control and remote operation using the built-in server.

The following radios are considered the "core" supported radios for wfview, and thus have the most supported features:

- IC-9700
- IC-R8600
- IC-7850
- IC-7610
- IC-7300

- IC-905
- IC-705
- Xiegu X6100 (beta)

If your radio is not on the above list, please see the chapter for [Using Older Radios](#).

Support for these older radios is coded into wfview. Some radios have more features implemented than others:

- IC-7800 (needs more testing) IC-
 - 7700 (needs more testing) IC-
 - 7600 (needs more testing) IC-
 - 7410
 - IC-7200 (needs more testing) IC-
 - 7100
 - IC-7000 (needs more testing) IC-
 - 703 (PTT using RTS)
 - IC-706 MKGII (PTT using RTS)
 - IC-756
 - IC-756 Pro I/II/III
 - IC-736 (PTT using RTS) IC-
 - 737 (PTT using RTS) IC-
 - 738 (PTT using RTS)
-
- IC-718 (PTT using RTS)
 - IC-706 (PTT using RTS)
 - IC-9100 (needs more testing)
 - IC-910H (needs more testing)

For the most up-to-date list, you can [look in the code](#) and see the currently supported radios. You can also cause an unsupported radio to masquerade as a different model by changing the radio's CI-V address to match a supported radio's default CI-V address, and then checking the "Use CI-V address as Model ID too" box. It's probably best to tell wfview to use the Manual Radio CI-V Address as well. More information can be found on the manual page for the [Radio Access](#) page of the Settings tab.

Remember, even though you may not have as many features as the newer radios, wfview can get these radios to work remotely, for audio and control, with ease.

Supported Radio Features

Provided the radio supports these commands, you can expect the following radio features controllable from wfview:

- Spectrum and Waterfall (core radios only)
- Tuning control
- RIT
- Mode Selection
- Receive Filter selection
- Twin Pass Band Filter adjustment
- Receiver Filter width adjustment
- RF Gain (Receiver)
- AF volume (currently however, network-access instances assign this slider to the local computer audio volume)
- Squelch
- Transmit Power Level
- Mic, USB, ACC, or LAN modulation level (depending upon the source selected)
- Waterfall reference level
- Transmit (via command or RTS on some radios)
- Enable/Disable Automatic Antenna Tuner (“ATU”)
- Begin ATU tuning cycle
- Repeater duplex/simplex selection
- Repeater offset programming
- Repeater tone, both transmit and receive
- Repeater tone selection, both CTCSS and DCS (aka “DTCS”) for supported radios
- DCS inversion
- Split mode programming and activation
- Quick Split
- Main/Sub and A/B VFO functions
- Preamp selection
- Attenuator selection
- Antenna selection
- Bandscope Mode (center, fixed, scroll-c, scroll-f)
- Bandscope Span
- Bandscope Edge selection
- “To Fixed” and “Custom Edge” bandscope edge programming
- Bandscope enable/disable

Band Stacking Register (“BSR”) read access to jump to previous band positions

- Modulation source selection (Mic, ACC, LAN, USB, etc)
- Metering:
 - S-meter
 - Power Output
 - SWR
 - ALC
 - Compression
 - Vd (scale may be incorrect on some radios)
 - Id (scale may be incorrect on some radios)
 - Center Tune (IC-R8600 only)
- Frequency Calibration, fine and course (IC-9700 only) (under [Radio Settings](#)).
- IF-Shift and/or Tunable Pass-Band Filter adjustment (press “Show More” on the View tab)
- Send CW using the radio’s built-in keyer

Headless Server

This page describes how to setup a command-line wfview server. This topic is for **advanced users only**. For most users, the normal [graphical server setup](#) is sufficient.

The following documentation is very “beta” and is not ready yet... but feel free to have a look.

Building the Server

The normal wfview repository should first be cloned. (You can clone from other branches if you wish though — check our [branch list](#).)

```
git clone --depth 1 --recurse-submodules https://gitlab.com/eliggett/wfview.git
```

Next, create a build directory and enter the build directory:

```
mkdir build
cd build
```

Now run qmake. Be careful to specify the correct pro file:

```
qmake ../wfview/wfserver.pro
```

Now run make. You can run make -j4 for a faster build using four cores (or more) if your hardware is up to the task:

```
make -j4
```

Configuring

wfserver uses the same type of [configuration file](#) as wfview, with many of the same options. The default location for the settings file is `~/.config/wfview/wfserver.ini`. The first time wfserver is run, it will create a basic configuration file if one does not exist yet. This is the simplest way to get started.

```
[General]
```

```
AudioSystem=0
```

```
[Radios]
```

```
1\AudioInput=default
```

```
1\AudioOutput=default
```

```
1\ForceRTSasPTT=false
```

```
1\GUID={ee2725b2 - ca44 - 41c7 - 93f5 - d9efa0bc4445}
```

```
1\RigCIVuInt=0
```

```
1\RigName=<NONE>
```

```
1\SerialPortBaud=115200
```

```
1\SerialPortRadio=auto
```

```
1\WaterfallFormat=0
```

```
size=1
```

```
[Server]
```

```
ServerAudioPort=50003
```

```
ServerCivPort=50002
```

```
ServerControlPort=50001
```

```
ServerEnabled=true
```

```
Users\1\Password=ks/\xyzyyw
```

```
Users\1\UserType=0
```

```
Users\1\Username=user
```

Users\size=1

[General]:

- AudioSystem sets the audio system to:
 - 0: QtAudio, default, most compatible and best usual choice
 - 1: Port Audio
 - 2: RT Audio

[Radios] (these start with "1\" since this is for the first radio):

- AudioInput: This is the input to wfview, from the radio's receiver. "default" may work on some installs but it is best to set this variable manually. If possible, use wfview's usual GUI client to see what the audio device would be called for your radio. You can also run wfserver with the debug flag to see a list of audio devices found:
 - `wfserver --debug`
- AudioOutput: This is the output of wfview, used for the radio's transmit audio.
- ForceRTSasPTT: Set this to true if your radio interface needs the RTS pin used for PTT. For radios that have a PTT CI-V command (such as the 7300), set this to false.
- GUID: Internally-generated ID, do not alter
- RigCIVuint: The CI-V address, in unsigned integer, for the radio. Note: If your radio has CI-V Transceive enabled, then you should not need to set this (set it to zero). wfserver will automatically find the radio. For older radios, set the CI-V address here to match the radio.
- RigName: String, radio name used in some debug messages. Recommended format is the same as the model — "IC-7300" for example. If set to "<NONE>", wfview will fill this in with the model number of the radio discovered.
- SerialPortBaud: Set this to the same baud rate as the radio. Higher baud rates are recommended but not always better. For the IC-7300, you must use 115200 for the waterfall to work.
- SerialPortRadio: Set this to the device path for the serial port. "auto" will simply grab `/dev/ttyUSB0`.
- WaterfallFormat: Using this variable, wfview can force the waterfall into the "standard" ethernet or serial-style format. This is useful because some radio control programs expect the waterfall one or the other format.
 - 1: "combined" format, which Icom radios send over network
 - 0: "serial" format (default), which Icom radios send over serial. In this format, a single line of spectrum data is split up into about 11 chunks.
- size: Indicates how many radios are defined. Yes, it may be possible one day to handle several radios from one server. For now, this value should be set to 1.

[Server]:

- Server X Port: These are the three UDP ports that wfserver will use. Remember, do not remap the ports with a router, always keep the ports the same number on both sides of the router.
- ServerEnabled: Set to true to enable the server
- Users and passwords: Set the users and passwords here. Passwords can either be plaintext or "pseudo-encrypted" format. Use wfview to generate them on a GUI client.
- UserType:
 - 0 = Full User, has transmit capability
 - 1 = Full User with No Transmit
 - 2 = Monitor only, no controls

Audio Levels

You can adjust the input (capture) levels in the terminal using alsamixer and amixer. Be careful, sometimes there are adjustments that merely scale the level and then there are adjustments that actually set the ADC conversion gain. We want the conversion gain.

To bring up the capture levels for card #1, use this command (try card #0 and #2 as well until you see the controls you expect):

```
alsamixer -c 1 --view=capture
```

Once adjusted, make a note of the adjustment level. You can then script amixer to make this adjustment for you just prior to starting wfserver using a startup script:

~~amixer --set-volume=33 -c 1 --device="capture"~~ (note this is wrong, I need to [figure](#) this one out)

Running Automatically

You can create a systemd script to automatically run wfserver on startup. **DO NOT EVER run wfserver as root.** This is entirely unnecessary and not supported at all. If you have permission errors, try and dig to the root of the issues (ie, serial port ownership or group membership, typically).

To run wfserver in a terminal with persistence, for now, it is recommended to use the “screen” session manager program. Simply hit control-a, d, to detach the screen session and keep wfserver running. We will add a background fork mechanism later.