Syncing SDR Console with WSJT-X Using the DxLab Commander Program (or Not)

John Price – WA2FZW
November 22, 2022

Table of Contents

Introduction

A few weeks ago, I watched a video of a Zoom call of the Front Range 6 Meter group on using SDRs along with a transceiver. The approach looked interesting, but the video lacked a lot of details about how to make it all work.

It didn't take long to figure out how to make *SDR Console* and *WSJT-X* work together, but I use the DxLab *Commander* program to run my 6 meter radio which is a Yaesu FT-891. That was a bit more complicated! It took several hours of reading the documentation for all three programs before I figured it all out.

So, here I'll attempt to walk the reader through the setup step by step.

Infrastructure

Other Required Programs

Besides the three main programs, you need three other programs:

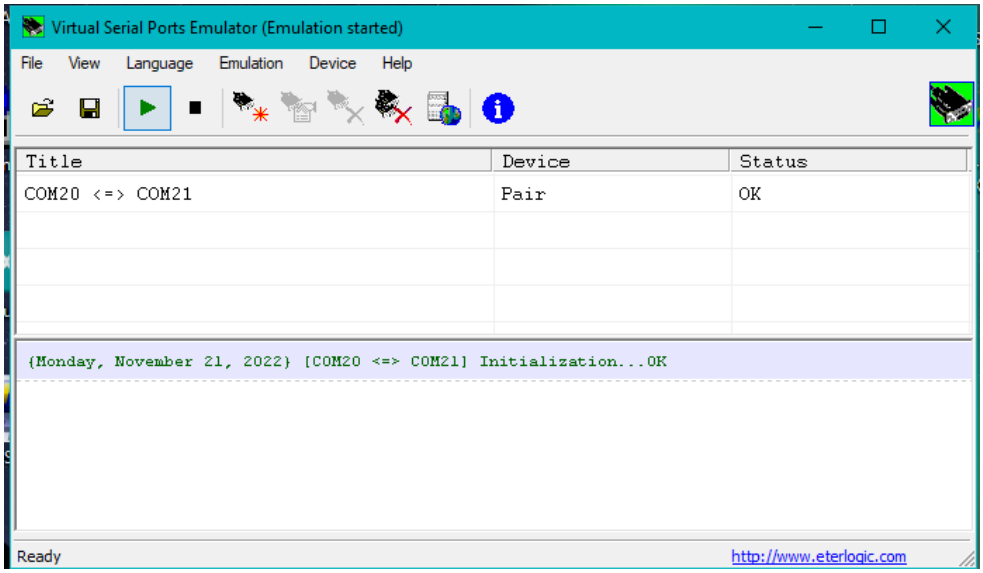
- You need a program that can set up virtual COM ports. Two choices are *com0com* (freebie) and *VSPE* (costs a few bucks). I use *VSPE* as it can do some other tricks that I've used in other projects.
- You need a program that can implement virtual audio cables. Two choices are *VB-Cable* (freebie) and *Virtual Audio Cable* (you need the paid-for version).
- *SDR Console* uses the *OmniRig* program to communicate with real radios. In the scheme of things here, the *Commander* program looks like a radio to *SDR Console*.

Download and install OmniRig and whichever of the other 2 programs you choose.

Virtual COM Port Setup

Using whichever program you chose to use you need to setup a virtual pair of COM ports. In my case, I paired COM20 and COM21; neither of which are real COM ports on my computer.

Here's what the *VSPE* setup looks like:

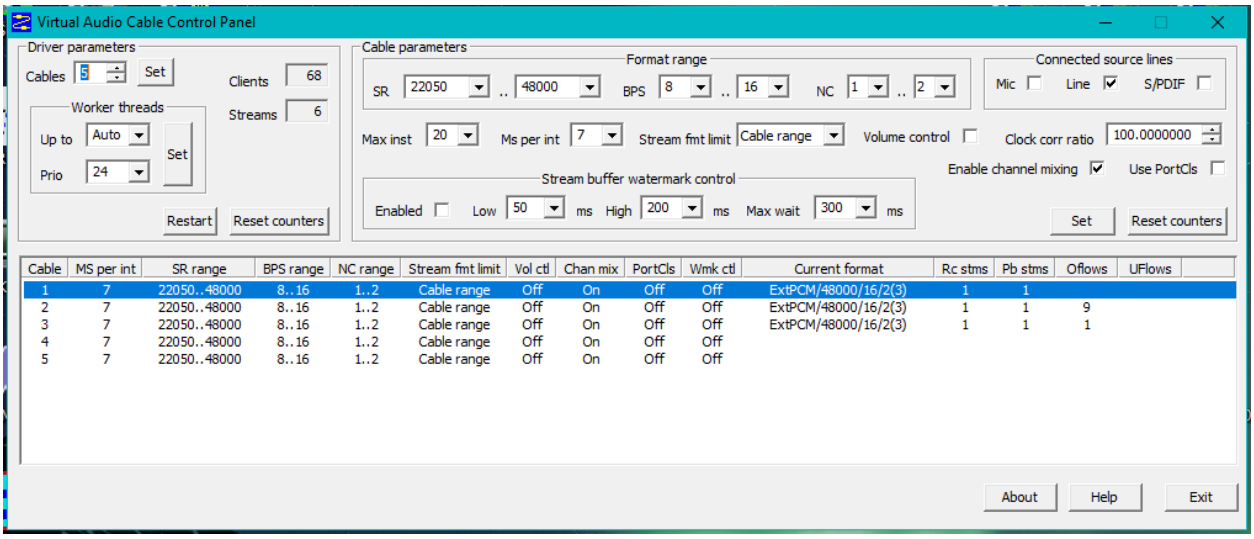


You can save the setup and include the saved setup in your Windows startup menu so it will be initiated automagically whenever the computer is booted.

I don't have a copy of *com0com*, so I can't show you what that setup looks like.

Virtual Audio Cable Setup

I created 5 virtual audio cables using the *Virtual Audio Cable* program. Here's what that looks like:



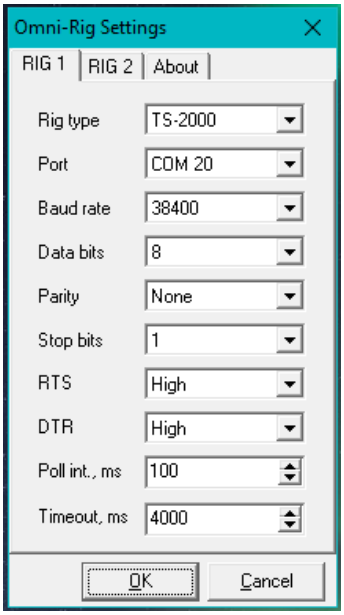
Again, I don't have the *VB-Audio* program, so I can't show you that setup.

Once the VACs are initiated, they will always be there; no need to reinitiate them when the computer is booted. They will also show up in the Windows control panel 'Sound' manager.

OmniRig Setup

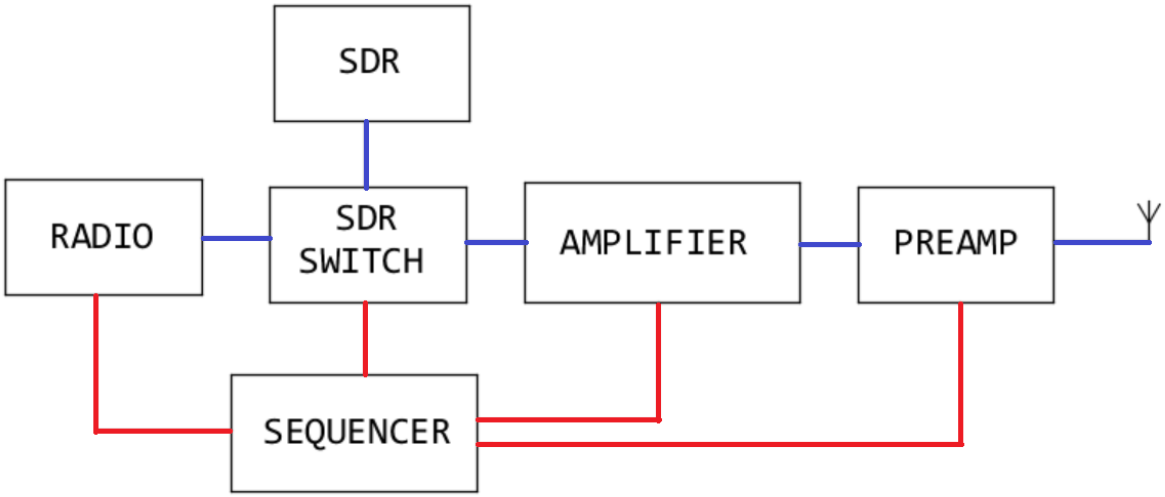
OmniRig comes with a collection of '.ini' files for all the radios that it can work with.

When installed, open it and on the ‘Rig 1’ tab set the ‘Rig Type’ to ‘TS-2000’. Set the ‘Port’ to one end of the virtual COM port pair you already set up; I used COM20. Set the ‘Baud rate’ however you like. The rest of the parameters as I have them set seem to work:



Hardware Configuration

Here’s a block diagram of how I have the hardware in my 6 meter system configured:



The blue lines are RF paths and the red lines are control lines. The sequencer I use is of my own design. It is different than most sequencers in that it can control the SDR switch in addition to the preamplifier and amplifier. It is software controlled using an Arduino Nano (or a cheaper clone).

I was initially using an SDR Play RSP-2 SDR, which is no longer available. It has been replaced by the RSPdx. I'm now using one of those.

I have 3 different SDR switches. One is made by Paul (N2EME) and is unique in that it allows simultaneous reception on both the transceiver and SDR at the same time. Paul's switch is described in the video referenced in the *Introduction*.

The second switch is from Amazon and it allows reception on either the radio or the SDR, but not on both simultaneously.

The third switch is of my own design, and unlike the others. It has the splitter built in so both the radio and SDR can receive from the same antenna.

The operation is simple; when the radio's PTT goes active, the SDR switch and the preamp are immediately switched into transmit (bypass) mode. After a brief delay (adjustable in the sequencer software) the linear amplifier is keyed.

Commander Setup

In Commander, click the 'Config' button and go to the 'Ports' tab.

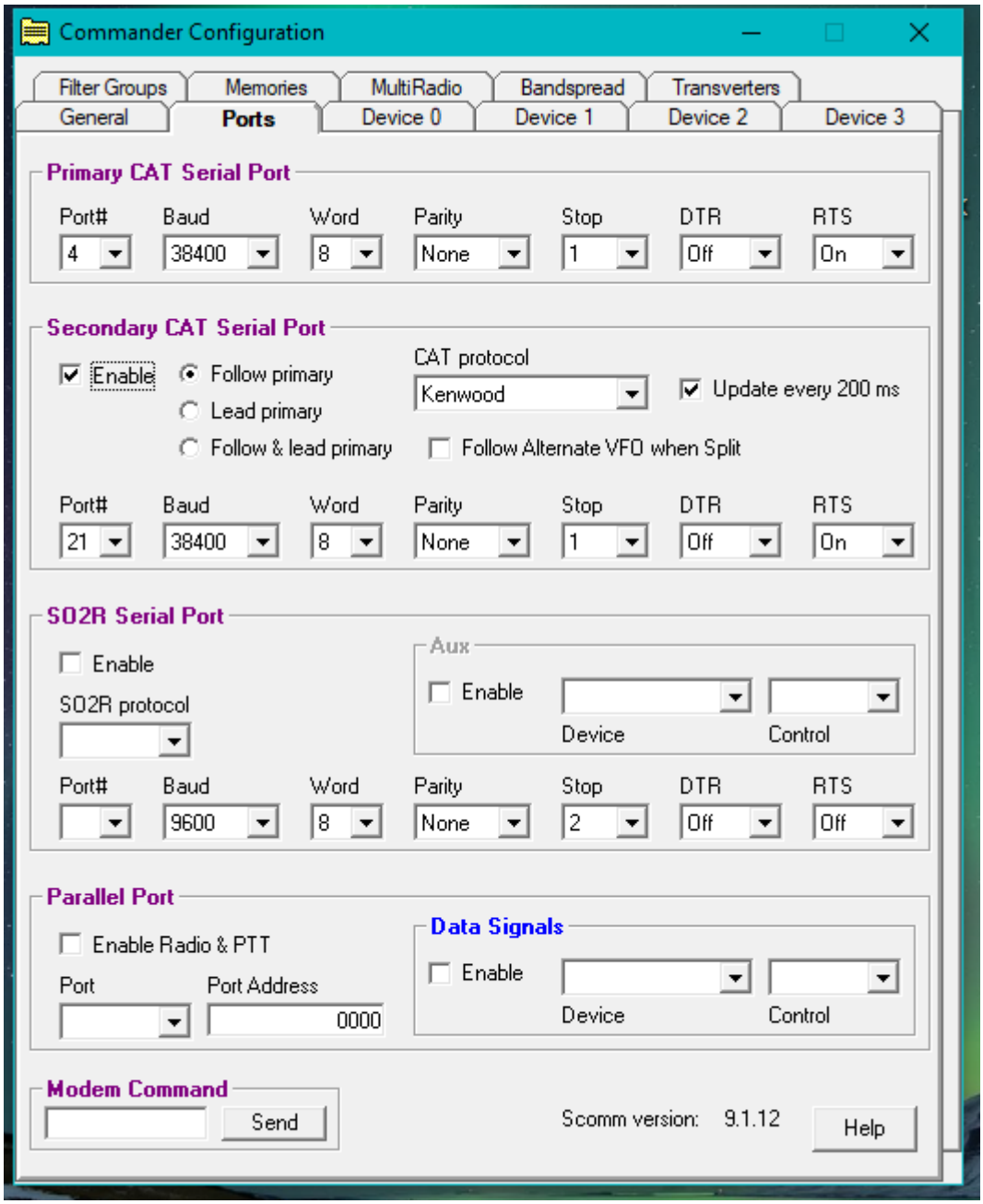
In the 'Secondary CAT Serial Port' section:

- Set the 'CAT Protocol' to 'Kenwood'.
- Select 'Follow Primary' (Haven't played with other options yet)
- Set the 'Port#' to one end of the virtual COM port pair (I set mine to '21'.
- Set the 'Baud' to the same speed that you set in *OmniRig*.
- Enable the secondary CAT port.

I selected 'Follow Primary' which means *SDR Console* will track whatever you do in *Commander* or on the radio itself; I haven't played with the other options yet to see if they might be better choices.

I also checked the ‘Update every 200 ms’ box; just seemed like a good idea!

Here’s what that looks like:



The ‘Primary CAT Serial Port’ has the proper settings for my FT-891.

SDR Console Setup

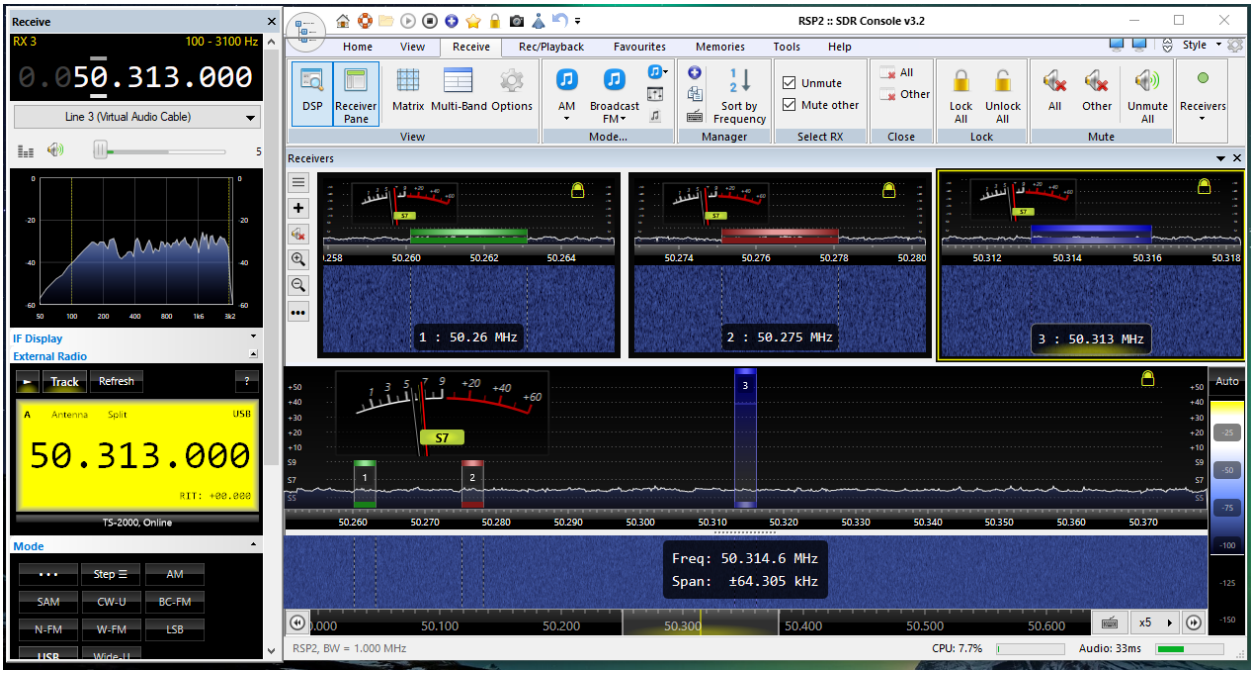
I’m running version 3.2, so everything here may or may not be accurate for other versions. I’m assuming that you already have *SDR Console* working with your SDR; if not, follow the *SDR Console* instructions to get that much working first.

This is a bit more complicated than everything else:

1. Click on the 'Home' tab then click the 'Stop' Button to stop the SDR receiver(s).
2. If the ‘DSP’ panel isn’t already displayed, click on the 'Receive' tab then click in the ‘DSP button to turn it on.
3. If an 'External Radio' is already enabled, there will be a yellow panel in the DSP panel; if not:

    - Click on the 'View' tab.
    - Click on the dropdown arrow in the 'More options' button and select 'External Radio'; you'll get a box that says the program needs to restart to show the option; click 'Yes'.
    - When the program restarts, the 'External Radio' panel in the DSP panel should show 'TS-2000' at the bottom of the 'External Radio' panel.
    - If it says 'Not Tracking', click on the 'Track' button. (I found that sometimes I had to tweak the radio's VFO to get it to start tracking).

Here's what the *SDR Console* window will look like (more or less):



I said "more or less" as I have 3 receivers enabled. The 3 receivers in the example are listening to the segments of the 6 meter band used for meteor scatter (MSK144), Q65 and FT-8. *Commander* and the radio are currently monitoring the FT-8 band segment. The ability to do this was one of my primary reasons for wanting to use the SDR.

WSJT-X Setup

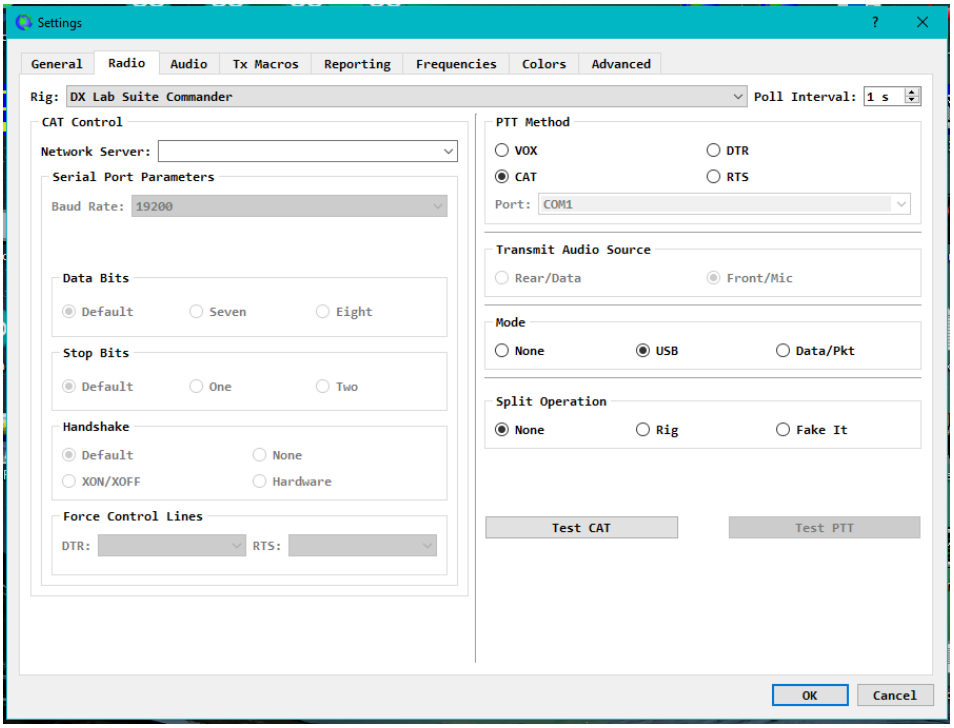
In the system configuration for my 6 meter station that I showed above, one can actually run multiple instances of *WSJT-X*. One instance listens to the transceiver and is used to actually transmit.

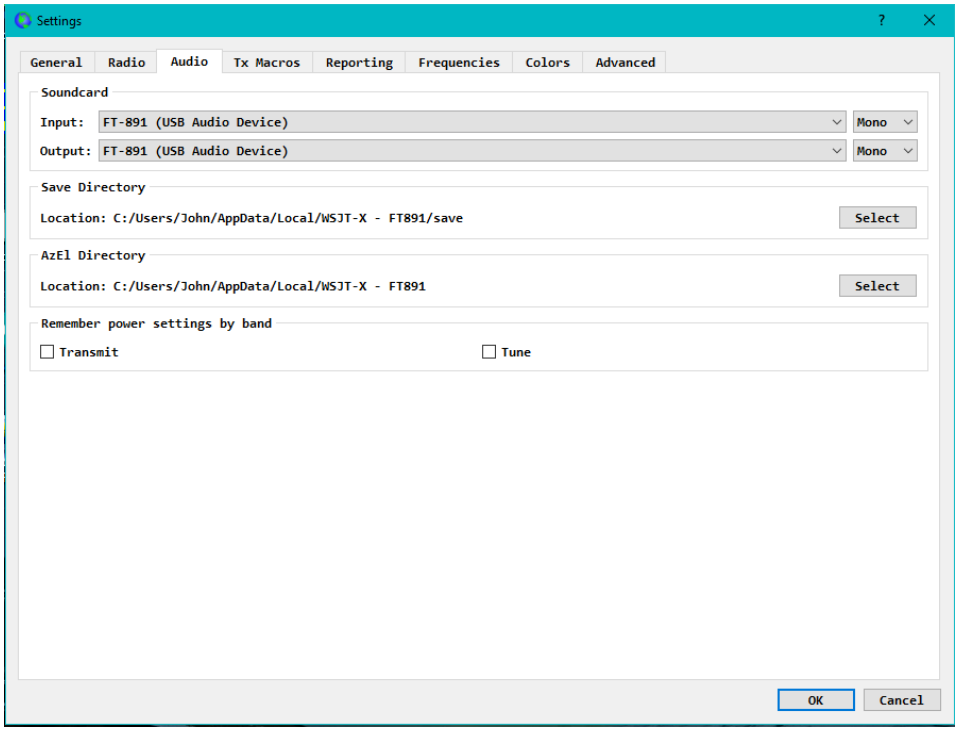
The other instance(s) only listen to the SDR receiver(s). The *WSJT-X* documentation explains the procedure for running multiple instances of the program.

They are set up a bit differently than the instance running the transceiver.

WSJT-X Transceiver Control Instance

The *WSJT-X* instance that controls the transceiver is set up as one would normally set it up to use *Commander* when not using the SDR in addition to the radio:





In the ‘Radio’ settings, select ‘Dx Lab Suite Commander’ as the ‘Rig’ and set the appropriate ‘PTT Method’ for your radio.

Notice the audio devices selected are named ‘FT-891 (USB Audio Device)’. In Windows, using the ‘Sound’ options in the Control Panel, you can rename the audio devices and even change the icon used to represent them. I’ve found doing so saves a lot of confusion when trying to remember which is which.

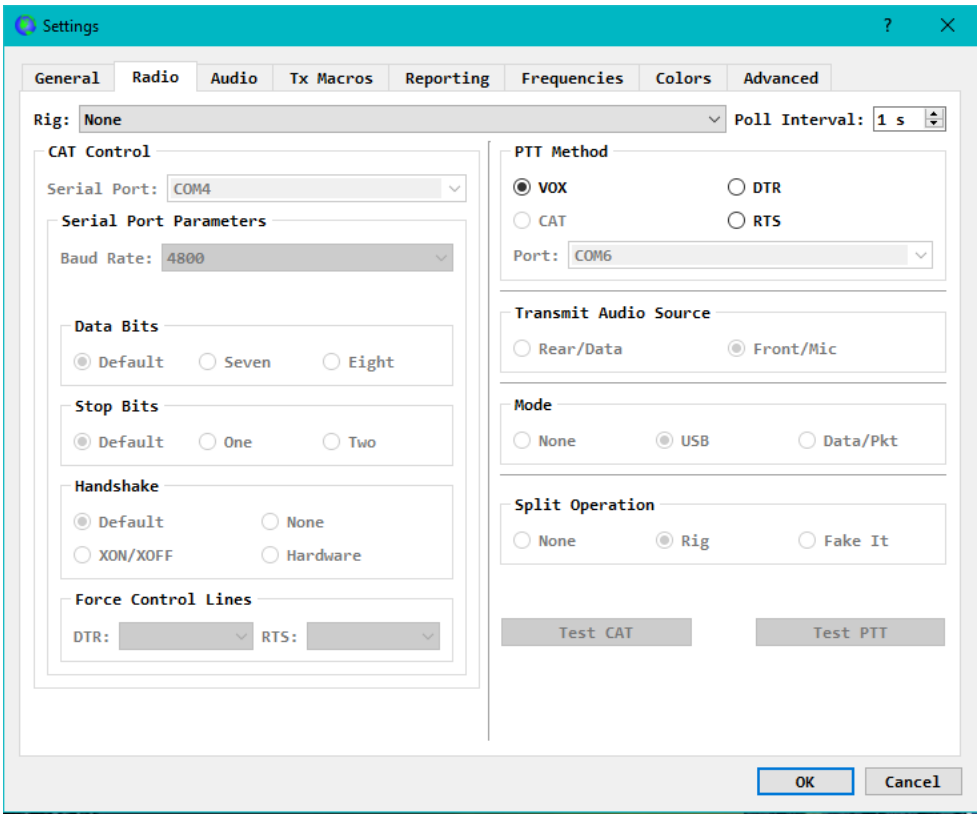
When making or answering calls, you use this instance to control the transceiver. If you want to answer a call that decoded in this instance of *WSJT-X* or you want to call CQ, do so using this instance.

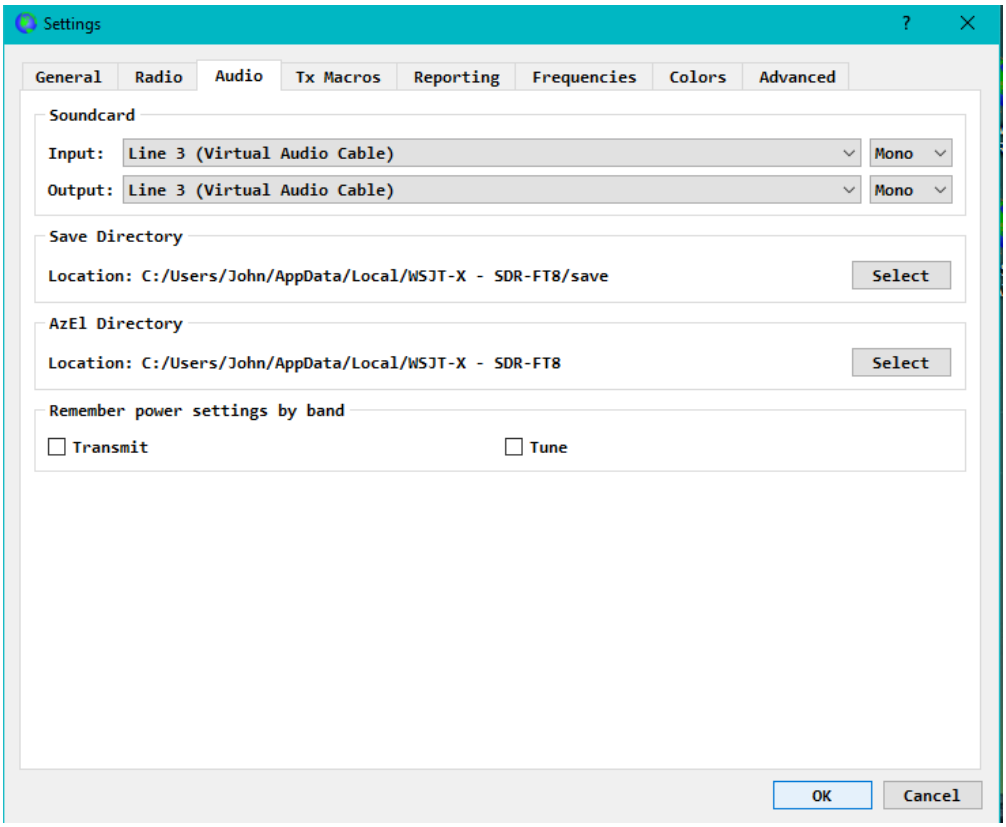
If, however, you get a decode in the *WSJT-X* instance listening to the SDR but not in the instance running the radio, you’ll have to copy the call into this instance, click the button to generate the appropriate messages and proceed from there.

In my preliminary testing (mostly using MSK144) I’ve found that I do get decodes on the SDR but not on the radio and if I get decodes on both simultaneously, the ones from the SDR are generally stronger!

WSJT-X SDR Listening Instance(s)

The *WSJT-X* instance(s) that monitor the SDR receiver(s) are set up like this:





The key differences are that the ‘Rig’ is set to ‘None’ and the ‘Soundcard’ input and output are both set to the appropriate virtual audio cable. The ‘Output’ setting doesn’t matter as you never transmit using the SDR instance(s) of *WSJT-X*.

WSJT-X Configuration without Commander

It is also possible to simply use one instance of *WSJT-X* that listens to the SDR but transmits on the radio. This approach is useful if your setup doesn’t have the ability to receive on both the radio and the SDR at the same time.

To set things up for this type of operation:

- In the ‘Radio’ settings tab, set the ‘Rig’ to the specific model transceiver you are using (FT-891 in my case).
- In the ‘Audio’ tab, set the ‘Input’ to the appropriate virtual audio cable.

- In the ‘Audio’ tab, set the ‘Output’ to the sound device for the transceiver; in my case, this would be ‘FT-891 (USB Audio Device)’.

Issues

There is one issue to be aware of when using *Commander* to run things. *Commander* does not send transmit and receive commands to the secondary CAT port. The *DxLab* author feels that doing so could cause someone to blow up their primary transceiver if a real radio connected to the secondary port was to be set to transmit mode.

While I don’t particularly agree with his reasoning, I do understand his concern.

In my case, both SDR switches that I’m using have front end protection by grounding the SDR input when the switch is in transmit mode so this hasn’t been a problem for me. If you are using a switch that doesn’t protect the SDR front end you could potentially blow up your SDR.

DxEngineering has a gimmick that is designed to protect a receiver’s front end. I know of one person using this in a configuration where his SDR is connected to a separate antenna than his transceiver (and 700W amplifier) without a switching device. He hasn’t had any problems with the SDR.

There are cheaper front end protectors on eBay; some of these are discussed in the video I referenced in the *Introduction*.

If you’re using the SDR without *Commander* in the mix as described above, *SDR Console* will get the TX/RX messages and handle them properly.

Suggestion Box

I welcome any suggestions for further improvements. Please feel free to email me at WA2FZW@ARRL.net.