The Big Book of SDR# Studio

(...to be used with "frequency")

Paolo Romani IZ1MLL

2024

DAYTON 2024 HAMVENTION

RANGER AIRSPY Communications Receiver

✓ Aligned to v1.0.0.1920

Introduction

ITALIAN STYLE

Much of my energies and passions for SDR could not be dissipated, but rather gathered and reasoned out since the energy released from each and every word has the power to redraw the map of everything we know and it is unspeakable fun for me to unravel these threads and wind them onto the spindle. In fact, this book was born from here: this book was born here: from the giddiness of exploring and disseminating the power of SDRSharp because today, more than ever, our world is SDR (Software-Defined Radio). Since a work like this did not exist, I decided to write myself a small guide in the beginning and a big book now. A rich word map for all SWL friends, radioamateurs, specialists or just curious.

Throughout the years I have realized that the world of SDR is often surrounded by an aura that discourages many people even of good intentions and longtime radio enthusiasts. But it doesn't have to be that way, and with this work I hope to show you otherwise. Every SDR should be technically unimpeachable, understandable, aesthetically beautiful and immediately usable by everyone besides of course being as fun and satisfying precisely as turning the VFO of a normal radio but with a thousand more possibilities. Over time I have tried many, perhaps all of them on the market (even for various OSs) are often either very complicated and boring to use or great for one reason or another but too impractical in daily use... What is reported in the following pages is the fruit of my years of listening, dedication, passion and a great deal of personal effort in the search for the best possible configurations and optimizations as well as in the operational suggestions *that I have collected and highlighted typographically in italic blue color.*
Enjoy reading and good listening as when we turn on our SDR we will be able to easily understand that this world really has many faces but one heart. Be ready to learn together the new way of radio listening!

SDRSharp (or SDR#) is the most complete, high-performance, integrated, continuously updated and customizable (with plugins for every need) FREEWARE software for all RTL-SDR dongles and of course the highest performing AIRSPY devices.

Visit to update it freely: **https://Airspy.com/**

Note: due to evolutions in the development of SDR# and various third-party software, some illustrations, indications or comments, despite my constant updates, may slightly differ from the current versions on the net.

ACKNOWLEDGEMENTS

Heartfelt thanks to Youssef Touil and all those who interact with SDR# on a daily basis, and there are indeed many as I have witnessed over the years, because it is a common learning and growing experience: on your own you will get nowhere...

I would like to thank friends and colleagues radioamateurs whom I have mentioned as I have gone along, having made valuable contributions in the production of some chapters of the work, and of course to the native translators who have made themselves available so far:

French - Philippe Diaz & Michel (F1AUX)
German - Gerhard Schweizer
Russian - Dmitry Mezin
Spanish - Miguel Iborra (EA4BAS)
Ukrainian - Rostyslav "Rost" Iablonskyi (UT2YR)

At this point I am launching a request for missing languages: Chinese, Japanese, Portuguese, etc. Those who want to come forward are of course welcome! :-)

SDR# download & installation

The main thing to know is that even the most inexperienced user can easily start with SDR# and successfully even with the most sophisticated plugins... So let's see how to start using the software starting from the installation.

In fact, since there is NO real installation procedure, you only need to remember this:

- *Extract the zipped content into any directory **(obviously excluding only "C:\program files" and "C:\program files (x86)" !!!)***
- *All the necessary files are in the previous directory and nothing in the Window registry.*
- *Plugins must be inserted in the relative subdirectory and are automatically recognized.*
- *For "quick" updates, it is suggested that you keep your Config files (which contain the various customizations) and replace only the binary EXE and DLL files.*

Same for the uninstallation... to delete the software it is sufficient to delete the directory where it resides since no other dependency and/or registry key is used. Once started, SDR# resides in memory with a small active set and little to no swap will be required.

SDRsharp is a software in continuous and perpetual search for improvement and refinement. Many releases are completely different from the previous ones, even though they use the same configuration files, plugins, Band Plan and memory files, but always with better overall performance. ***For the chronology of the single versions, see the appropriate chapter "SDRsharp history".***

.NET 8 Microsoft *(current)*	April 25, 2024: Finally after much development work and subsequent testing, v1920 was officially released. The software has been ported to **Microsoft's brand new .NET 8** and the Telerik framework has also been dropped in favor of Open Source libraries with tulle user interface features similar to the previous Telerik plus many, many fixes and improvements across the board. *The most important aspect is the jump in performance, a direct consequence of abandoning the previous interface in favor of a set of specific high-performance controls. It is believed that resource utilization is currently the lowest for equivalent if not better DSP quality and user interface!* The new SDR# focuses on usability, accuracy, the highest performance and low power consumption. Many DSP blocks have been updated with the latest algorithmic refinements, such as FFT display, filtering, noise reduction, channel cancellation, and Multi-Notch. User mode drivers have also been improved to ensure greater reliability and resistance to USB problems. Preliminary support for high-performance radios from AOR Ltd's Japanese partners and distributors has been introduced. If you have an AR-5700D with an IQ module, for example, you can take further advantage of it with SDR# and its state-of-the-art DSP. The list of supported AOR radios: AR5700D, AR2300IQ, AR5001D with IQ5001 option AR6000 with IQ5001 option. The current community package installer has been updated with a list of plugins guaranteed to load correctly: IF Processor (completely rewritten in Multi-Notch); NetRemote (did not work properly with UI blocking due to improper use of blocking sockets within the UI thread); Audio UDP Streamer (Same multi-threading problems as NetRemote, as well as referencing dlls that no longer exist); Tetra Demodulator plugin has been patched again so that it can be used with profiles created by the community installer. Perhaps there are other plugins that need some patches, so I encourage the authors to check SDR# v1920 to fix what is needed. **https://Airspy.com/?ddownload=3130**

.NET 7 Microsoft	November 2022: upgrade to Microsoft's **NET 7**. Previously v1832 introduced Microsoft's brand **.NET 6**, a platform that combines the .NET Framework and .NET Core, which is increasingly aimed at cross-platform software developers. In fact, the idea is to have a single .NET framework to be used on Windows, Linux, macOS, Android, etc. etc.

.NET 5 Microsoft	Version 1785, officially released on 5 February 2021, has made a big leap towards Microsoft's **.NET 5**. This multi-system, open source development platform is capable of supporting side-by-side execution without the need to install the runtime. This is not a simple code recompilation effort but involves a lot of changes, some superficial and some fundamental! *Even externally you can see the difference with far fewer files in the distribution and a large executable file. There are far fewer DLLs that shorten the start-up sequence of the program. The **Telerik framework** enables the dinamic window management.*

.NET 4.x Microsoft	Previously, the graphical user interface developed in Visual Studio with fully customisable layouts was released at the end of November 2020.

v1777	Latest version with collapsible panels.
v1716	Latest version unskinned build.

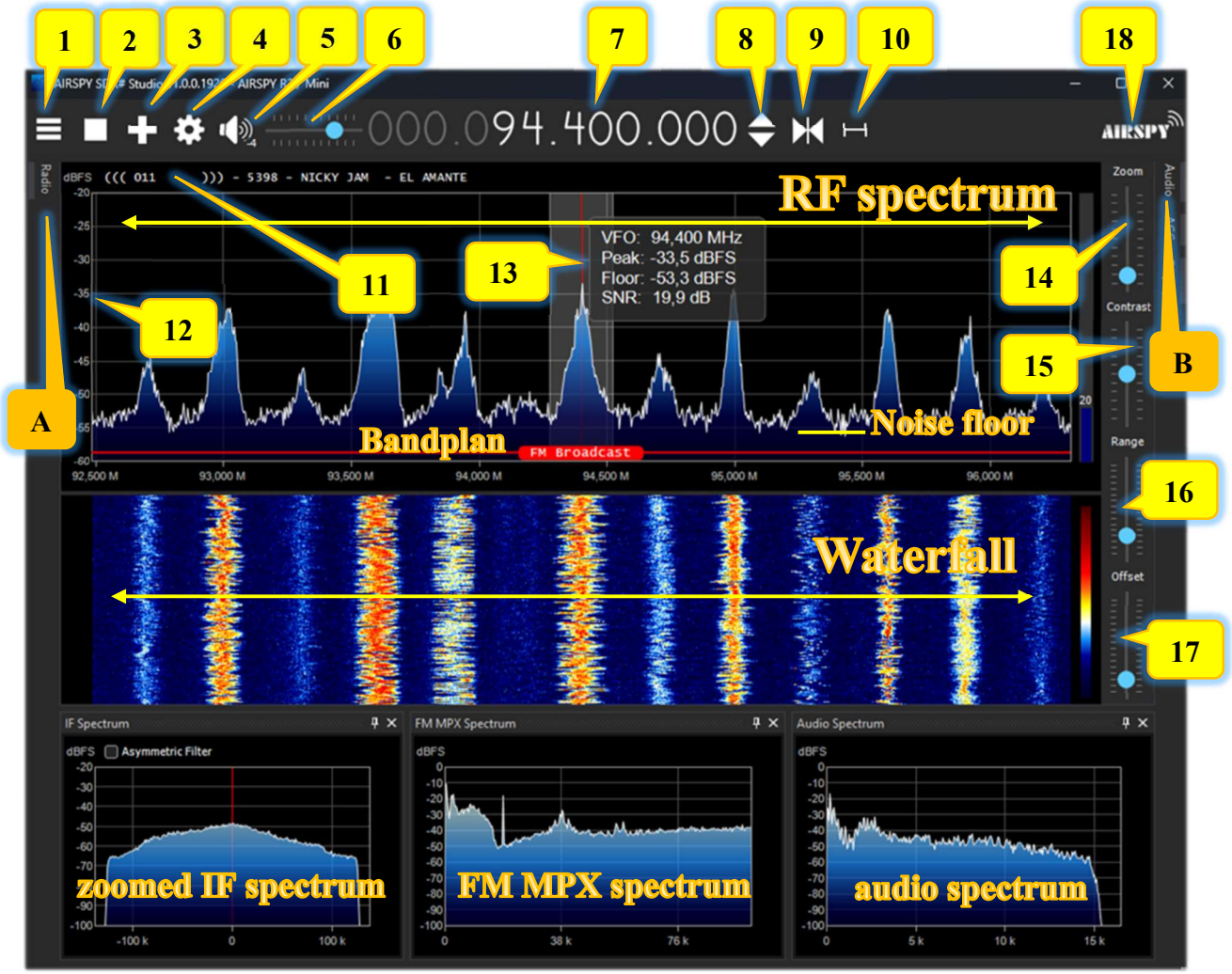
Normally Airspy is a plug-and-play device that Windows (Vista to W10) automatically detects and recognises when plugged into a USB port. If this does not happen, you can download, unpack and install the following driver from the Windows device manager:

https://Airspy.com/?ddownload=3120

The screens will present the dongles RTL-SDR and all the various devices Airspy (but little change for the other devices if not the configuration menu and the bandwidths/decimations used). The graphic theme used in this guide (skin) is the dark one named "Fluent Dark" (selectable in the Display menu).

Obviously, since these are radio signals that can extend from long waves to the GHz of UHF, it is advisable to equip oneself with specific antennas (for HF: Youloop, vertical, wire, while for V-UHF: discone or collinear) to be installed outdoors and as far away as possible from other elements that can attenuate or interfere with the signals...

Main screen



These in detail the main points and *my tips*:

A. Left Menu (example: Radio, Source, various plugins)
B. Right Menu (example: Audio, AGC, Display, various plugins)
1. Main Menu *(in jargon as "hamburger menu")*
2. Start/close the program
3. Opening new session (slice)
4. Device configuration
5. Audio On/Off (mute)
6. Volume control bar
7. VFO frequency
8. VFO increase/decrease buttons
9. Tuning type buttons
10. Step bar
11. RDS decode (PS, PI, RT) for broadcaster stations in WFM *(88-108 MHz)*
12. Signal scale in dBFS (decibel Full Scale)
13. Vertical tuning bar *(center red line, bandwidth and signal info)*
14. Zoom bar for RF Spectrum and Waterfall
15. Contrast bar
16. Range bar
17. Offset bar
18. Airspy logo *(click above to visit the home page directly)*

Airspy family

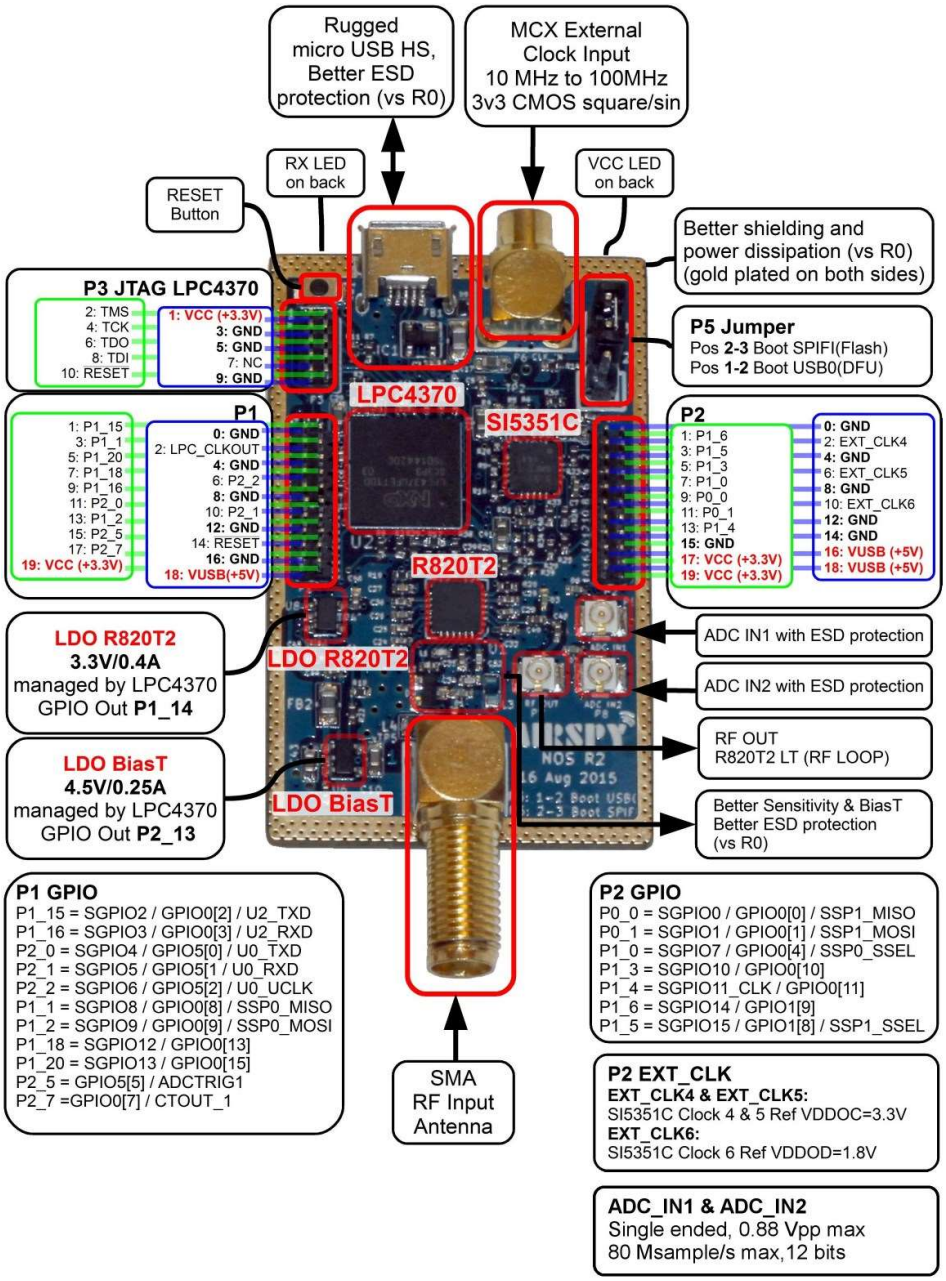
HF **VHF** **UHF**		**AIRSPY RANGER** (*) 0.5 kHz / 1750 MHz continuous coverage (double SMA input)
HF **VHF**		**AIRSPY HF+ Discovery** HF 0.5 kHz / 31 MHz VHF 60/260 MHz (single SMA input)
HF **VHF**		**AIRSPY HF+ Dual port** HF 9 kHz / 31 MHz VHF 60/260 MHz (double SMA input)
VHF **UHF**		**AIRSPY R2** 10 o 2.5 MSPS IQ continuous coverage 24/1700 MHz
VHF **UHF**		**AIRSPY Mini** 6 o 3 MSPS IQ continuous coverage 24/1700 MHz
HF		**SpyVerter** in combination with R2 / Mini increases coverage 1 kHz / 60 MHz

(*) Previewed at the Dayton USA Hamvention on May 18/19, 2024 these the technical specifications:

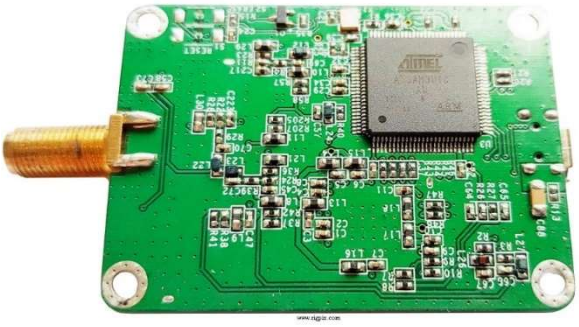
Technical Specifications

- Frequency Coverage: 0.5 kHz - 1.750 GHz
  - HF: 0.5 kHz - 31 MHz
  - VHF-I: 31 - 66 MHz
  - VHF-II: 66 - 118 MHz
  - VHF-III: 118 - 260MHz
  - UHF: 260 - 1750 MHz
- Sensitivity (MDS @ 500 Hz BW)
  - HF: -141.0 dBm Typ.
  - VHF/UHF: -142.5 dBm Typ.
- Linearity (at maximum gain)
  - HF: +15 dBm IIP3
  - VHF: +5 dBm IIP3
  - UHF: 0 dBm IIP3
- Blocking Dynamic Range (BDR)
  - HF: 110 dB
  - VHF/UHF: 95 dB
- Reciprocal Mixing Dynamic Range (RMDR)
  - HF: 89 dB at 20 kHz separation
  - VHF: 82 dB at 50 kHz separation
  - UHF: 74 dB at 50 kHz separation
- Input Impedance
  - VHF/UHF: 50 ohms
  - VLF/LF/HF: 1k ohms
- Selectivity
  - RF Preselectors
  - Polyphase Harmonic Rejection Mixer
  - IF Filters
  - Sigma Delta Modulator
  - DDC
- Image Rejection
  - HF/FM: 120 dB (Zero-IF)
  - VHF/UHF: 75+ dB (Low-IF)
- Architecture
  - HF/FM Broadcast: Direct Conversion (Harmonic Rejection Zero-IF)
  - VHF/UHF: Double-Conversion (RF to Low-IF to Zero-IF)
- RF Filtering
  - HF: 5 Preselection Filters
  - VHF/UHF: 5 Preselection Filters
- Gain Control
  - Smart AGC with real time optimization of the gain distribution
  - User controlled Attenuators with 4 dB steps and 0 – 28 dB range
- Bias-Tee
  - User controllable for HF and VHF/UHF
  - HF Bias-Tee bypass switch for the best VLF sensitivity
- Connectors
  - HF: SMA (+15 dBm Max)
  - VHF/UHF: SMA (+15 dBm Max)
  - Ref. Clock: MCX (10 MHz)
  - USB: Type C
- ESD/RF protection with double BAV99 and fast TVS diodes in the RF ports
- ESD protection with USB TVS chip in the USB port
- Up to 710 kHz alias and image free output for 912 KSPS IQ
- High Dynamic Range HF/IF circuit based on 2 x Sigma Delta ADCs @ 36 MSPS followed by a Digital Down Converter (18-bit DDC) and controlled by the Core AGC
- 0.5 ppm high stability, low phase noise VC-TCXO with software-controlled DAC
- Low phase noise PLL (-110 dBc/Hz @ 1kHz separation @ 100 MHz)
- No drivers required - 100% Plug-and-play on 10 and 11. Open-source user mode driver for Linux and other platforms.
- Operating Temperature: -20°C to +48°C
- Ultra small Form Factor with RF tight CNC enclosure

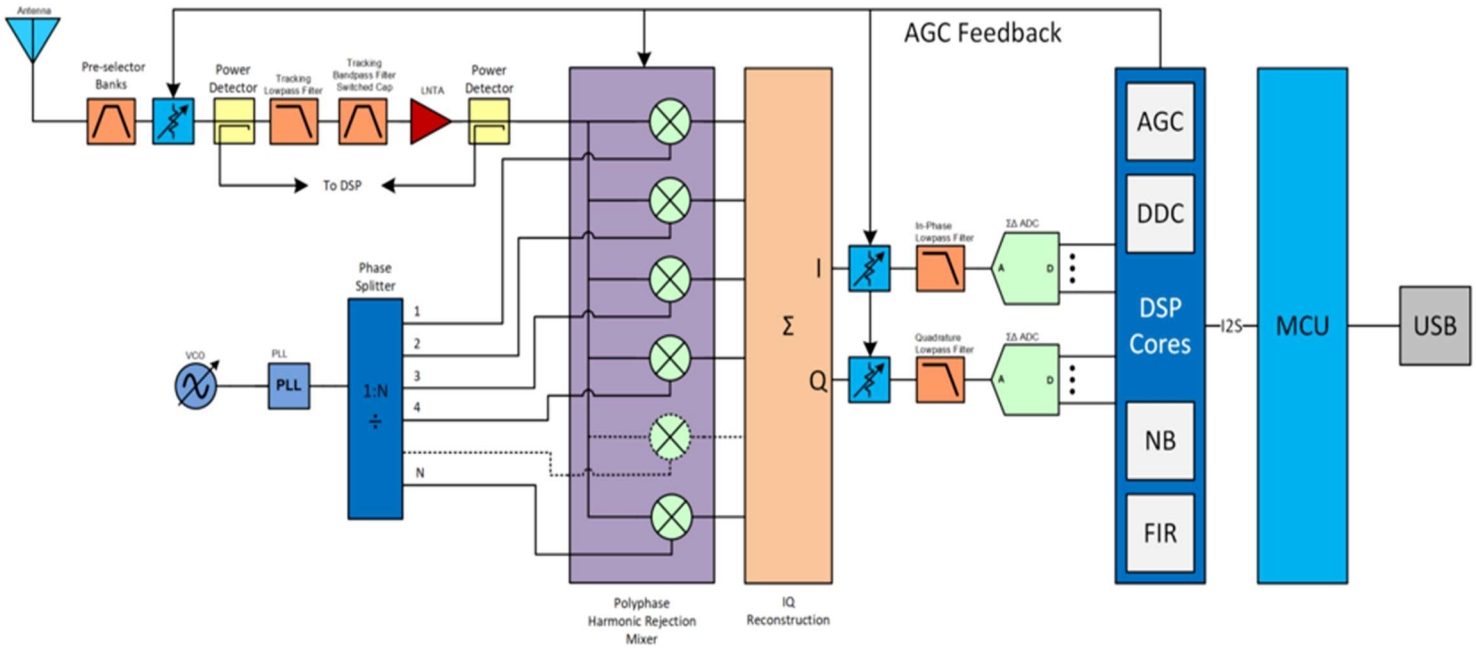
But what, for example, is inside the Airspy R2? We can here take a look at the inside...



While this is the inside of an **HF+ Discovery** thanks to the excellent images from the site: ***https://www.rigpix.com***



This is an opportunity to recall how the front-end of the HF+ series was designed.



Here where everything is born!!! at the ITEAD itead.cc

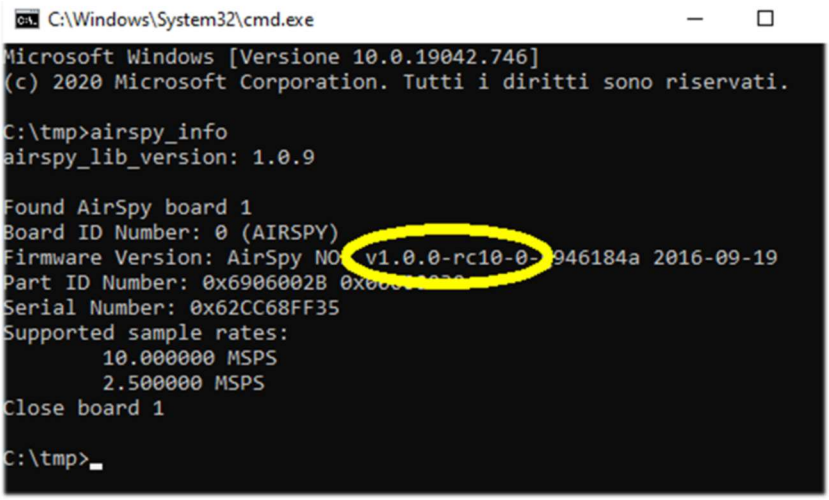
Airspy R2 / Mini firmware upgrade

Unlike the Source panel for HF+ devices, here there is no indication of the firmware installed. To check the your firmware it is necessary to use the "AIRSPY HOST TOOL", downloadable here:

https://github.com/Airspy/Airspyone_host/releases

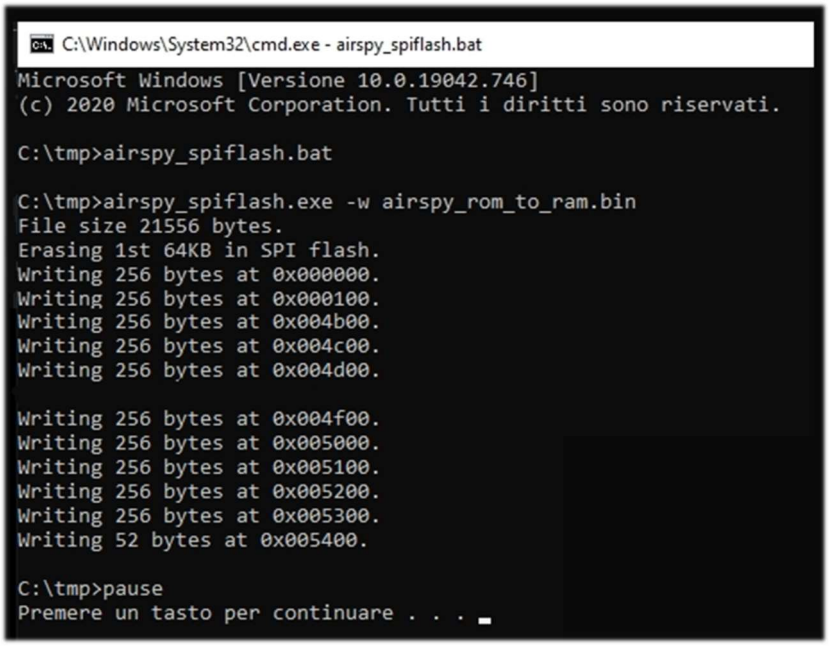
Start by extracting the content into a temporary directory (e.g. C:\TMP)

- In that folder, run the command line interpreter by typing CMD
- Type Airspy_info.exe and press Enter
- Immediately, the screen below will appear and will read your "Firmware version".



The firmware update procedure should be carried out under Windows 7/10/11. Make sure you do not have any other Airspy devices connected to your computer and follow these steps:

- Download and unpack in a temporary directory (e.g. C:\TMP) the contents of this file:
  https://Airspy.com/downloads/Airspy_fw_v1.0.0-rc10-6-g4008185.zip
- Connect the device to be updated to a USB port on your computer
- From the command line, run the file "Airspy_spiflash.bat", wait for the finish procedure (see screen)
- Disconnecting the Airspy device from the computer
- Reconnect the Airspy device to the computer and delete the temporary directory.

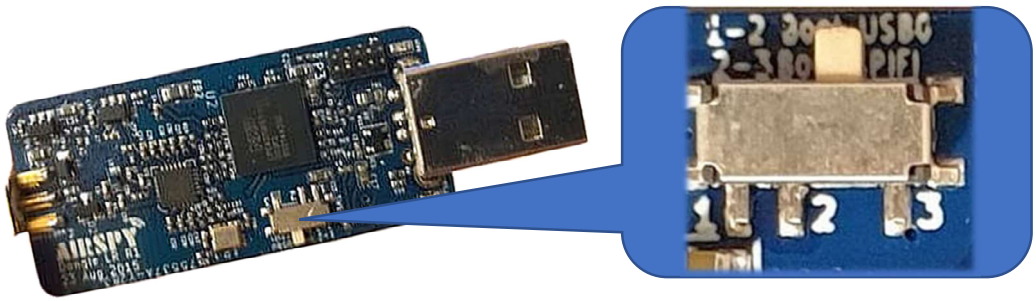


*The current and latest firmware release for the Airspy R2/Mini is **v1.0.0-rc10-6** (08-05-2020)*

While reading in a forum, I happened upon a thread that may be helpful to some readers.

The topic was about the difficulty to update the firmware of an **Airspy Mini** bought used...

Basically, it turned out that the upgrade was not successful because of the position of the internal microswitch on the side.



Position 2-3 is the correct one and allows Window to show the hardware in the device manager.

There are in fact two positions for the microswitch:

Position 2-3 Boot SPIFI (Flash) ← normal startup

Position 1-2 Boot USB0 (DFU) ← boot from ROM

The position 1-2 should only be needed if there was some difficulty during the firmware update or if it was not flashed correctly.

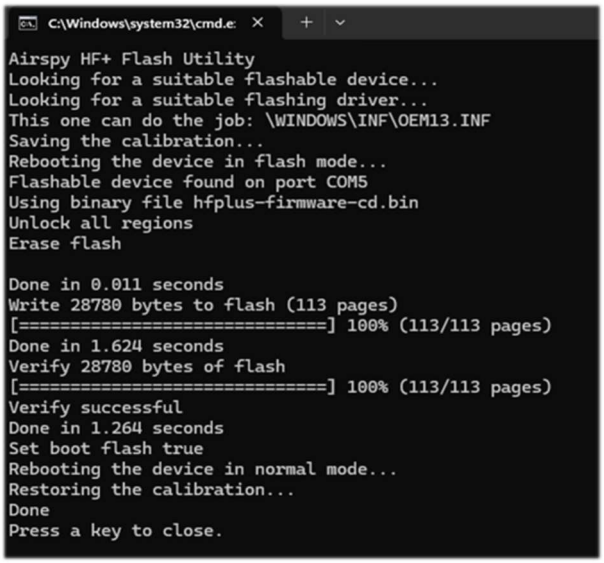
Please also read this disclosure carefully:

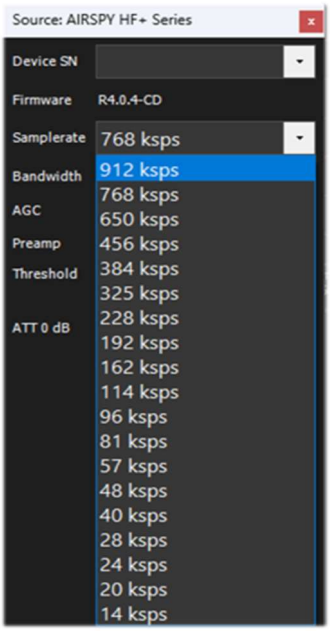
https://github.com/Airspy/Airspyone_firmware/wiki/Windows-how-to-flash-Airspy-firmware

Airspy HF+ Dual/Discovery firmware upgrade

The firmware update procedure should be done in Windows 7/10/11. Make sure you have no other Airspy devices connected to your computer and follow these steps:

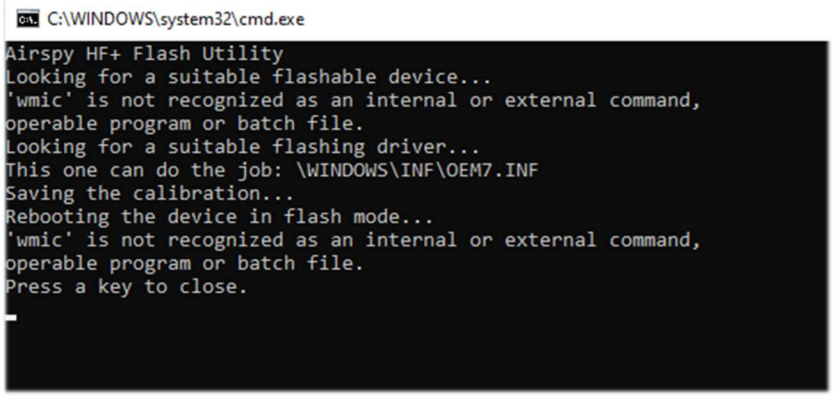
- Download and unpack the contents of this file into a temporary directory:
- unpack the contents of this file into a temporary directory:
- **https://airspy.com/downloads/airspy-hf-flash-2024-03-14_4.0.8.zip**
- Connect the device to be updated to the computer's USB
- From the command line, run the "FLASH.bat" file as administrator and wait for the procedure to finish
- Disconnect the device from the computer and reconnect the device (no longer needed since R4.0.3)





The current and latest firmware version is R 4.0.8 dated March 14, 2024, which has improved AGC, faster tuning and robust USB streaming, as well as the replacement/addition of some sample rates. It can be applied to HF+ Dual port devices, HF+ Discovery (BB and CD). Refer to the following table for the latest HF+ firmware CHANGE LOG Revision. The complete list can be downloaded here: **https://Airspy.com/downloads/hfplus_changelog.txt**

Revision	Date	Change log
R3.0.0	2019-07-19	Added processing gain compensation. Ready for Discovery.
R3.0.1	2019-07-30	Adjusted the Minimum AGC threshold to be 4 dB lower.
R3.0.2	2019-07-30	Set the AGC on by default.
R3.0.3	2019-08-16	Added support code for Pre-selector addon for the HF+ Dual Port.
R3.0.4	2019-08-19	Enabled the LNA control for AGC and Manual gain modes.
R3.0.5	2019-08-19	Adjusted the Low Gain theshold for the LNA.
R3.0.6	2019-08-20	Optimized the high AGC threshold.
R3.0.7	2020-06-04	Optimized the USB data streaming. Added 912 ksps and 456 ksps rates.
R4.0.0	2024-03-10	Updated the ST firmware. Better AGC. Faster tuning. More robust USB streaming. Replaced sample rate 256k => 228k.
R4.0.1	2024-03-10	Added fall-back support for legacy systems (out-of-date Linux, old libs, etc.)
R4.0.2	2024-03-11	Ported the firmware code from Atmel Studio to Visual Studio 2022 + VisualGDB. Added more compatibility code for manual gain in the HF band.
R4.0.3	2024-03-12	Added initialization code to make the unit usable right after flashing, without a hard reset.
R4.0.4	2024-03-13	Added 650 ksps sample rate with Low-IF. Stop the streaming when the USB connection is closed.
R4.0.5	2024-03-14	Adjusted the buffering to support slow USB controllers and USB hubs.
R4.0.6	2024-03-14	Increased the interrupt frequency for better handling of the AGC.
R4.0.7	2024-03-14	Fixed the initialization of the 384k and 192k sample rates.
R4.0.8	2024-03-14	Faster HF AGC convergence.



Very rarely, during attemting to flash the device, was found a messages like this one...

Try the operation with a different USB port o another computer.

Recovery procedure for firmware upgrade from initial R1.0.00

Due to a bug in the very first firmware, there is a specific procedure that should ONLY be used for this purpose when updating the R1.0.00 firmware. Subsequent updates should work with the standard procedure listed above.

- Open the HF+ case
- Connect the device to the PC
- Connect the "Erase points" for one second (see picture)
- Disconnect the device from the PC
- Connect again the device to the PC
- Double click on the FLASH.bat file
- Wait for it to be updated and verified
- Disconnect the device from the PC
- Connect the device to the PC again (the procedure is finished)

In the forum, I happened to read these notes that may be useful if needed...

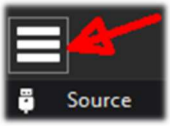
I wanted to upgrade the firmware of an HF+ Discovery from R3.0.6-CD to R3.0.7-CD. I had last done it two years ago on Windows 10 and it had worked without problems.
Now on Windows 11 and a new PC the upgrade stops with the warning "No free instances" in the CMD window and unfortunately the HF+ is no longer detected by SDR#...
In the Windows Device-Manager it is found in "COM & LPT" as COM9 (unknown device). Reinstalling the drivers with Zadig, the HF+ is recognized as a serial USB device and not as AIRSPY HF+ and also Windows does not allow to manually install winusbcompat.inf, but indicates that "the best driver for this device is already installed."

I then retrieved the old W10 laptop and connected the Airspy HF+ Discovery (which of course was not recognized). I opened the HF+ casing and shorted the "ERASE" points for a second. At this point it was possible to update the firmware without any problem.

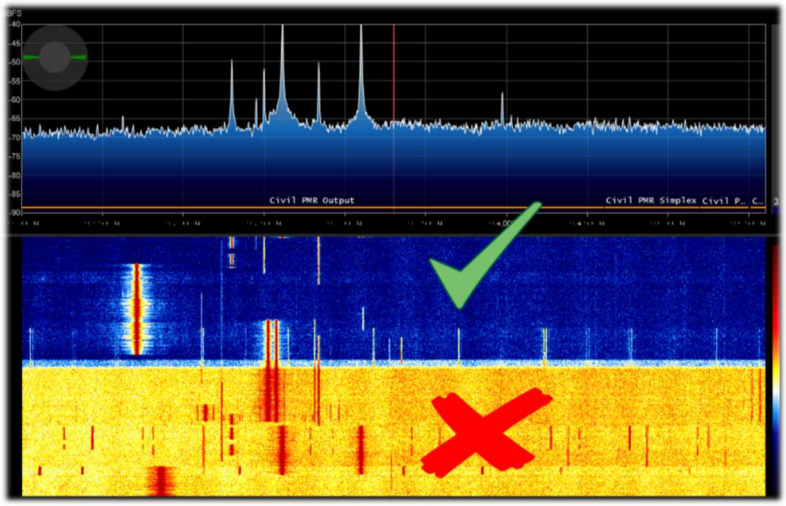
Also try disabling the Windows UAC-user account control...

Startup of SDR#

At startup, check the following points:



- Select from "Source" your receiver from those available starting from the "hamburger menu"
- Adjust the RF gain level as best as possible ***(sliders from zero to the right for higher values), taking care that the waterfall is not "saturated" too much with strong input signals represented with the color orange/red. Adjust the gain to bring them accordingly toward a dark blue color, only then can even weaker signals be detecte.***



- Reduce the "Range" slider (step 16) to about 30% from the bottom.
- Enable the "Correct IQ" field to remove the centre peak ***(only when using R820-T/R820-T2 dongles)*** or enable "Offset Tuning" in the configuration menu if using a dongle with an E4000/FC0012/13 chip.
- Disable the "Snap to grid" field ***(only up to v1918)*** in order to tune any signal independently of the specific step of the planned services or set it according to the preferred step (e.g. in FMN the step is 12.5 kHz). *If necessary, also disable the "Auto update radio settings" item in the "Band Plan" panel (read the specific function later).* ***For the demodulation of digital signals it is very important to tune the correct frequency: therefore if the transmission is at 430.512,5 kHz in DMR it is NOT good to tune for example at 430.515,788 kHz!!!***
- Set the correct "emission mode" according to the signals you intend to listen. ***Example WFM is not correct to demodulate FMN or digital signals!***

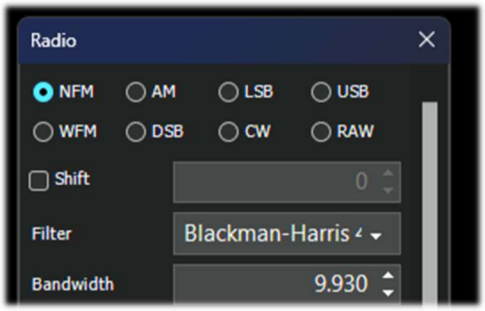
The following adjustment procedure ensures that you get the maximum SNR on what you receive while preserving dynamic range:

- Start with the RF gain set to the minimum level.
- Gradually increase the gain until the noise floor increases by about 5 dB.
- Check that increasing the gain does not also increase the SNR. Then increase the gain one notch higher and so on.
- Use the vertical blue bar of the "SNR meter" (to the right of the waterfall) to display the value.

Let us now take a moment to familiarize ourselves with the dynamic side menus (A and B). The different panels and plugins (including third-party ones, see the appropriate section below) can vary in number and relative position.
The menus (A and B) are all dynamic, you just have to hover over them to open them... For the various panels, in the upper right part, two options appear related to the positioning of the window, i.e. starting from the left the **Auto Hide** and the "X" of the **Close Window**.

Auto Hide – Pressing this control minimizes the panel window in the dynamic side menus (A and B). It opens/closes by hovering over it with the mouse.

The "already known" part, accustomed with previous Teleriks, is that of positioning individual panels with the recently introduced GUI wizard. Keeping the left mouse button clicked on the dark blue border of the panel/title, example the one related to "Radio"...



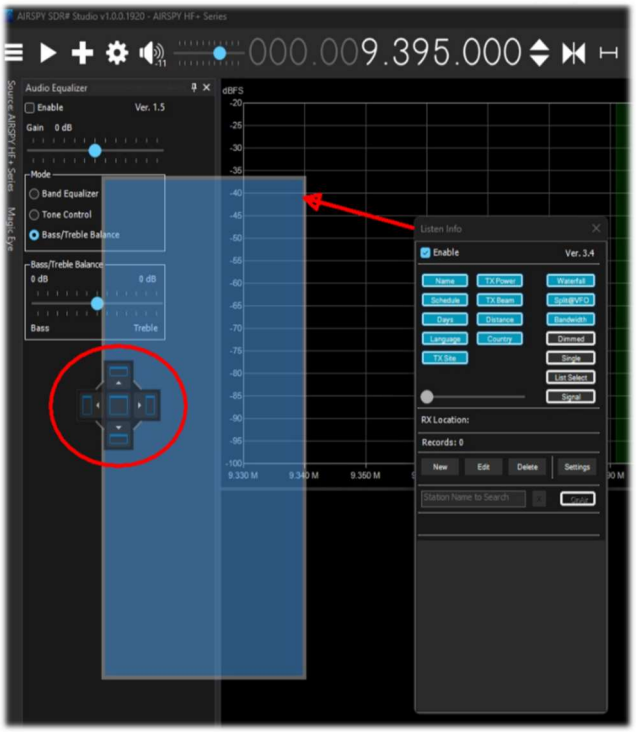
We start to move the mouse slightly (always keeping the left button clicked), four blue pointers with a side arrow will appear on the screen, we will have to position ourselves over the one in the area of our interest, move to the desired position and release the mouse button.

This is the grid of possible target areas with the four positioning icons (top, bottom, right, left).

icon for positioning toward left side

icon for downward positioning

Let us evaluate this example together:



We want to bring the floating window of the "Listen Info" plugin to the left...

We start by clicking with the mouse on the title window of the plugin and holding it down we move to the left where the pointers of the target area (circled in red color) appear.
We choose the desired point and release the mouse.

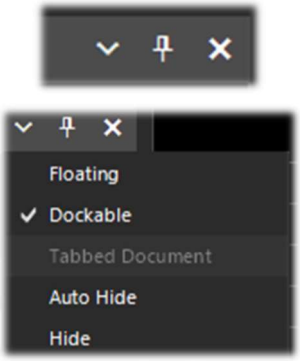
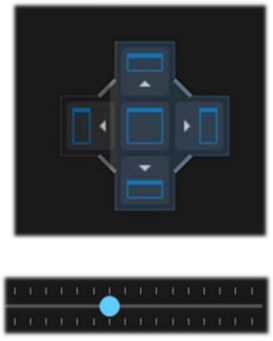
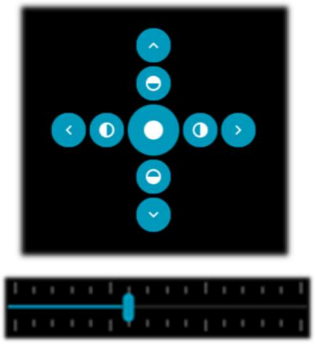
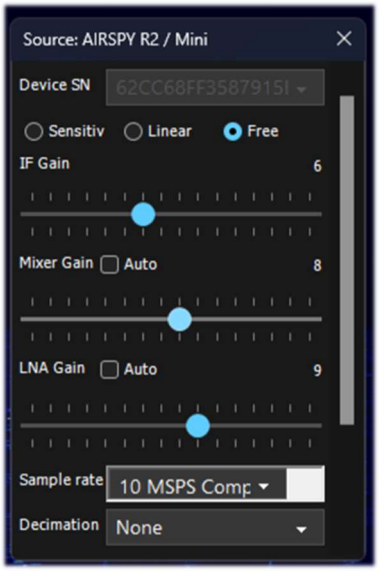
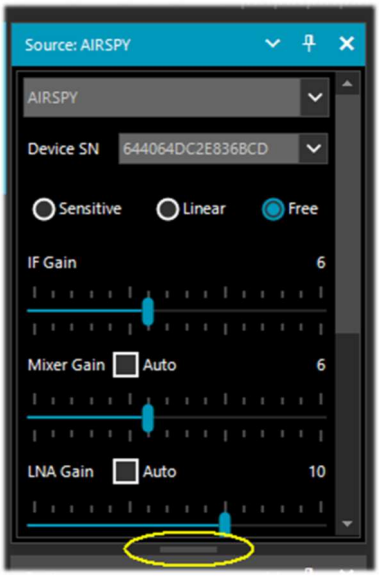
At this point, if the final composition is to your liking, you can decide to save the layout with a name of your choice so that you can recall it later through the items in the hamburger menu:

"Save Layout… e Load Layout…"

Personally, I have created several specific layouts: one for example for purely HF listening with active NINR and Co-Channel Canceller, another very minimal one to have in the foreground all the Spectrum/Waterfall and my ever-present Listen Info plugin, others then for V-UHF monitoring of amateur radio bands and FM 88/108 with the specific plugins.

As of v1919 the "Save/Load Layout..." function has been completely rewritten making it much faster and more accurate than previous versions. By activating it, SDR# shuts down for a few seconds and automatically reboots with the new GUI: worth a try!!!

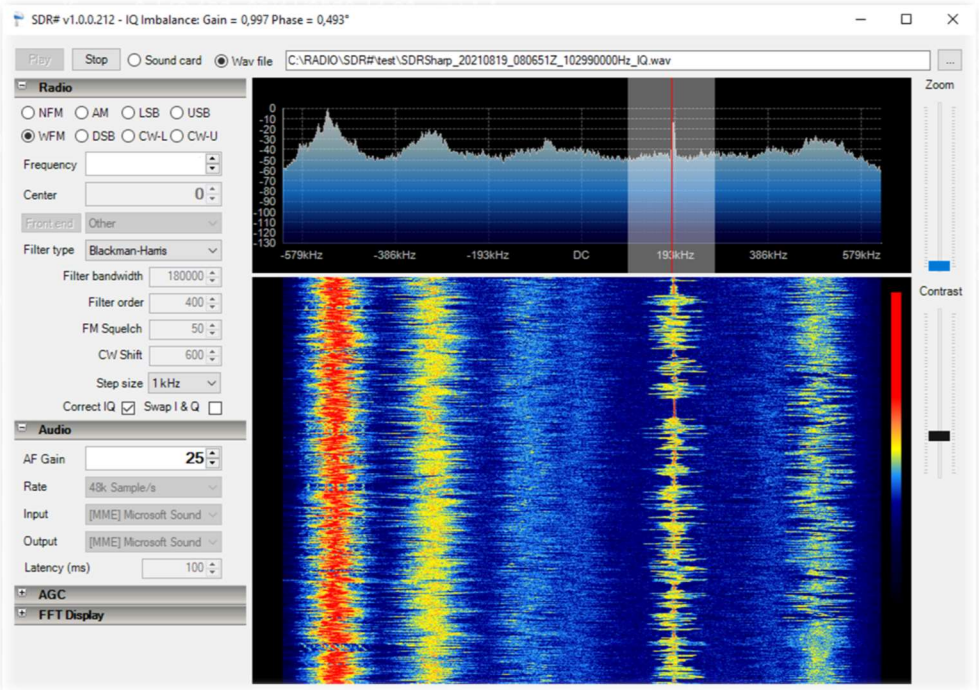
Some differences between the new GUI (v1920) and the previous Telerik (v1919)

	v1920	Telerik (v1919)
Options for window placement...		
Pointers used for targeting windows... And sliders…		
From v178x the various panels have an automatic scrolling function within the window. Until v1919, for resizing, you can dock with the mouse the horizontal bar highlighted by the circle in yellow color.		

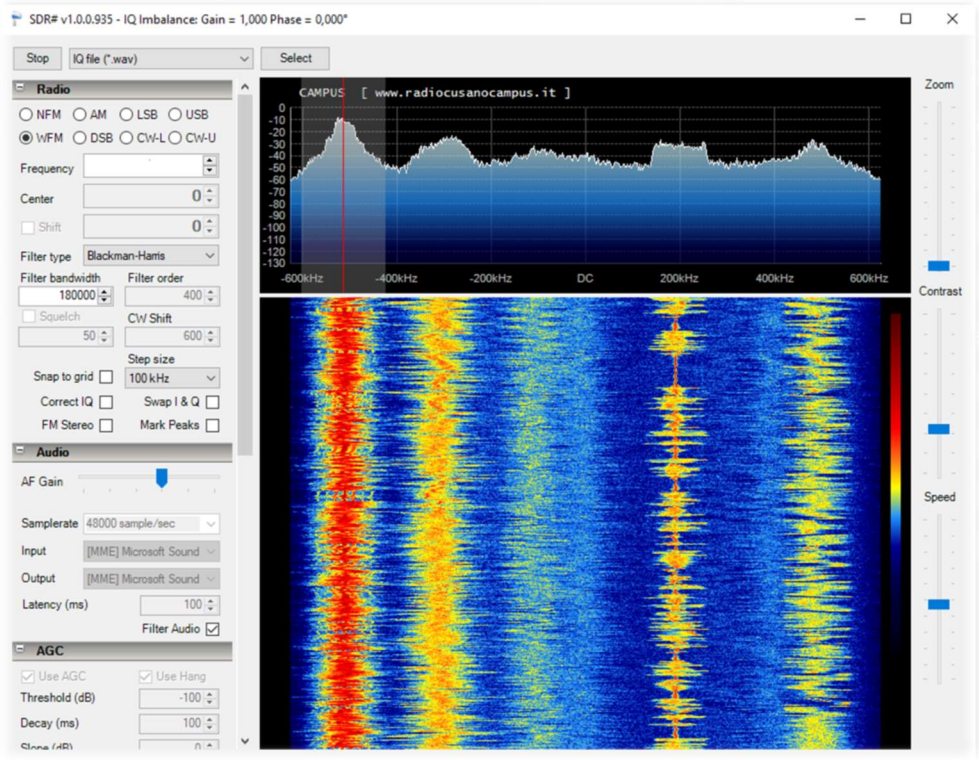
Amarcord (back to the past...)

"Amarcord" is a word of the italian dialect of Romagna, that simply means "I remember" and in this chapter I like to take a leap into the "distant past" and bring back some pleasant memories... I just found in the backup of an old hard disk some releases of SDR# that was taking its first steps in 2012, I offer them to you for the joy of amarcord (between distant past and present).

It was running April 30, 2012, and with only 412 k of software this was the SDR# v1.0.0.212 at that time: few commands and controls, but that was the essence. Here I tested it with an IQ file in FMW band.

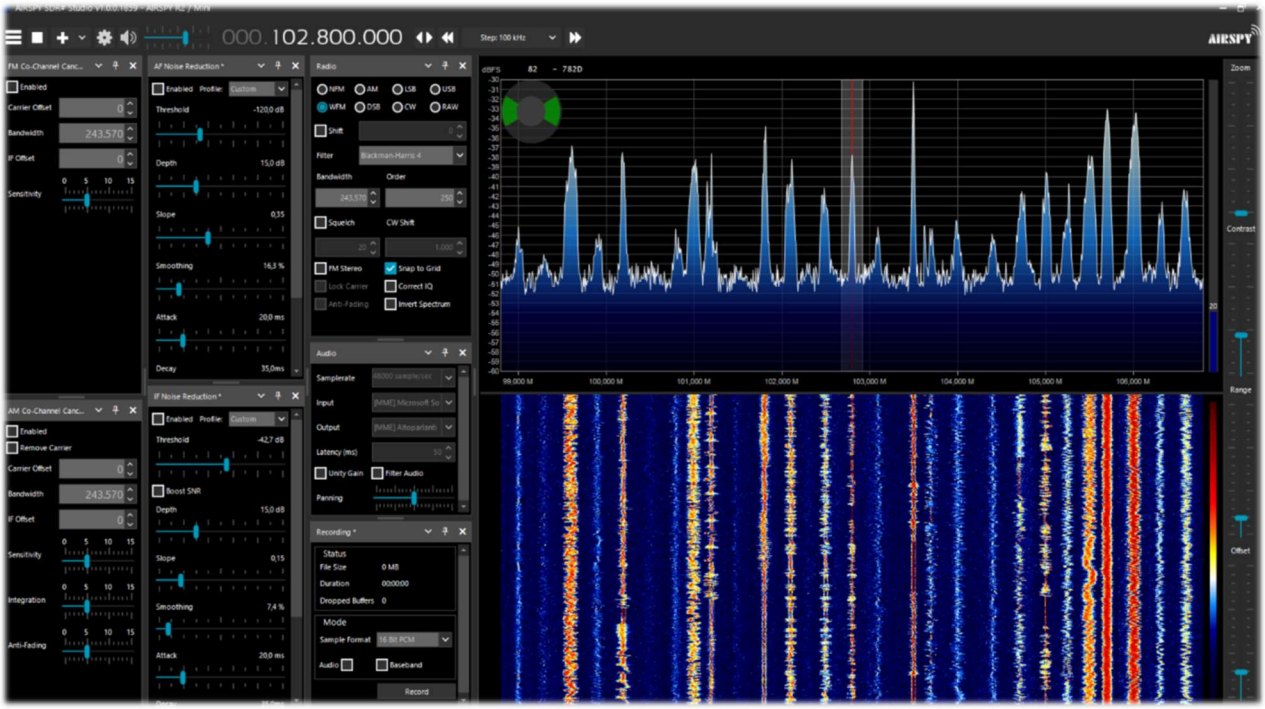


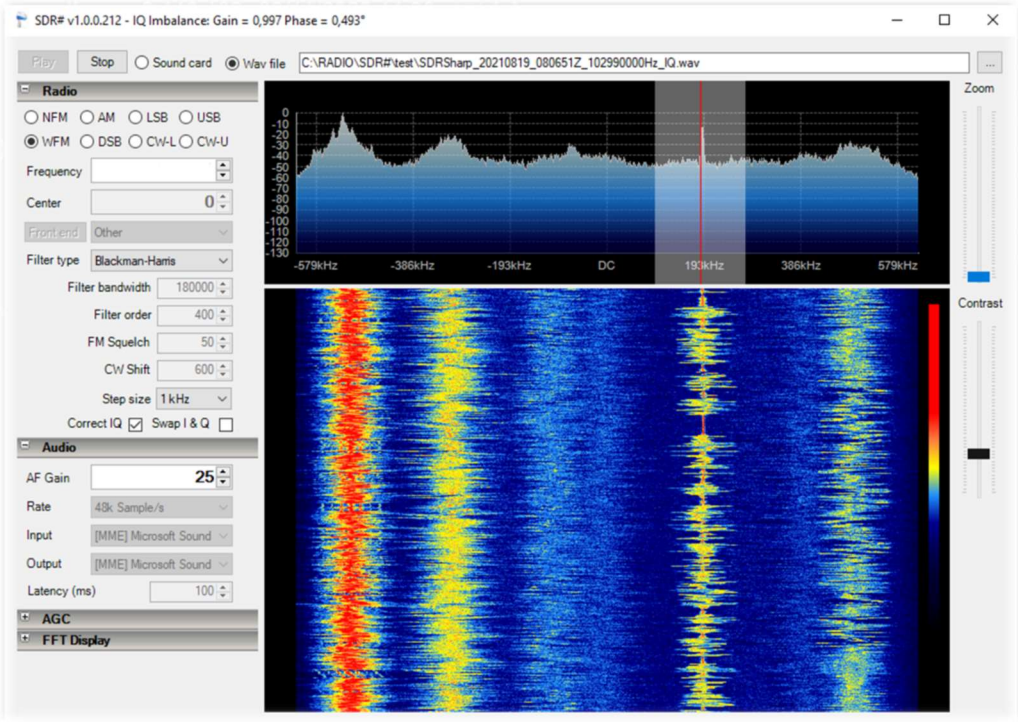
As early as October 2012, it had reached 621 k with SDR# v1.0.0.935: some more commands and functions such as RDS decoding, in the waterfall in the upper left...



Of course it is really impressive to compare ten years later (!) the screens of the two releases.

How much progress, development, ideas, and lines of code have been written to get this far...

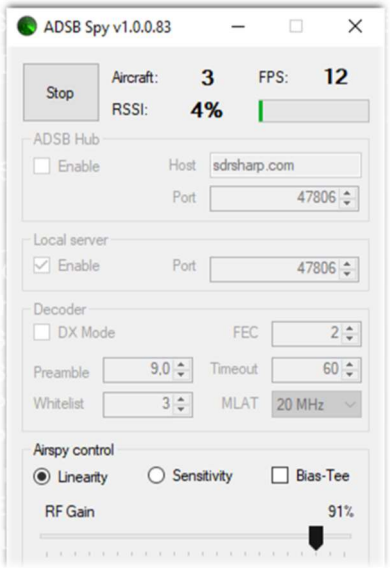




Who keeps even older releases?!

Until v1784 (which can still be downloaded from the previous links), the distribution included some standalone utilities **for use with Airspy devices only**, which many of you will remember and which can still be used today: ADSB Spy, Astro Spy and Spectrum Spy. They are briefly showned below...

ADSB Spy v1.0.0.83



Once started, after a few moments, the "Aircraft" and "FPS" counters will appear, indicating the data packets received, providing a real-time view of correct reception, as well as the received signal strength indicator (RSSI).

The default port address is 47806 and is used to communicate with the decoding programs (see below).

The "ADSB Hub" and "Local server" boxes are used to send data to a specific host, IP address and port.

Previous versions of ADSB Spy also allowed the use of normal RTL-SDR sticks with good results.

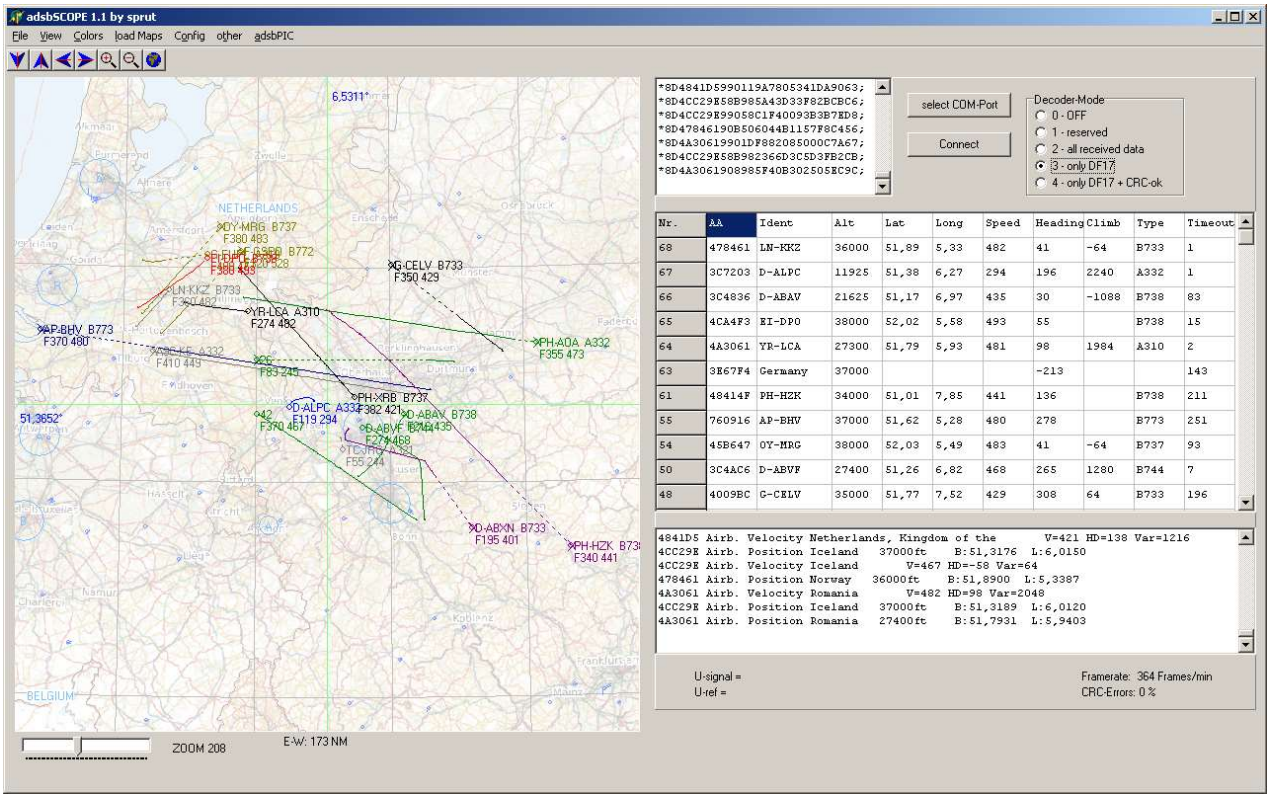
The most popular radar software, in alphabetical order, are:

adsbSCOPE (see screen): ***http://www.sprut.de/electronic/pic/projekte/adsb/adsb_en.html***

Planeplotter: ***https://www.coaa.co.uk/planeplotter.htm***

Virtual Radar Server: ***http://www.virtualradarserver.co.uk/Default.aspx***

Each one needs its own specific configuration and settings and this is not the place for a detailed individual sheet. Please refer to the links and various enthusiast sites on the net.



Astro Spy

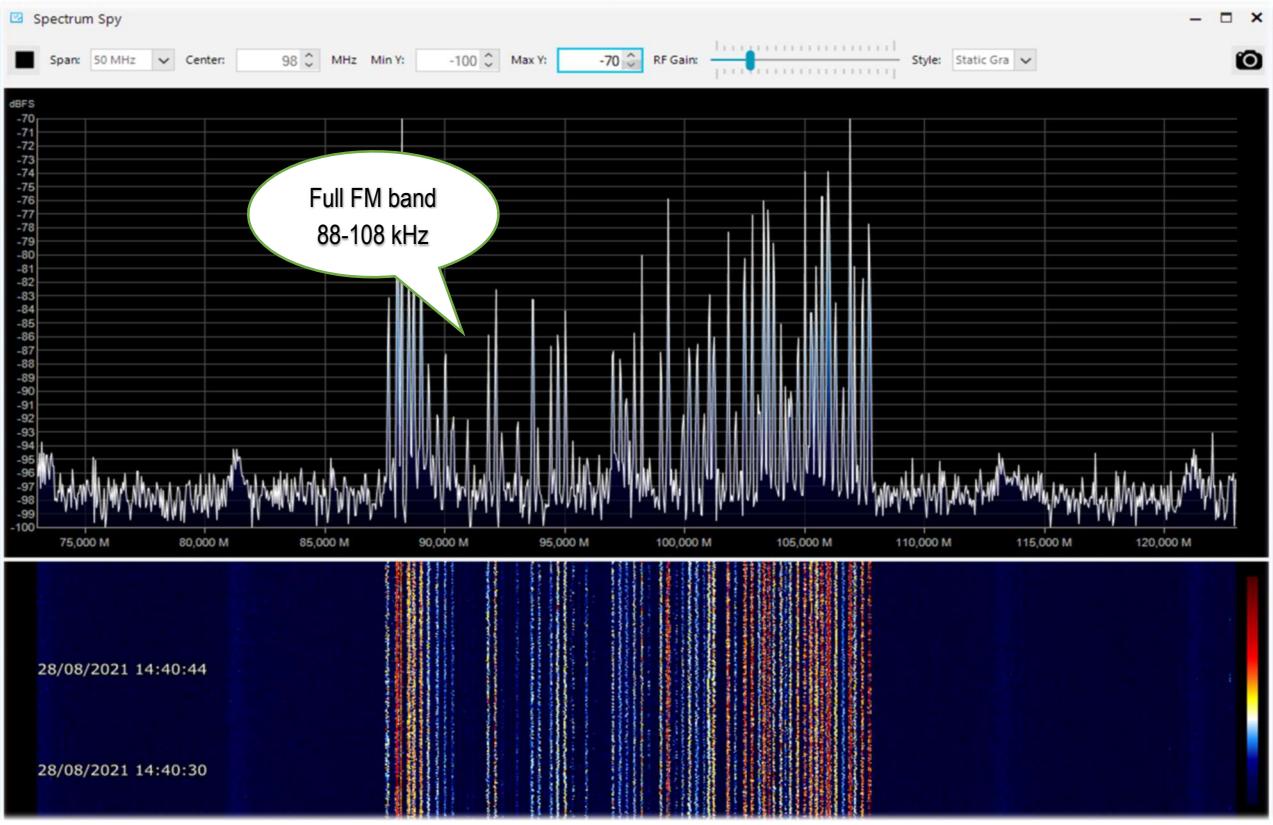
Developed for radio astronomy to observe a specific L-band frequency over time. I have not been able to test it, it should detect the hydrogen line 21 cm at 1420 MHz perhaps with a horn antenna pointed at the Milky Way.

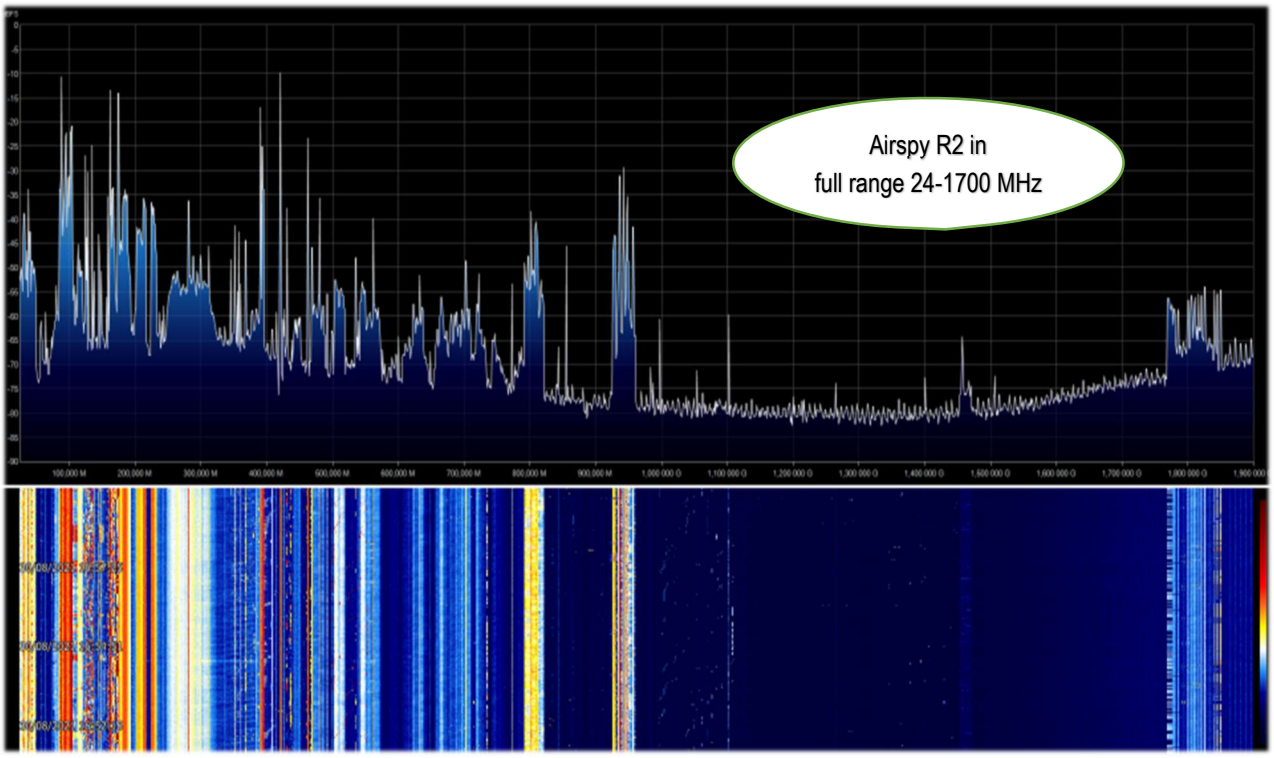
Spectrum Spy

The Spectrum Analyser allows the display (no sound) of wide frequency ranges (or the whole range in 'Full' mode) by exploiting the scanning speed which is comparable to 'real' spectrum analysers (...and maybe even more!).

I've loved it since the first time I used it and resort to it whenever I need to analyse small or large portions of the spectrum or to see the source of some new signal (often unwanted like a local noise) or if a slot has opened up in the 88-108 MHz range to try FM-DX...

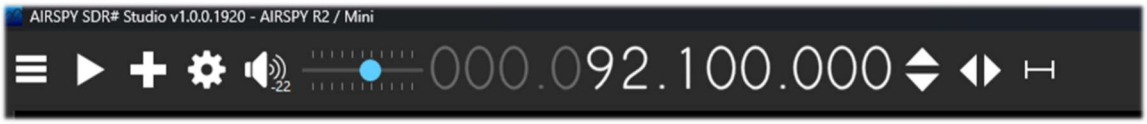
Key	Features
	Starting / closing the programme
Span	Allows you to choose a specific portion of the range for analysis (10, 20, 50, 100, 200, 500 MHz, 1 GHz, Full)
Center	Allows the desired frequency to be centred on the screen. *The combination of Span / Center allows the best analysis of the signal in the desired range.*
Min Y	To choose the minimum values for the ordinate axis (-80 / -120 dBFS)
Max Y	To choose the maximum values for the ordinate axis (-70 / 0 dBFS)
RF Gain	To increase or decrease gain
Style	Allows you to choose the style of signal representation in the Spectrum (Simple curve, Static gradient, Dynamic gradient, Old school)
	Allows a screenshot of the Spectrum/WF to be saved at any time.


Spectrum Spy
Span: 50 MHz
Center: 98 MHz
Min Y: -100
Max Y: -70
RF Gain:
Style: Static Gra
dBFS
Full FM band
88-108 kHz
75,000 M
80,000 M
85,000 M
90,000 M
95,000 M
100,000 M
105,000 M
110,000 M
115,000 M
120,000 M
28/08/2021 14:40:44
28/08/2021 14:40:30


Airspy R2 in
full range 24-1700 MHz

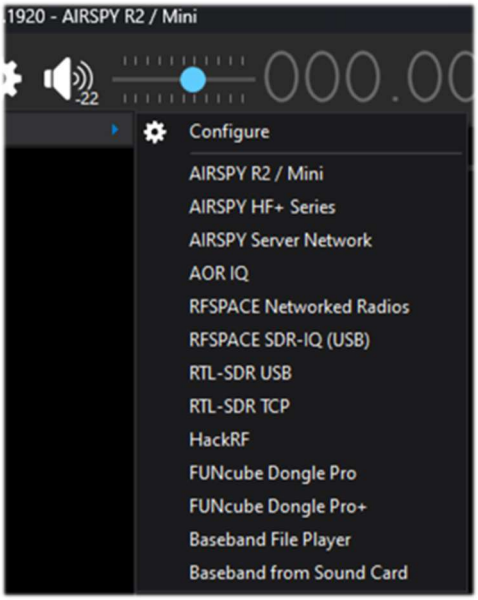
Default panels

The following are the default panels that allow all the basic functions provided by the software as well as some specific and unique features of SDRsharp. All others defined as "plugin" can be inserted and used by the user (see later the appropriate section) or even developed independently for their needs by those who have the knowledge and appropriate technical skill.



Source

Select from the "Source" panel your device from those available, and for some you can have additional settings by clicking on the "Configure Source" icon. In detail all supported devices:

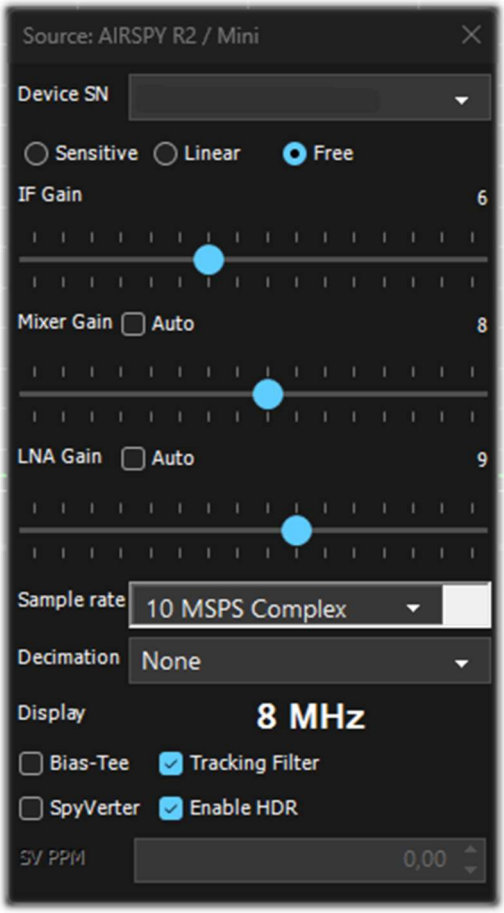


- AIRSPY R2 / Mini
- AIRSPY HF+ Series *(Dual/Discovery)*
- AIRSPY Server Network *(see apposite chapter)*
- AOR IQ *(Since v1920, support for high-performance radios from Japanese partner AOR Ltd. has been added. Those who own AR5700D, AR2300IQ or AR5001D and AR6000 with the IQ module can use it with SDR#).*

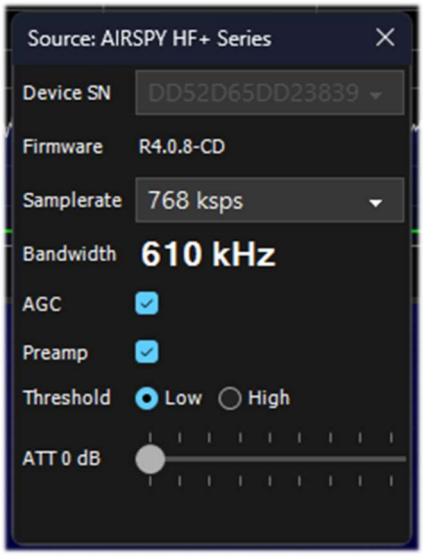
- RFSPACE Networked Radios *(Net-SDR, SDR-IP, CloudSDR)* / RFSPACE SDR-IQ
- RTL-SDR USB or TCP *(new I/O interface)*
- HackRF
- Funcube dongle Pro/Pro +
- Baseband File Player to load & play I/Q files *(see below the chapter "Baseband Recoder").*
- Baseband from Sound Card

For AIRSPY there are then to adjust: Gain controls (IF, Mixer, LNA in a simplified or specific way Sensitive/Linear or Free), Sample rate, Decimation, Bias-Tee *(this option should be used carefully as it sends 5 volts via SMA antenna connector to optional accessories such as LNA or UpDown converter),* SpyVerter which allows the hardware option to receive the HF (0 – 60 MHz), Tracking Filter and HDR. The HDR feature gives a boost in dB in dynamic range.
This means that the gains can be turned up further without overloading occurring, and that weaker signals can come in much stronger without strong signals overloading and drowning them out.

Airspy R2 / Mini

Panel	Feature
	**Device SN** – Serial number of your device. **Sensitive/Linear/Free** – Three different choices for the gain adjustment at IF Gain, Mixer Gain and LNA Gain level. *"Free" is the one that allows greater user intervention and customisation: there are no predefined settings and everyone will have to adjust it as best they can according to their own operating environment.* **Sample rate –** Allows you to choose the sampling: • Airspy R2: 10 or 2.5 MSPS • Airspy Mini: 6 or 3 MSPS ***On dated computers with poor CPU/RAM it is better to use 2.5 and 3 MSPS or else croaking audio...*** **Decimation –** Allows a lower bandwidth to be used to the benefit of bit resolution and therefore lower quantisation noise. Values: none, 2, 4, 8, 16, 32 and 64. *To make the best use of it, recommend adjusting the Gain levels (shown above): the more you work in decimation, the more you can increase the gain!* **Display –** The value shown of the bandwidth displayed in the Waterfall and Spectrum windows is linked to the previous "Sample rate" and "Decimation" settings and changes for the different devices: • Airspy R2 10 MSPS (from 125 kHz to 8 MHz) • Airspy R2 2.5 MSPS (from 31.25 kHz to 2 MHz) • Airspy Mini 6 MSPS (from 75 kHz to 4.8 MHz) • Airspy Mini 3 MSPS (from 37.5 kHz to 2.4 MHz) **Bias-Tee** – Allows the use of optional devices requiring an additional power supply: 4.5v *(at 50 mA?)*. **Tracking filter** – Taking advantage of decimation and enabling this filter will result in better selectivity, *so more gain can be used!* **SpyVerter** – Enables the optional "SpyVerter" device (see relevant chapter), which allows reception from longwave to 35 MHz and the initial portion of VHF. *In HF, the "Linear" mode, is recommended for the gain.* **Enable HDR** – When activated (with software off) applies a combination of analogue and digital filters to optimise the dynamic range for the visible spectrum. *A high decimation ratio can be activated and selected for better reception.* **SV PPM** – Airspy devices are factory calibrated to approximately 0.05 ppm. This value can be adjusted for the SpyVerter. *Updating the firmware will not change this value which is stored in a different location.*

Airspy HF+ Series

Panel	Feature
	**Device SN** – Serial number of your device. **Firmware** – Indicates the firmware version installed on the device *(see firmware upgade procedure below)*. **Samplerate –** Allows you to choose the sampling rate: from a minimum of 14 ksps to a maximum of 912 ksps. **Bandwidth** – The bandwidth displayed in the Waterfall and Spectrum windows is linked to the previous Samplerate value: from a minimum of 10 kHz to a maximum of 725 kHz. **AGC** – Automatic gain control. *It is recommended that you leave it ON (at the same time the Threshold to "Low" value) or disable to OFF and manually adjust the HF ATT cursor value.* **Preamp** – Activates or deactivates the preamplifier. *Set to ON for low signal reception, OFF for strong signals.* **Threshold** – The "Low" threshold value introduces attenuation but gives the signal better linearity, unlike the "High" position which favours sensitivity. *You have to wait a few seconds to appreciate the differences when changing the threshold.* **HF ATT** - With the value set to OFF in the previous AGC setting it is possible to change the attenuation value slider from 0 dB to a maximum of 48 dB with 6 dB steps.

Dongle RTL-SDR's configuration

The general rule of thumb for all of the following setups is that the default settings work and any changes require a good understanding of the underlying algorithms and own hardware.

It starts in plugging one of our dongles into a USB port (various types are available on the market with R820T/T2 or R860 chips and the older E4000, FC0012/13).

For sensitivity and stability, obviously is ***recommend the original RTL-SDR.COM (V4 or V3).***

rtlsdr.dll - 266.752 bytes Sept 18, 2023

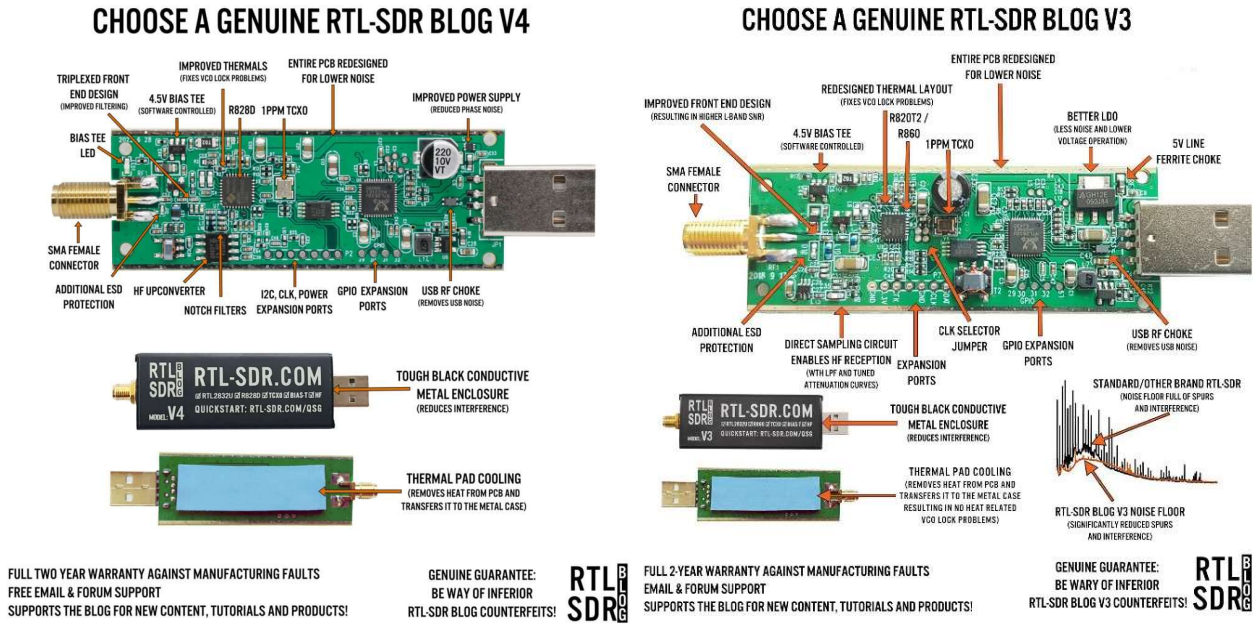
rtlsdr.dll - 44.032 bytes Sept 17, 2019

A detailed English-language user guide for V4 is available: **https://www.rtl-sdr.com/V4/**

Thanks to Carl Laufer of RTL-SDR.com and of course beware of clones by checking well before purchase that this logo appears on the device!

RTL-SDR have these main features:

- Temperature compensated oscillator (TCXO) at <1 PPM. Accurate tuning and almost zero temperature drift (2 PPM max. initial offset, 0.5-1 PPM temperature drift).
- SMA antenna connector. Generic dongles use MCX or PAL antenna connectors, which are less common and wear-resistant. The SMA port is more common and adapters are available in various formats. Also is such connector has lower RF insertion losses.
- 4.5V Bias tee via USB. Allows the RTL-SDR to feed low noise amplifiers (such as the LNA4ALL, the HABAMP, the LNA from the RTL-SDR ADS-B blog) and active antennas directly through the coaxial cable. Can be enabled via software.
- HF Direct Sampling (V3) or HF upconversion (V4) modes. Receive 500 kHz to 28.8 MHz with direct sampling on the V3 model or upconversion of the V4 model. There is a built-in 25 MHz low-pass filter, but additional HF filters (e.g., to attenuate strong MW AM frequencies) may be required for optimum performance.
- Aluminum case and passive cooling. These devices have an aluminum case and passive cooling via a silicone thermal pad. This prevents deterioration in reception due to heat when used above ~1.2 GHz.
- Improved front-end filtering for V4s. This model is less prone to saturation by strong transmitting stations.
- Various improvements over other RTL-SDRs: R820T2 tuner, higher-quality passive components, an inductance on the USB line to reduce USB noise, improved circuit board design to significantly reduce spurious and internal noise, improved ESD protection, additional bypass capacitors and ferrite inductances on the power line, improved front-end coupling circuit, a modified power supply design for long-term reliability, and improved LDO voltage regulators.

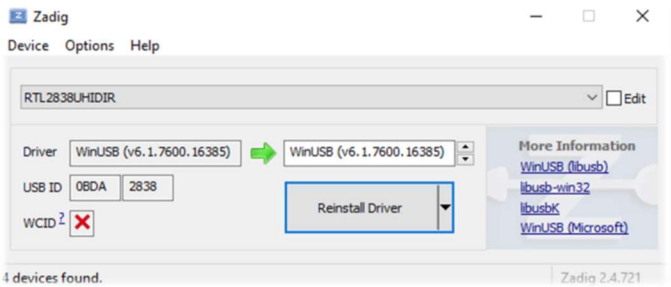


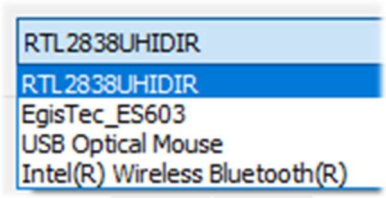
SDRsharp is preconfigured for AIRSPY but is fully compatible with any RTL-SDR dongle via installation of the drivers, not in the original package, by running the batch file "INSTALL-RTLSDR.BAT" found within the zipped package.

With the release in August 2023 of the new RTL-SDR v4 dongles (https://www.rtl-sdr.com/rtl-sdr-blog-v4-dongle-initial-release), an update of the 32-bit driver "rtlsdr.dll" for support in SDR# was necessary, to be carried out again via the updated "INSTALL-RTLSDR.BAT" script. If it is not seen try changing USB port !!!

You then run the ZADIG.EXE software.

In the OPTIONS menu select "LIST ALL DEVICE" (if possible without any other USB devices connected to the computer), in the drop-down window should appear the identifier of your dongle to be selected, example "RTL2838UHIDR".



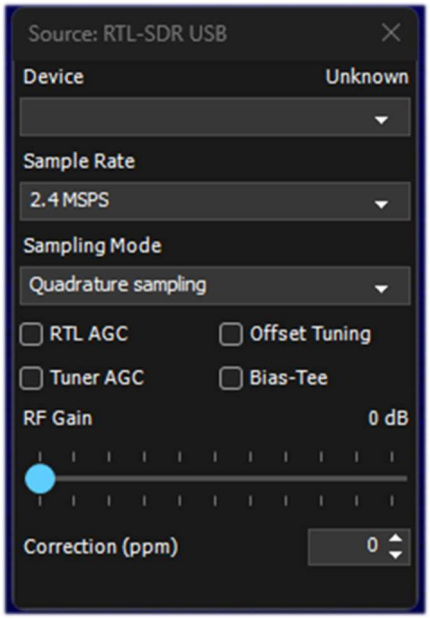
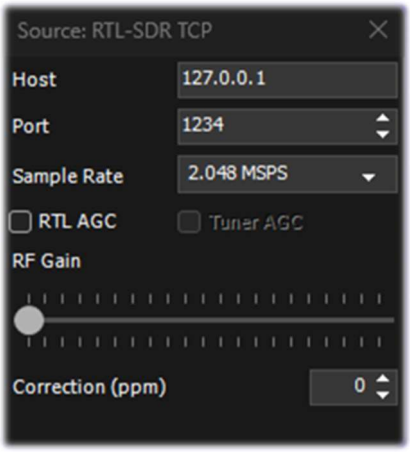


You proceed by clicking on INSTALL DRIVER (or REINSTALL DRIVER if the operation has already been performed). You can also try connecting your USB dongle after ZADIG is running, as the list is automatically updated by the system

Always be very careful in this operation and select only the ID of your device and not e.g. mouse or bluetooth keyboard or anything else otherwise you will create serious problems with these peripherals by ending up overwriting their original drivers!!
You have to check the USB ID well several times before clicking on "INSTALL DRIVER"!
Sometimes some dongles on first installation are recognized by the system as "Bulk-in, interface 0" and "Bulk-in, interface 1" (the latter is for TV remote control functions).

You then choose "Interface 0" with the target "WinUSB" and click on INSTALL DRIVER.
If you still do not see your device you must go to Control Panel/Windows Device Manager and remove those devices marked with a triangle and start again.

After a few moments, you can start SDRsharp and select "RTL-SDR USB" in SOURCE.

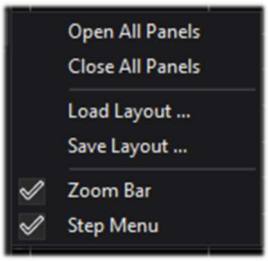
Pannel	Feature
 TCP/IP configuration 	Click on the configuration button (4) (the cogwheel). **Sample Rate** – Allows you to choose the bandwidth to be displayed (0.25 to 3.2 MSPS). *Generally, settings up to 2.4 MSPS work well on most PCs, but for slower machines we recommend reducing this value.* **Sampling Mode** – To tune above 30 MHz, leave set "Quadrature sampling". "Direct sampling" (I/Q branch) mode should be selected for lower frequencies for those dongles that are already set up for HF operation (otherwise a hardware change is required). **Offset Tuning** – For use on E4000/FC0012/13 tuner chips only. Selecting this option will eliminate the centre peak in the spectrum. **RTL AGC** – Enables AGC for RTL2832U chips only. **Tuner AGC** - Enables AGC *In many cases it is better not to flag it and manually setting the slider below.* **Bias-Tee** - Introduced with v1920, allows the use of optional units that require the additional power supply. **RF Gain** – Use this slider to manually set the RF gain value. *Start from an average value in dB and gradually increase towards the maximum on the right according to the signals received.* **Frequency correction (ppm) –** Allows to set a correction value for those cheap dongles that do not have a TCXO. *Not needed for Airspy users device!* If the dongle is not centred in frequency, tune in a strong and stable signal *(after ten minutes of powering up the dongle having reached the correct temperature and stability),* changing the ppm value a little at a time so that it is centred at the tuning bar (point 13). *It should be pointed out that RTL-SDRs go high in sample rate even up to 2.56 MSPS which is the maximum stable sample rate, but at 2.56 MSPS you cannot use the ppm offset adjustment because you will lose packets. This is because the ppm adjustment slightly changes the sample rate and anything above 2.56 MSPS* ***becomes unstable and critical especially in decoding of the digital signals!*** *Therefore, if you feel the need to make a ppm adjustment, it is best to stick to sample rates of 2.4 MSPS or 2.048 MSPS.*

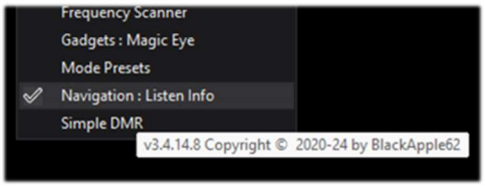
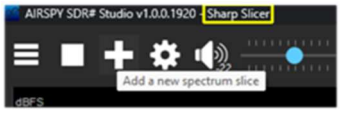
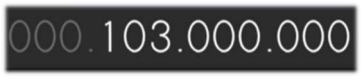
Please note:
If you notice that something is not working properly, it is best to uninstall everything and start over, especially with the cheaper and unfamiliar dongles… very often it is a problem of driver conflict or driver obsolescence. It is worth remembering that to always have optimal performance it is more convenient to use an Airspy since it is always difficult and costly to fix obsolete drivers and plugins for often critical third-party hardware.

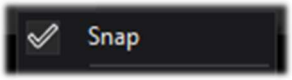
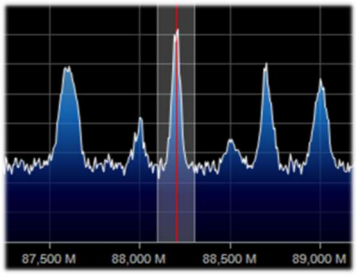
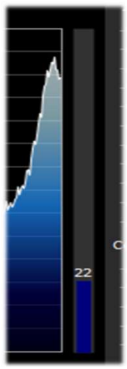
Main settings

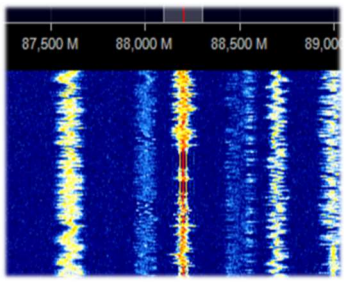
As of v1825 SDR# shows some information about the user interface and plugins loading on the splash screen when starting...

The main settings and controls apply to all devices. The only differences, in some menus, may concern the side to which SDR# interfaces. All devices must be configured in the menu where you can find your RF gain controls, sample rate, AGC, PPM, etc...

Key	Action
Menu panels 	With the button  *(referred to as a "hamburger menu" in the jargon)* you access to individual panels/plugins. The check mark, which precedes some items, is a "visual reminder" of the activation of certain options within the plugin (e.g., "audio or service xxx enabled"). From v1892/1894, as a result of "popular demand," there is a radical change in the main menu now divided into alphabetically ordered blocks/topics, complete with icons, to better manage the default panels and multiple third-party plugins that before, all together, risked clogging up the user interface. Starting from the top we have the indispensable panels: **"Radio", "Source", "AGC", "Audio" and "Display".**  In "Layout" we find: **"Open All Panels"** and **"Close All Panels"** opens and closes all panels altogether. **"Save Layout..."** and **"Load Layout..."** to save/load your layout, i.e. the layout and sizing of the active windows ***Since v1919 this function has been completely rewritten making it much faster and more accurate: activating it closes SDR# for a few seconds and automatically restarts it with the new layout (which will also be retained on subsequent restarts).*** *Normally, by closing SDR#, many of the settings are automatically saved in the internal configuration files (e.g. Source device, VFO frequency, mode, volume, audio output), but not everything, e.g. Zoom level is reset every time.* For **"Zoom Bar"** and **"Step Bar"** please refer to the dedicated chapter.

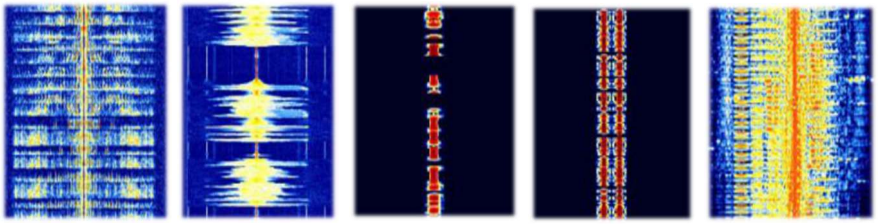
	In the last row, in **"Plugins",** we find precisely all those developed by third parties, *sorted alphabetically and complete with po-up about author and last installed release!* 
Start  **Stop** 	With this buttons you start / turn off the SDR# software.
New slice (VFO)  	This button can be used to open one or more new SDR# sessions *(not just present in the "Spy Server Network")*. The "slice" is a separate session showing a portion of the spectrum of the "master" with full separate controls, but still in the sampled portion of the band! *For example, it is not possible to open a session in UHF if the "master" receiver is tuned to VHF. Attention: with the introduction of this function from v1741 the previous Aux-VFO plugins (which used the same internal DSP algorithms) are no longer usable. To reduce CPU usage, disable the slice you do not need and minimise its bandwidth.* See the chapter "Listening Recipes" for a significant example...
Configuration menu 	Configuration menu of your hardware and settings: gain, sample rate, bandwidth, RF, PPM controls, etc.
Volume  	Turns the volume on/off *(from the speaker or for an external device example a VAC).* You control it with the slider on the right for the desired level. By clicking on it the icon becomes "muted," then moving from left to right goes from -80 to + 20 dB.  *It is useful to remember that if you use external software decoders for digital systems (e.g. DSD+), the volume control should be adjusted accordingly to have an optimal output signal level (and minimum errors).*
Input and VFO frequency 	The frequency input is represented in 4 sections (000.000.000.000). The first section from the left represents the values in GHz, the second the MHz, the third the kHz and the fourth the Hz. In the example to tune 103 MHz the input must appear as 000.103.000.000 while if you want to tune a frequency e.g. in MW at 999 kHz, in addition to needing an up-converter (or the optional unit Airspy Spyverter) you must enter 000.000.999.000 • *Move the mouse over the first digit that you want to impute (without clicking) and enter all the numbers that make up the frequency and confirm with the Enter key. This is the input I prefer! For example, you stand where the arrow points and*  *type in the number 103 followed by Enter. Quick and easy.* • *Left click on the top of the digits (a small red rectangle will appear) to advance one unit* • *Left mouse click on the bottom of the digits (a small blue rectangle will appear) to decrease by one unit*

	• *or on the desired digit by turning the mouse wheel on it.* • *Right click the mouse to bring a digit to zero and reset all the ones to the right of it as well* • *UP / DOWN arrow keys change the digit* • *The Right/Left arrow keys move along the section in the input* Since v1904, a new scalable font has been introduced, managed in the SDRSharp.config settings (see the appropriate chapter).
VFO frequency control 	With this dual control we can increase/decrease the frequency of the VFO with quick mouse clicks according to the Step you defined in the previous field. *Function introduced by v1899.*
Tuning types	 **"Free tuning"** - free tuning throughout the range, by clicking anywhere in the RF spectrum or waterfall, the receiver will tune it, also changing the below frequency range indication.  **"Sticky tuning"** - the frequency remains "connected" to the VFO and you can scroll the frequency bar left and right by "hooking" it with the left mouse button.  **"Center tuning"** - the tuned frequency will be always displayed in the center of the RF spectrum and waterfall.
Step control 	Clicking the button opens the control to choose the appropriate tuning step, or you can disable the "Snap" to have completely free tuning.  *See the "Step Bar" section below.*
RF Spectrum 	In this window the RF Spectrum is shown visually as a real-time graph. The active signals appear as peaks of greater or lesser intensity. The lower part represents the "noise floor. *A recently introduced feature is the "peak Color" which is activated by clicking with the right mouse button on the Spectrum window, where a yellow line of persistent memory relative to the received signals will appear. It is possible to modify the color by changing the following line in SDRSharp.exe.config:* ***"spectrumAnalyzer.peakColor" value="FFFF00"***
SNR meter 	On the right side of the RF spectrum there is a vertical bargraph that shows the SNR value (in dB). The Signal-to-noise ratio is a numerical quantity that relates the power of the useful signal to the the noise in the system. In the case of analog transmissions, a decrease in SNR causes a gradual deterioration of the received signal, which is still receivable and understandable. In digital transmissions instead there is a minimum threshold of SNR below which the system no longer works for too many errors received. *There is no S-meter to detect the signal strength, intended as an S-unit and mainly used in the radioamateur world.*
Waterfall	This window shows the graphic representation in real time of the intensity of the received signals as a function of frequency (on the horizontal axis) and time (vertical axis) with the new data represented



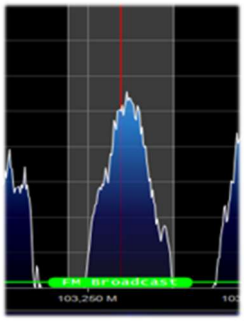
in cascade starting from the top and going down: hence the name waterfall.

This representation is a great help to learn about the various types of signals visually.
A trained eye detects and recognizes an interesting signal at first glance, even if it is weak and in the midst of disturbances, because each signal has its own "footprint", as well as electrical noises of all kinds !!
Here are some examples of signals:



For easier recognition of the very many types of signals and modulations, I refer you to the chapter "Signal Decoding and Analysis"...

Tuning bar

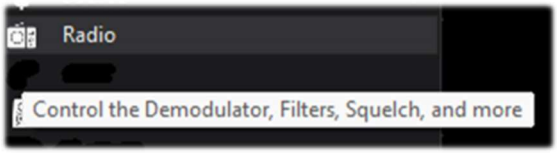


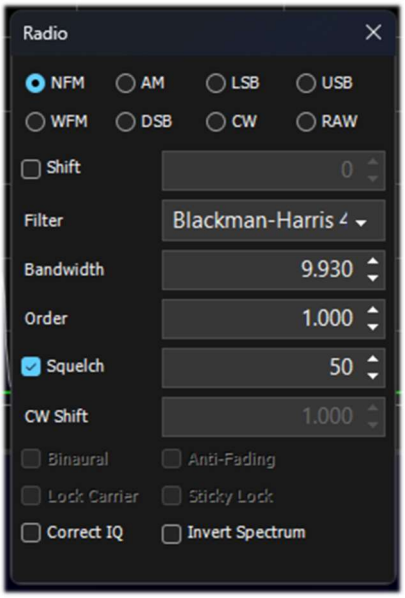
The vertical red line in the center of the RF spectrum windows shows which frequency is currently tuned the receiver.
The inside of the gray rectangle is the active bandwidth (or BW) that can be changed by simply dragging the left/right side of the rectangle.

The bandwidth must be set so that it covers the area of the tuned signal (not too wide or too narrow, especially when receiving digital signals).

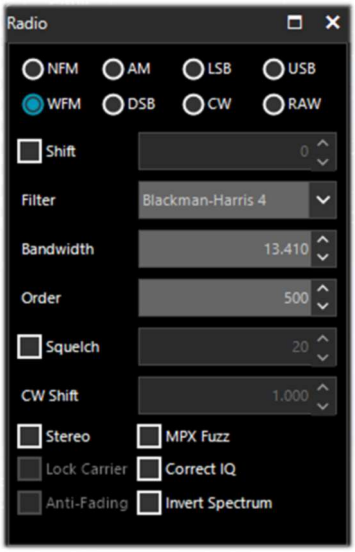
Radio

In this panel you select the various types of demodulator for the tuned frequency, Filters, Bandwidth, Squelch, Step Size, etc…

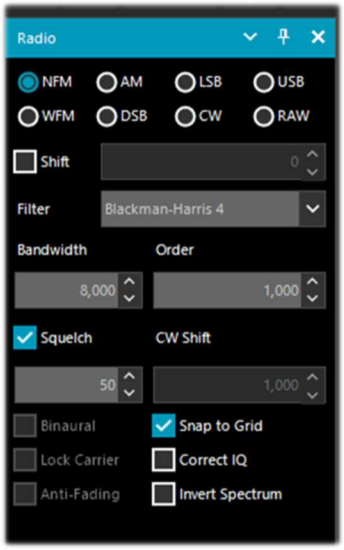




v1920

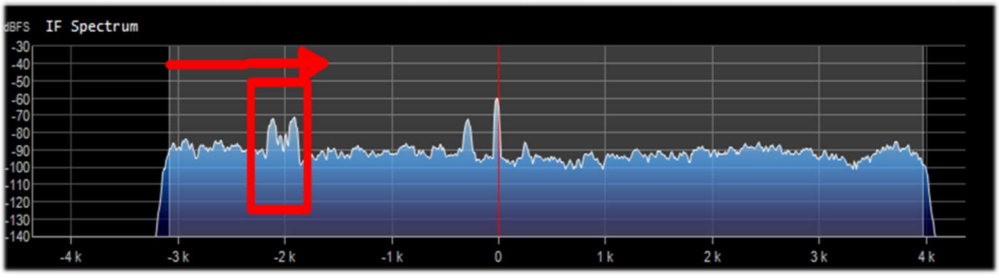


v1919

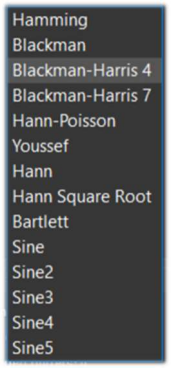


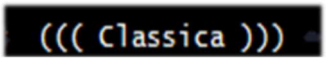
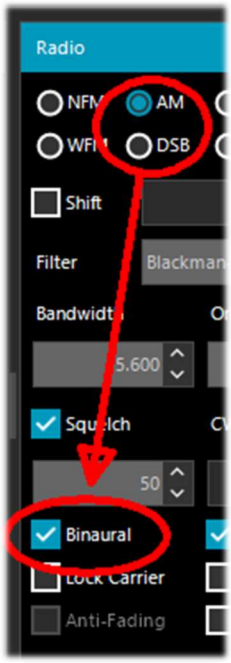
until the v1910

Mode	Features
NFM	Transmission technique that uses the frequency variation of the carrier wave. Mode commonly used by civil and radioamateur services for both analog and digital modes in VHF/UHF but not under the 27 MHz. *The v1861 adds a mixed frequency-domain and time-domain FM demodulator. It improves everything related to FM, including weak NFM signals and RDS (still improved in v1913/1920) for FM Broadcast.* ***Important note:** the new demodulator works best when the entire signal is covered appropriately by the bandwidth (BW) filter on the VFO.*
WFM	This is the mode used by FM stations (88-108 MHz band). For stations with RDS-Radio Data System, in the upper part of the Spectrum RF, on the left (see item 11) there is the dynamic decoding of some codes RDS that carries a lot of informations: ➢ **PI, Programme Identification**. Unique four-character alphanumeric code that identifies the radio station. ➢ **PS, Programme Service**. They are eight characters used, usually, to send the name of the radio also in a dynamic way. ➢ **RT, Radio Text**. It allows to send free text from radios such as, for example, the author and the title of the song on air. *The v1861 adds a mixed frequency-domain and time-domain FM demodulator. It improves everything related to FM, including weak NFM signals and RDS for FM Broadcast.*

	 *Since v1912 RDS detection and decoding has been further improved even on poor and difficult signals*. In the screen you can see the difference with a third-party external decoder (please see the chapter: "MPX Output e RDS-Spy"). ***Important note:** the new demodulator works best when the entire signal is covered appropriately by the bandwidth (BW) filter on the VFO.* The **deemphasis** process (in the receiving phase) is designed to achieve linearity of the frequency response in order to improve the signal-to-noise ratio and minimize undesirable phenomena (different attenuations or saturation). *In Italy and European countries the time constant is 50 µs while in the US it is 75 µs. SDR# defaults to 50 and does not have a selector for immediate change, which must be done manually as indicated in chapter SDRsharp.exe.config.*
AM	Transmission technique that uses a radio frequency signal as a carrier signal. Used by worldwide long wave/medium/short wave broadcasting stations and by civil and military aeronautical communications in VHF/UHF. ***Note:** there is no "AM stereo" mode, but you can use the free Sodira software for this purpose, after setting the RAW mode to SDR#. The same applies if you want to decode DRM.*
LSB/USB	Technique that foresees the modulation of a signal eliminating besides the carrier (as in DSB) also one of the two lateral bands. Used in the HF band (0-30 MHz) by utility and military services, by radioamateur radio in HF but also in VHF to transmit voice and data efficiently with small bandwidths.
CW	Morse Code. System to transmit letters/numbers/symbols by means of a signal in pre-established code made of points and lines. Used from always from the radioamateurs and a lot of military stations still today in the age of the digital. *In SDR#, however, with this mode it is possible in HF to tune and decode correctly (i.e., without tweaking the nominal frequency) many other emission modes. For example, we can try H24 with the RTTY-ITA2 50Bd broadcast on the Hamburg Weather frequencies at 7646 kHz and 10100.8 kHz.*
DSB	Use similar to AM but allows a higher modulation performance by suppressing the carrier and transmitting only the sidebands. *It can be used to tune stations with interference (together with IF Spectrum window where you can best configure the signal window by removing the interfering signal) or with new AM Co-Channel canceller plugin...* 
SAM *(even if it does not formally*	To enable it, just set the DSB mode and check the option “Lock Carrier” on this panel. The IF filter helps choosing which part of the DSB signal to use: LSB, USB or both.

exist it is fully supported!)	*A very high-performance and adaptive PLL is used, developed with the help from high-profile DXers that has no match in performance. It just locks everything even if the signal is barely visible in the RF spectrum. Even when the signal is completely gone, it finds its way to keep the lock until it reappears again. You don't get that nasty loss of lock in portable radios or other software.* *This can be combined with the "Anti-Fading" to improve SNR when there is no co-channel interference.*
RAW	Used for playback or recording of RAW IQ streams or with external decoding program, for example DReaM (DRM). *DReaM works with RAW mode by setting the its input to IQ, or using USB mode in SDR#.*

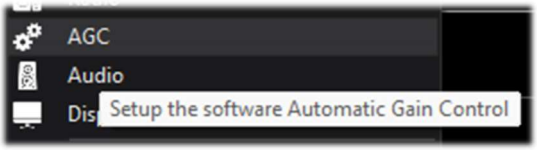
Key	Default	Features
Shift	0 (if you dont use UpConverter)	This box is useful only if you use an UpConverter; it is used to rectify the frequency tuned to the value entered. For example, if you use an UpConverter (with a 100 MHz oscillator) you set the Shift to -100,000,000. Without the Shift, when using an UpConverter to tune a 7 MHz signal, you should tune 100+7 = 107 MHz. With the Shift set, you can tune normally to 7 MHz without artifice.
Filter	Blackman-Harris 4	Set the type of filter used in the Fourier transform. It is used to receive the signal highlighted in the RF window *(where each filter has a different response curve and characteristic)*, The default "Blackman-Harris 4" is usually the best filter to choose without the need to change it, but there are nevertheless 14 other filters to try... 
Bandwidth	AM: 10.000 WFM: 180.000	This is the bandwidth (BW) in the window of the gray rectangular area. You can set it manually in this field or by dragging the edges of the window with the mouse.
Order	500	This cell changes the steepness value of the filter sides. With low values (from 10 to 50), the transition between the pass band and the out-of-band zone takes place gradually. With high values (from 100 to 500), the transition is immediate. The effect of this adjustment is audible in the audio signal. *Very high values, however, can cause AGC instability or less clean listening. You may increase the order of the filter when there are strong signals near your tuned area.* *However, using higher filter orders can cause a higher CPU load, so on slower PCs you should reduce this value.*
Squelch	OFF	Squelch is used to mute the audio when the signal strength is below a specified threshold. A high value requires a stronger signal strength to activate the audio. *The Squelch is only implemented for AM (Amplitude Squelch) and FM (Noise Squelch). SSB needs yet another type of squelches that is not implemented yet. It is especially useful in NFM waiting to hear speech and not just listening to*

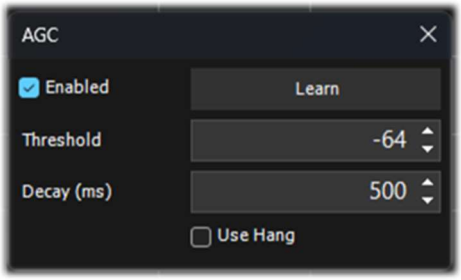
		background noise, ***but should be turned off when decoding digital signals (e.g. via DSD+ or DReaM software).*** ***Often I have collected opinions of difficulty, from friends and guys, in decoding digital signals with DSD+, perhaps it is better to use the original VAC rather than other similar software on the market!***
CW Shift	600	Mainly useful in receiving CW (Morse code) where you can set the offset between transmission and reception frequencies.
FM Stereo	***(in WFM)***	It will enable stereo output for WFM signals (in the 88-108 MHz band) from FM broadcasting stations, *but may worsen the sound of weak and distant stations.* *If a RDS signal is detected, the display (in item 11) will show the name of the broadcaster in a few round brackets.* 
Binaural 	***(in AM,DSB)***	Introduced since v1870, through the new "Binaural" checkbox (which activates or deactivates the function provided in AM/DSB modes only), audio is carried on different channels creating, especially for headphone listening, special "spatial diversity" effects. The term "Binaural," i.e., "two-ear," is not a new concept (please refer to the Internet for a historical discussion of the term), here we just need to know that it is not a demodulation method and is not classic "stereo" but can create specific and subtle listening effects under certain circumstances that, however, require some listener skill as well as, of course, a good pair of headphones. *AM (which is a monophonic signal) has two specular "sidebands" on either side of the carrier, evident on the FFT spectrum, but often the received signal will have lost some of the original symmetry due to propagation effects. Binaural tries to make up for the imbalances in the sidebands by presenting itself as a difference signal, and if there is slight background noise this is spread throughout the signal, giving an effective improvement in the signal-to-noise ratio. On strong signals (e.g., local stations) you can also try to activate the "Lock Carrier."* ***Remember that new tools require some practice and a lot of willingness to experiment!!!*** *The interesting part is the interaction of the binaural with existing IF filters, Anti-Fading, NR, Co-Channel Canceller, and other SDR# plugins. Now the Anti-Fading improves the SNR opportunistically by exploiting precisely the spatial diversity. Many other improvements in the DSP chain as well.*
MPX Fuzz ***(only in v1919)***	***(in WFM)***	The v1919 introduces temporarily this experimental control to improve and sensitize RDS under multipath and co-channel interference conditions. *The Fuzz algorithm helps decode RDS*

		from desperate signals, but beware: it can also degrade good signals… use it when really necessary!
Quadrature	***(in LSB,USB)***	The v1884 introduced this new quadrature audio output for LSB/USB modes only.
Snap to Grid ***(only up to v1918)*** *See also the "Step Bar" section.*	ON	The activation of the "snap to grid" and relative "Step Size" drop-down menu helps a lot the fast and correct tuning of the signals centering the correct tuning for each type of emission. For example in civil air band the channels are now spaced with the step of 8.33 kHz and this field, enabled with this value, allows the correct tuning by clicking directly on the RF Spectrum or Waterfall. *To use it with a non-TCXO dongle RTL-SDR, the PPM frequency offset correction must be set finely after at least 10 minutes after using the dongle, otherwise the frequencies may not be aligned on the grid with real frequencies.*
Lock Carrier **Sticky Lock** 	OFF	Active only in AM or DSB modes. Allow the Synchronous AM which can greatly improve reception and keep the signal perfectly locked, even if it is poor and unstable. *Try it in DSB mode, it makes all the difference for pleasant listening!* ***New introduced with v1920: Sticky Lock****. In DSB and used together with the Lock Carrier it locks onto the signal and never lets go...* ***Another feature introduced with v1892: Lock Carrier*** *which is achieved by using a special "Super PLL" with great resistance to lock loss. Basically, when the PLL loses lock, it starts another process at exactly the same stage where it was lost and continues to be maintained at the same frequency. When the carrier is available again, the PLL locks and resumes the signal without phase discontinuity or other inconvenience. This tool is especially useful for high-end DXing with intermittent signals.*
Correct IQ ***(only up to v1919)***	OFF	This setting removes the small, annoying center peak present with the dongles RTL-SDR R820T/R820T2. *Normally it should be activated.*
Anti-Fading	OFF	Use when "Lock Carrier" is activated. Leverages the symmetry of AM signals which helps in the presence of weak signals. *Activate it for better AM reception, but can increase CPU load.*
Invert Spectrum	OFF	If you use SDR# as a panadapter, some receivers may have the I/Q signals reversed and you must activate this option. *The I/Q signals, (or I/Q data), are a fundamental element of RF communications systems, often represents signals in the time-domain.*

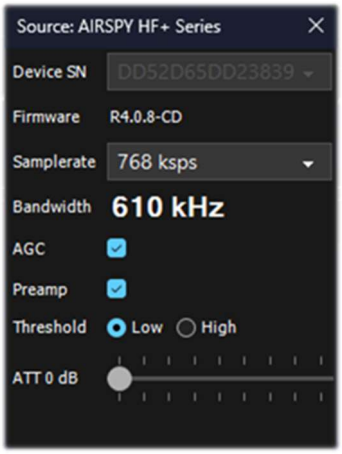
AGC

The function of AGC acts in real time on the amplification of the input signals by varying it in order to obtain an optimal level on the output on low signals and avoiding distortion on high signals.



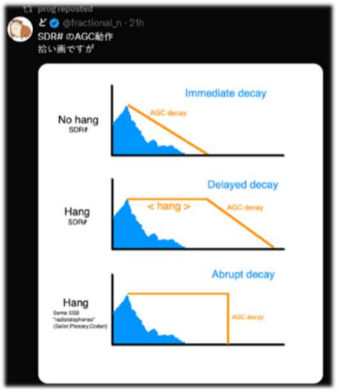


In the Airspy HF+ the AGC gain is fully controlled by the software running in the DSP, which optimizes the gain distribution in real time for optimal sensitivity and linearity. *So remember to turn both on!*



Since v1906 and v1919 (a) and then again in v1920 with the new R4.xx firmwares, the AGC has been optimized with a view to better handle QSB in AM (see Glossary) and automatic learning. In WFM mode the AGC is disabled because the FM signal is limited and its amplitude is constant. For NFM the AGC acts on the Audio output.

This feature is useful with weak signals with low modulation index. For AM, SSB, CW and RAW, the AGC acts on the narrow band IF, as usual.

Key	Default	Features
Enabled	ON	Activates the automatic gain control. The AGC will attempt to control the audio volume level so that loud sounds are not too loud and the same for low sounds. The default settings work well for voice signals. *It is especially useful to turn it on when listening to AM/SSB/CW mode because loud signals may be distorted.*
Learn		*It has been introduced since v1920.* This button is very useful for adjusting the AGC. To use it properly, you need to stand at a point in the spectrum/WF with little or possibly no signal and then press the "Learn" button which will automatically adjust the Threshold..
Use Hang		It allows you to change the default behavior of the AGC in its Threshold / Decay (ms) / Slop (dB) components, although in most cases the default values are fine. *Enabling it slightly changes the response over time and may be useful for some SSB or Morse signals.* *Recently appeared on the "X social" this useful image that helps better understand...* 

Threshold (dB)	-50	This is the threshold of intervention of the AGC. The signals below that level are not amplified, while those above are amplified at the level of the strongest ones.
Decay (ms)	500	Response time. High values delay intervention, too low values can cause an annoying sound effect.
Slope (dB)	0	Line slope for gain correction. *(only up to v1918 !!)*

There are many AGCs at different levels:

Analog
- RF AGC, which activates a 6dB stepped attenuator,
- IF AGC, which controls the IF gain just before the digitization.

Digital
- IF AGC, to make sure the data is scaled properly (by digital amplification) before sending to the computer.
- Narrow band AGC, which is the AGC panel controlling the signal that passes through the VFO filter.

Important Notes

The role of the Analog AGCs is to adapt the sensitivity of the front-end to the signals at the input. When enabling the AGC in the Source panel, you are activating both Analog AGCs.

- The RF AGC is performed before the IF AGC. Its recovery time is much slower than the SNR sampling rate. In addition, there is an additional 6 dB hysteresis.
- The Digital IF AGC is always activated and only starts acting when the strongest signal(s) in the IF spectrum exceed -6 dBFS.
  This mechanism ensures your signals are always scaled properly for unattended operations.
- When turning the Analog AGCs off, you can control the stepped attenuator manually, but then, it's up to you to determine which attenuation level is adequate for your signal at input.
  In general, push the attenuation until the noise floor is around -100 dBFS. Higher levels do not necessarily improve your SNR but will definitely reduce your available dynamic range. If in doubt, turn the AGC on and let it do the job.
- When turning the Analog AGC on, you will notice that you also have a "Threshold" option available.
  It is used to instruct the AGC to tolerate an extra 3 dB of signal power before setting the next attenuation level. "Threshold Low" means the front-end is "less sensitive", and "Threshold High" means "more sensitive". This is really useful when chasing marginal signals in presence of very strong blockers (~ 100 dB of difference).

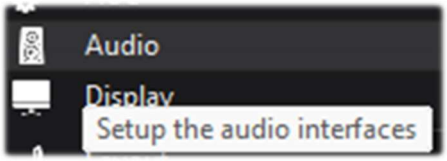
[a] Some notes on the new AGC introduced by v1919

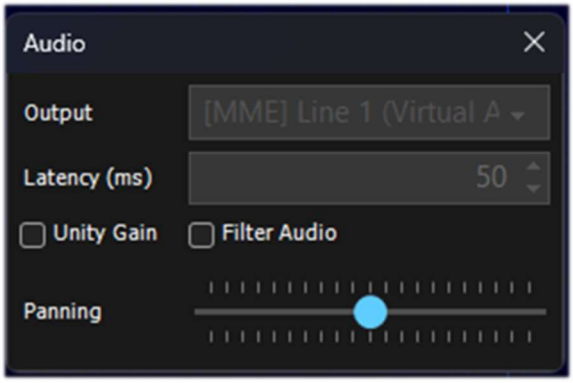
The goal of the new design is to let the user to adjust the threshold so that the level is regulated in a deterministic way without pumping. The level of the signals at the input of the AGC is that provided by the SDR hardware or IQ file, which means that the noise floor and spikes can be completely random. The output of the AGC targets -2 dBFS peaks to avoid occasional saturation (and subsequent distortion). The peaks above the threshold are attenuated so that the output target is not exceeded. Below that threshold AGC applies **a fixed gain**, which is what a few people intend to do by disabling AGC. There is no step transition around the threshold (continuous and derivable function). So, in the end, you may hear less noise and fewer NR artifacts when everything is adjusted properly.

Audio

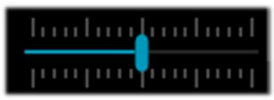
This panel adjusts the settings for audio processing.

Some important and fundamental changes have been introduced since v1920: now the samplerate automatically adjusts to the audio source/destination, so the related "Samplerate and Input" fields have been eliminated.



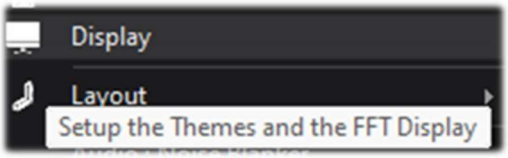


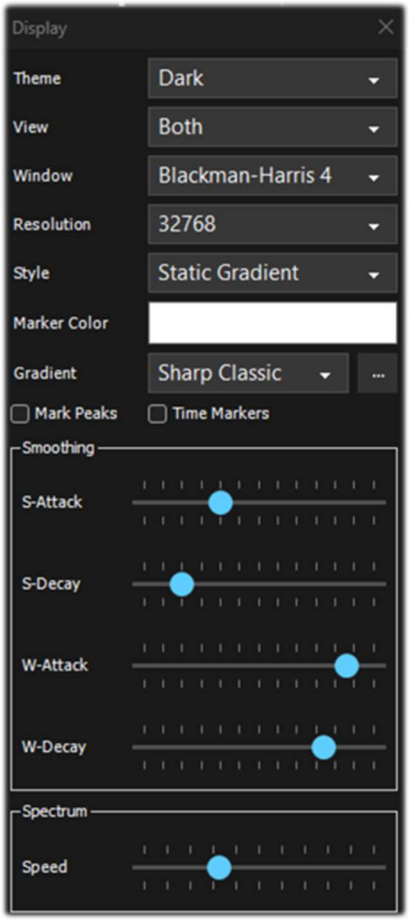
Key	Default	Features
Samplerate ***(only up to v1919)***	48000	Sets the sampling rate of sound card. Some decoding software may require you to set a specific sample rate. Usually the default value at 48000 samples/second should be fine for general purposes.
Input ***(only up to v1919)***	Sound card	The input sound card is highlighted in this field. *Normally it should not be changed, automatically detecting your card even if you are using SDR sound cards such as: SoftRock, Funcube dongle, Fifi SDR, etc.*
Output	Speakers	In this field you choose the output device from those available in your system: Speakers, Line 1, Digital Output and for the best audio quality: [Windows DirectSound...], [MME...] or the much better performing [ASIO...]. *Normally it defaults to the speaker line.* But what is ASIO? Audio Streaming Input Output is a low-latency communication protocol for digital audio signals developed by Steinberg. Thanks to it, a sound card can be able to record and play back audio without delays. Low-latency drivers, designed to obtain and manage incoming and outgoing audio streams with a small memory buffer (where stream exchange and digital/analog transformation takes place), allow for virtually zero delay! For audio devices that do not have native ASIO support, ASIO4ALL have been published universal and compatible ASIO drivers, however, not being native for many systems, this may result in slight reductions in performance but it is worth a try! If they are not already present in your Windows 10, low latency drivers can be downloaded from the site: ***https://www.asio4all.org/*** *You can then easily perform a test with these drivers enabled: tune in SDRsharp any broadcasting station and do the same with an external receiver… the audio coming out of both systems will*

		be virtually in parallel and without any delay as is the case with normal drivers.
Latency (ms)	**50** or lower with [Windows DirectSound] drivers **1** with [ASIO...] drivers	The latency value (expressed in milliseconds) is the time that elapses between the analog-to-digital conversion of the input signal, its processing and the digital-to-analog reconversion at the output. *It is advisable to keep this value as low as possible.* *The latest developments of SDR# (v1783) have almost halved the CPU/memory usage, while the latency is at the limit of what the hardware can do.* ***Since v1818 the latency has been drastically reduced again and now ASIO drivers work without problems with the value of 1 ms!***
Unity Gain	OFF	Normally it should be deselected as it sets the audio gain to the unit value of 0 dB. *This option is mostly useful when using the program as an IQ source for other programs or another instance of SDR# itself. This means it has to be used with "RAW", but the other modes work too, tho irrelevant. Basically, when using Unity Gain, the processing involves the tuning, decimation, filtering and eventually an AGC if you leave it enabled.* *No extra gain is added. The most of the desired signals at the input are very weak, it takes a good amount of decimation and filtering to make them usable.* *At the end you get something peaking at -120 dBFS. If you try to pipe the resulting IQ directly to VAC it will be quantized and effectively nulled out.*
Filter Audio	ON	Audio filter that improves the performance of all speech signals by removing the DC continuous component and reconstructing the audio by filtering out everything that exceeds the useful signal bandwidth. *See also the chapter "Audio Recorders", also it should absolutely be disabled when decoding digital signals (e.g., via DSD+ or DReaM).* ***The same applies to all other plugins (e.g. Audio Processors or Filters that act on the audio level), which must be absolutely deactivated when receiving digital signals (e.g. DMR, DSTAR, C4FM, satellites), otherwise they will result in incorrect decoding or dirty signals.***
Panning	middle	It is used to balance the audio between the left/right speakers. The  middle position of the slider is the standard one.Toward the left for left speaker, on the opposite side for right speaker. ***Warning: for use with DSD+ or other external decoders make sure the slider is in the center position!***

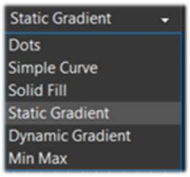
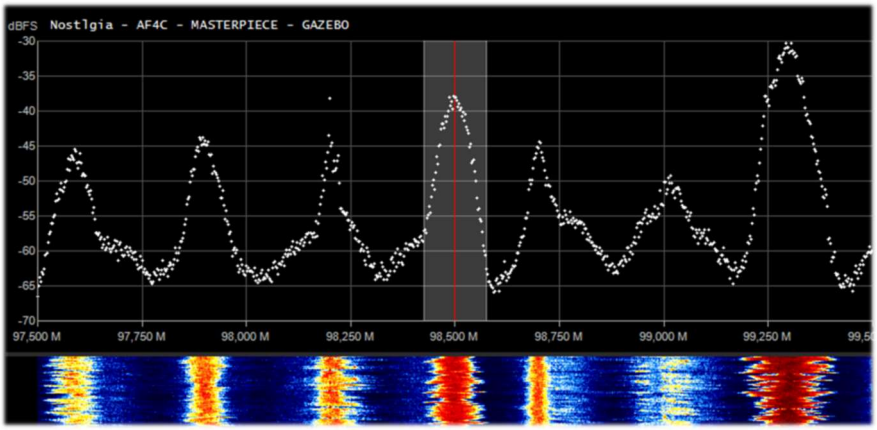
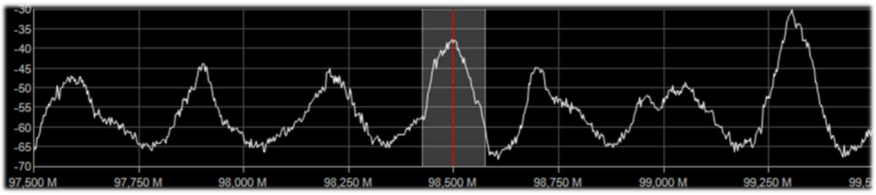
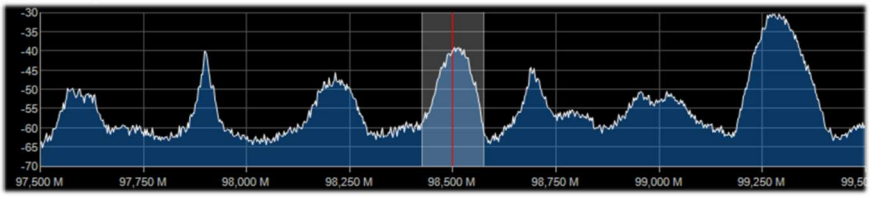
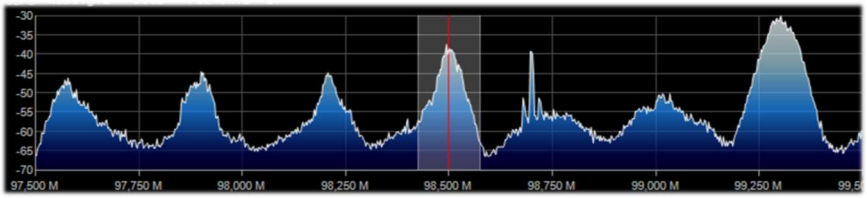
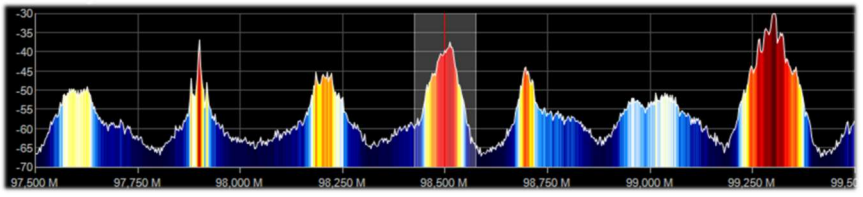
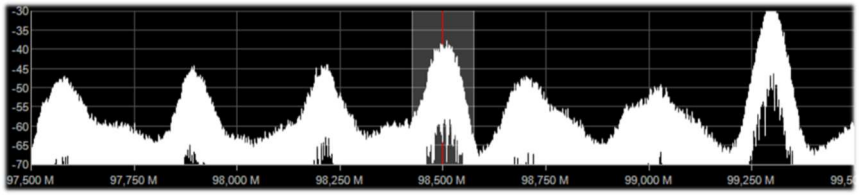
Display

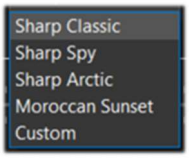
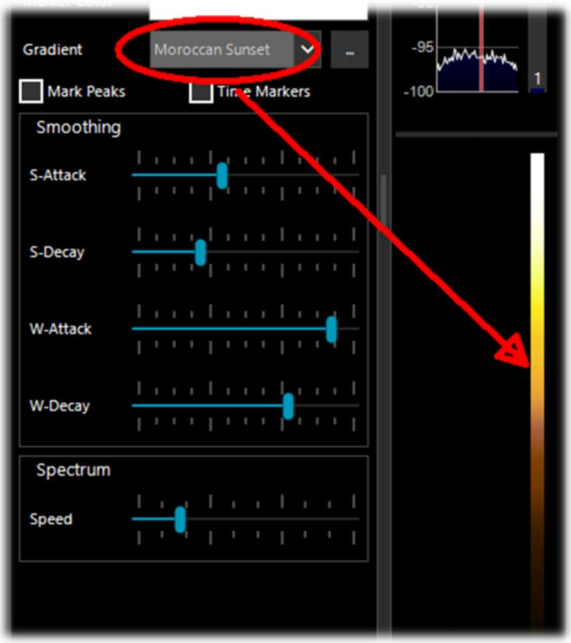
Display settings adjust the Theme, RF spectrum and waterfall screen options and other FFT settings.





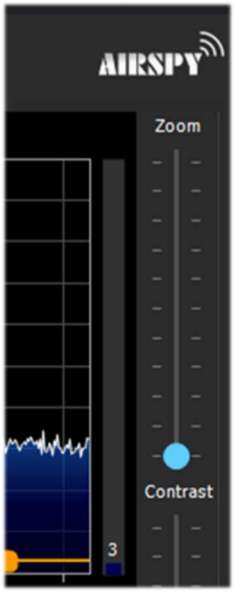
Key	Default	Features
Theme		Recently introduced with the latest graphic interfaces, allows you to choose multiple design layouts, many even in dark theme.
View	Both	Allows to set the display of the RF spectrum screen, waterfall, or only one of them, or none at all. *On older PCs it may be useful not to display the waterfall to avoid overloading the processing.*
Window	Blackman-Harris 4	Set the type of filter, *where each filter has a different response curve and characteristic*: the default of Blackman-Harris 4, it has balanced performance and is the best in most cases and should not be changed.
Resolution	32768	Adjusting the FFT resolution usually involves tradeoffs, as with most things in life... In fact, increasing the size of the FFT increases the frequency resolution, but so does the update time and CPU processing load! The result could be a more detailed display (in the RF Spectrum and waterfall) however with a slower update rate. A more powerful processor certainly can help but the adjustment will always be a compromise as I said at the beginning. *The use of a higher resolution can be useful when fine tuning, as you can see the peaks and signal structure better. Beware, however, that high resolutions can*

		slow down the PC and can cause problems especially with older machines. Normally, if the PC can handle it, at least the value of 32768 should be used.
Style 		Allows you to choose different styles of waterfall signal representation: **DOTS**  **SIMPLE CURVE**  **SOLID FILL**  **STATIC GRADIENT**  **DYNAMIC GRADIENT**  **MIN MAX** 
Marker Color		It allows you to change the color of the marker on the waterfall by simply clicking on the Windows color palette.

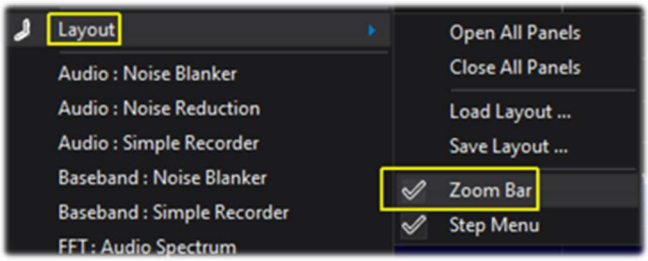
Gradient 		Allows customization of the color palette used in the waterfall. *Youssef suggests for High Dynamic Range applications to modify the file "SDRSharp.config" with these values:* *<add key="waterfall.gradient" value="FF0000,FF0000,FBB346,FFFF00,FFFFFF,7AFEA8,00A6FF,000091,000050,000000,000000" />*  Initially there was only one palette encoded within the configuration file, but starting with v1818 you can choose some pre-set gradients: Sharp Classic, Spy, Artic, Moroccan Sunset and Custom. *In order to immediately evaluate the most suitable one for us, once chosen from the menu, I suggest viewing the vertical bar on the right with the complete palette of colors represented.* The button  activates the "Gradient Editor" to further customize the color palettes.
Mark Peaks	OFF	Allows to highlight a circular marker on each signal peak on the RF spectrum window.
Time Markers	OFF	Displays a time indicator on the left side of the waterfall screen to date the signals transmission. *By definition it is set to 5 seconds.*
S-Attack / S-Decay		Changes the uniformity and average of the received signals in the RF spectrum display. *Set them halfway.*
W-Attack / W-Decay		Changes the uniformity and average of the received signals in the waterfall display. *Set them halfway.*
Speed		Changes the refresh rate of the RF spectrum and waterfall. *Never keep it at full speed! In fact, to further reduce CPU usage simply set the speed to minimum here.*

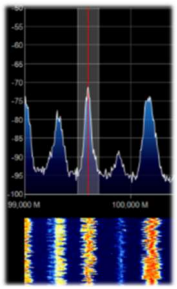
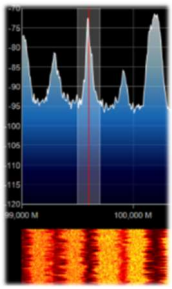
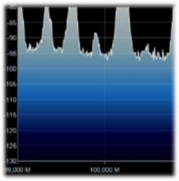
Zoom Bar

The "Zoom Bar" allows you to manage the four vertical sliders of the settings in the RF Spectrum/Waterfall display. It was originally located on the right side of the RF Spectrum - items 14/17.



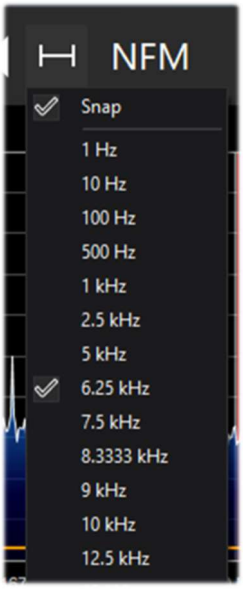
Since v1892 there are two options for use: the newer one allows you to have the entire control minimized (always next to the RF Spectrum). You click the "Zoom" button to open it temporarily... Instead, those who would like to use it as it was previously just flag the relevant control in the Menu / Layout / Zoom Bar.



Key	Default	Features
Zoom	down	Moving this slider up will magnify the RF spectrum and waterfall around the tuned frequency. However, the higher the magnification, the lower the resolution will appear. An alternative to zooming is to reduce the sample rate or use the decimation function in the Source panel.
Contrast	down	Adjusts the contrast of the waterfall. Moving the cursor upwards the signals will be distinguished from the background noise, *but do not exaggerate and avoid saturating by having an screen all yellow/orange or red...*
Range	down	Changes the level in dBFS on the left axis of the RF spectrum window. *You should adjust it so that the noise floor threshold is very close to the bottom of the RF spectrum window.* This will make the RF spectrum and waterfall signals more readable, making weak signals easier to detect. *Correct*  *Wrong* 
Offset	down	Adds an offset to the dBFS level range in the RF spectrum window. The offset is added to the upper value of the dB level range in the RF spectrum.  *Normally you do not need to adjust it, unless you need additional contrast on weak signals in combination with the "range" adjustment.* *Adjust it so that the height of the signal peaks are not clipped at the top of the screen.*

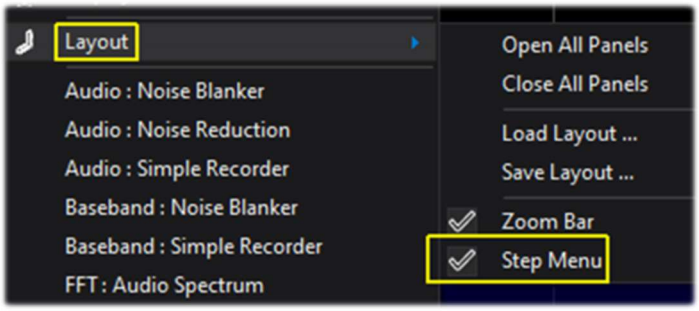
Step Bar

Since v1785, the "Step Size" field (next to the "Snap to Grid" item) is no longer present in the "Radio" panel, but the new "Step Menu" has been created now on the right-hand side of the VFO later modified in v1899 and again with the new encapsulated menu as a drop-down item of a smaller button to save UI estate from v1906 that now looks like this.



Possible choices are tuning steps between 1 Hz and 500 kHz or with free tuning, which is done by not flagging the first item "Snap" which is used to freely tune any signal regardless of the tuning steps specific to many in-band services and their emission modes.

From v1892 you can also enable or disable its functionality from Menu / Layout / Step Menu.

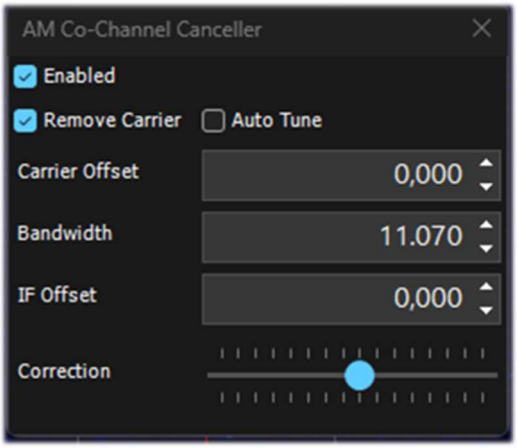


*To use a new step not foreseen it is sufficient to edit the SDRsharp.config file at the key starting with **<add key="stepSizes" value=..."** and insert the new value, example "3.125 kHz".*

Another very interesting and fast possibility is to use the mouse: just position yourself in the Waterfall or Spectrum window and with the central wheel of the mouse rotate it forward to advance with the frequencies of the pre-set Step or on the contrary, rotate it backwards to decrease the frequency.

Co-Channel Canceller for AM/FM

Following numerous user requests to improve AM DX reception in medium and short waves in the presence of isofrequency interference, the AIRSPY team developed the first and innovative "Co-Channel Canceller" (or CCC) algorithm. ***This unique, free and constantly updated plugin is not found in any other software!***

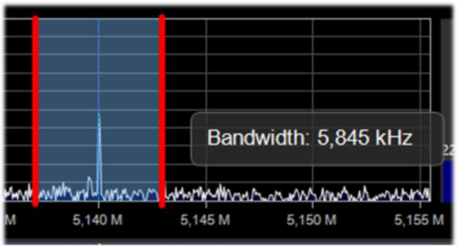


Leaving aside here the technical dfficulties behind the development of such a tool (still patent pending), I can only point out that the plugin latches onto the main (dominant) carrier and clears all the correlated frequency bands around it. If the signal is distorted (and has bad correlations) it tries to solve the problem using some specific and innovative algorithms.

There are two separate plugins, one for AM mode and another for FM, which not only recover audio plagued by interference, but can also be combined with the other plugins to combat QRM, QRN, and anything that can impair signal reception.

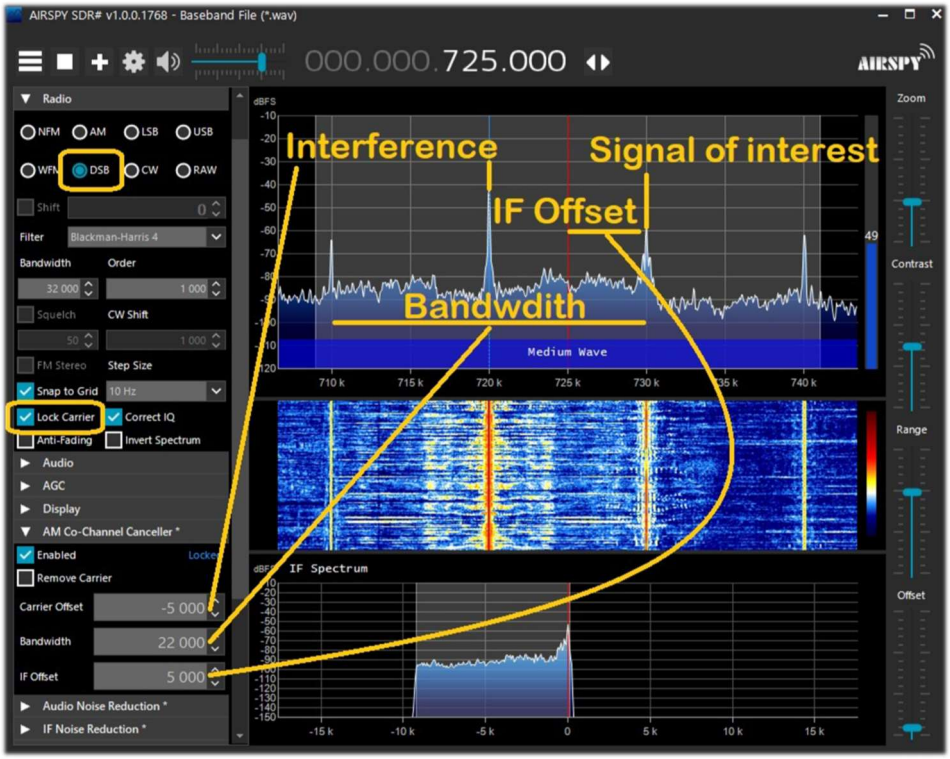
Co-Channel Zero-Offset also works, meaning you can remove the strong local station and listen to the DX station on the same frequency.

During a listening session it can often happen to find the right conditions for using these plugins, of course they are not a normal condition of use, but in the case of interfered stations to try to detect a DX signal devoting some time and attention because the procedure can be a bit laborious, at least the first few times...

Key	Features
Remove Carrier	Command to enable carrier removal. *For example, if we enter the value 4.500 we indicate that the interference is at 4.5 kHz from the station of interest.*
Auto Tune	Added since v1900 is a new optional control in the "AM Co-Channel Canceller" for quick operation without the "Micro Tuner."
Carrier Offset	Allows you to adjust the interference level of the "Co-channel" with respect to the signal of interest.
Bandwidth	Allows the bandwidth to be adjusted with respect to the signal of interest.  *This value can be easily changed by simply engaging the vertical red lines of the RF Spectrum and/or waterfall with a mouse click and move left/right to widen or narrow the bandwidth.*
IF Offset	Allows the IF Offset to be adjusted with respect to the signal of interest.

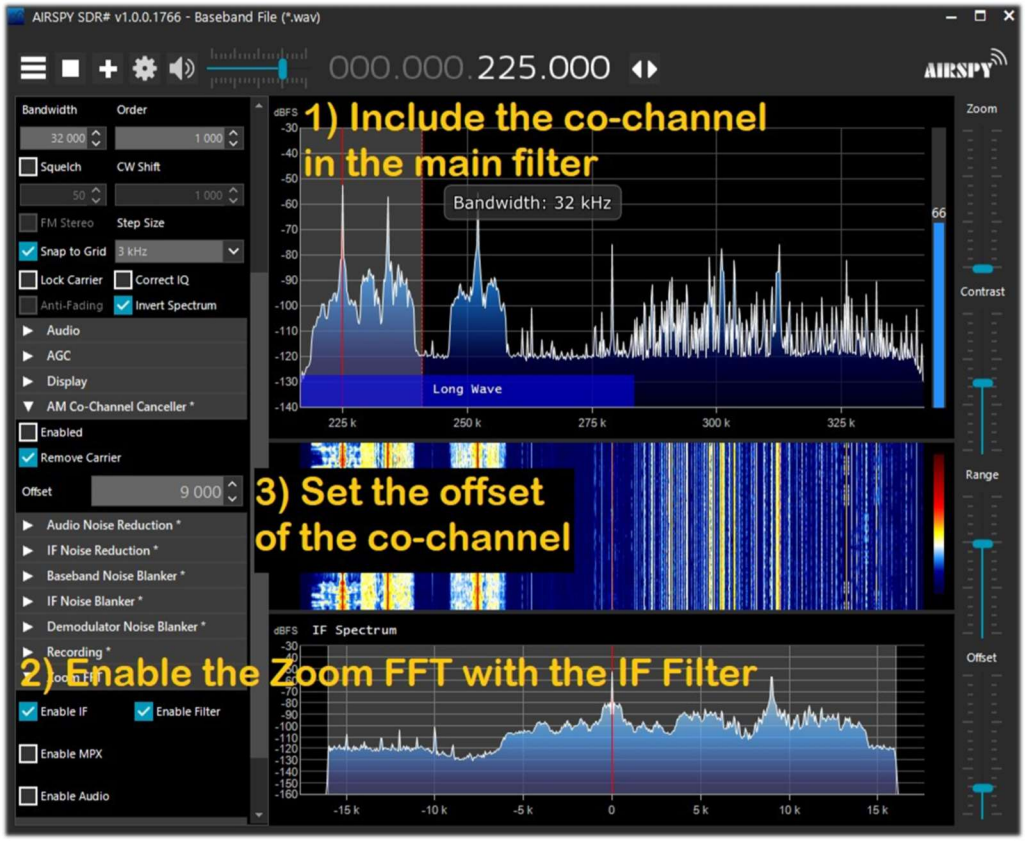
How to use the new controls:

- Make sure that both the desired signal and interference are selected in the main filter.
- If necessary, tune the VFO between the two signals.
- Adjust the IF offset to shift the signal of interest to the VFO frequency.
- Adjust the bandwidth of the Co-Chanel rejection so that only the interference and its splatters are covered.
- Set the carrier offset relative to the VFO.



Summarizing this in a few basic steps:

1. Include the Co-Channel in the main filter by enlarging the Bandwidth sufficiently.
2. Set the offset of the Co-Channel
3. Enable "IF Multi-Notch" to perfect the operation.



Let us now try a "practical paper example," although the procedure may vary slightly in specifics. However, later on I will report some much more explanatory videos!!!

A) local station with very strong signal, example at 819 kHz
B) DX station at 810 kHz interfered and unintelligible
1) Tune the station (B)
2) Enlarge the filter window from 810 to 820 kHz to include the carrier to the right of the local signal (A)
3) Enable the "Co-Channel Canceller AM" plugin and flag the "Remove Carrier" field with Carrier Offset value at 9,000 (to indicate that the interference is 9 kHz from the station). The function will lock and show in blue "Locked" on the right side, and in the RF Spectrum you will see a blue vertical line above the carrier to be removed from the signal (A).

4) Enable the "IF Multi-Notch" panel while remaining always tuned to the signal (B): in the window narrow the bandwidth (BW) to exclude the interfering carrier: now you can listen and appreciate the new result!

In recent versions, the "AM Co-Channel Canceller" algorithm has undergone significant implementations:

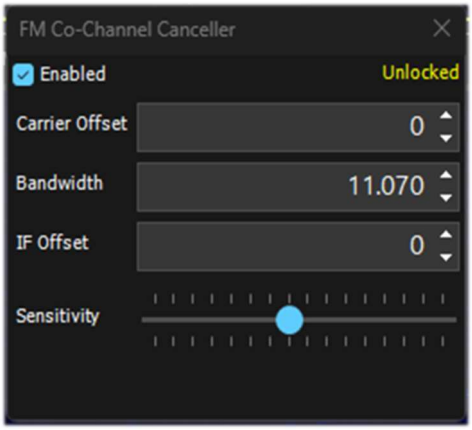
- ***Deeper rejection***
- ***Improved quality of remaining signals***
- ***Improved resistance to phase and amplitude distortions***
- ***Simplified user interface***
- ***Lower CPU utilization***

In addition, the new "Micro Tuner" plugin is used to select the signal to cancel, but it is also possible to disable it and manually select the offset of the signal to cancel in the CCC interface.

Some demonstration videos of the "Co-Channel Canceller AM" algorithm during DX sessions:

MW	**https://www.youtube.com/watch?v=KnGHun1E8Us**
	https://twitter.com/i/status/1784908222421184617
SW	**https://www.youtube.com/watch?v=N5rEnmCQun0**

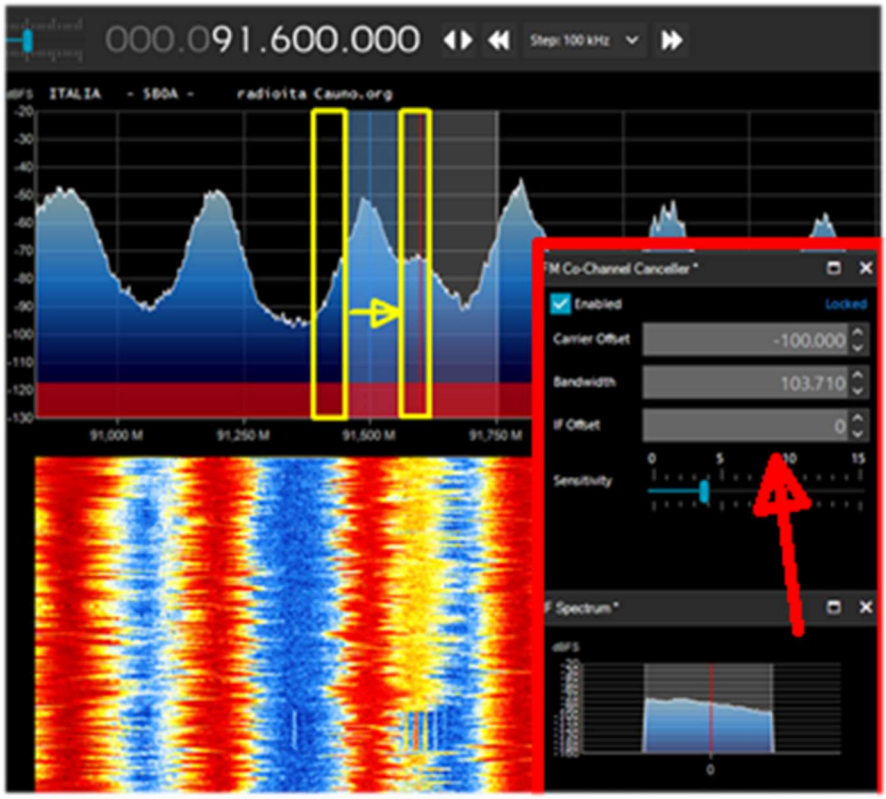
Similarly works the equivalent "FM Co-Channel Canceller" plugin, *still improved in the latest versions!*



From opinions gathered from friends and acquaintances it appears that it works well for waves arriving overland (FM and MW), contrary to what Dxers tested for signals via ES or transatlantic in MW.

I did some testing here in the city center where the WFM signals are really strong and present in the whole spectrum. In some cases with this plugin it was possible to receive as many as two isofrequency stations (where the strongest station however was not too "dominant" compared to the secondary one).

In this example we see how to set the panel to listen to a WFM station with a very strong signal at 91,500 MHz and a weaker station tuned to 91,600 MHz (vertical red line at BW center at about 100k).
You enable the plugin, setting the Carrier Offset to -100,000, to clear the signal at 91,500 MHz (vertical blue line on the left), adjust the left side of the filter a little in the IF Spectrum window, and slightly change the position of the "Sensitivity" slider to the desired effect... after a while it even manages to appear in RDS the name of the station with its PI code.



Not only does it work "live" in real time, but it also works equally well with previously recorded I/Q files, give it a try!

Always make sure that the cancelling filter covers the affected signal entirely. For example, if you leave the left side of the signal strong, you cannot get rid of the right side covering the signal.

In the following audio sample by Peter, titled "BBC Radio 2 cancelled during an Es opening from Italy," one can hear the marked effect of the "Co-Channel Canceller FM" plugin at work: **https://www.youtube.com/watch?v=mAmmy3Y_rQs**

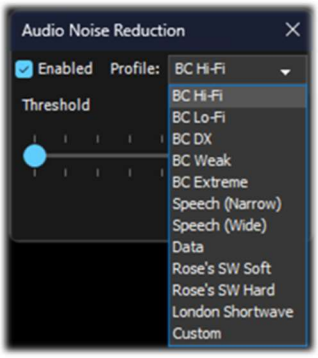
Very unique and comprehensive long video by Paul W1VLF with the CCC FM:
https://www.youtube.com/watch?v=FvshoNfv3ag

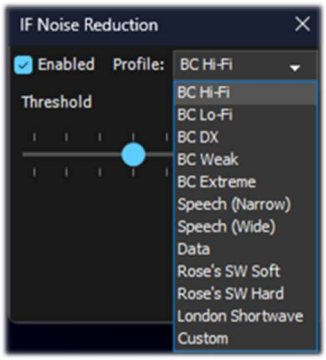
Audio/IF Noise Reduction (NINR)

When listening to speech signals, which are often weak and noisy, it is very important to activate digital noise reduction.

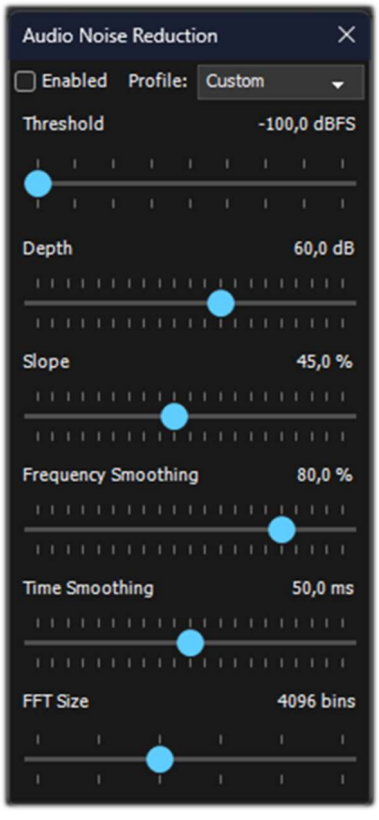
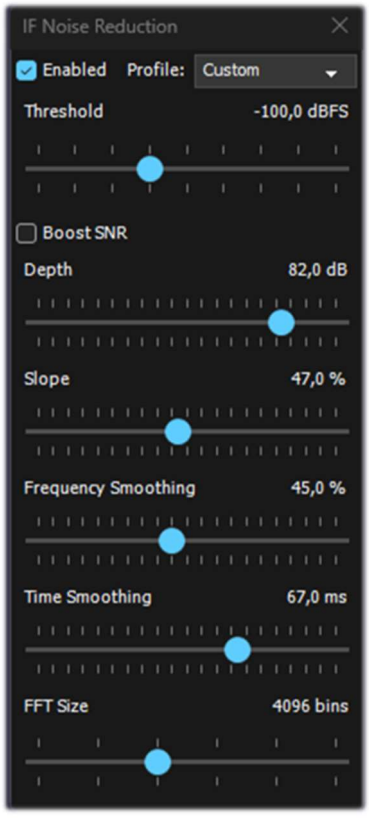
In SDR# two Noise Reduction options are available: Audio and IF. The Audio option uses the noise reduction algorithm on the audio output signal, the IF option on the IF signal.

Since **v1856**, the old noise reduction processor has been replaced with a brand new algorithm, **Natural Intelligence Noise Reduction (NINR)** which is constantly being updated and improved. This results in better intelligibility with fewer artifacts, deeper noise cancellation and lower CPU utilization.





Once the plugin is enabled, the "Threshold" slider is used to control the incisiveness of the applied algorithm, and many already optimized and predefined profiles can be used**: BC Hi-Fi, BC Lo-Fi, BC DX (**ex Speech since v1911**), BC Weak** (new with v1911), **BC Extreme** (new in v1919)**, Speech (Narrow/Wide), Data, Roses's SW Soft/Hard, London Shortwave.** With **Custom**, individual components can be customized as we see in the table at the level of: Threshold (dB), Depth (dB), Slope, Frequency Smoothing (%), Time Smoothing (ms), FFT Size (bins)

Audio Noise Reduction	IF Noise Reduction
	

But when is it appropriate to use one or the other or together at the same time?

There is no general rule that fits everything and everyone, everyone will have to experiment personally according to their own needs and the specific levels of noise present. For Airspy R2 owners, you can also try reducing the gain a bit...
To get started, you can use the predefined "Profiles" or try the "Custom" directly, which allows more flexibility and user control. One must take some time, however, to test all the parameters of this profile, which can give better results for specific types of signals and in various emission modes. Probably the best on the market today!!

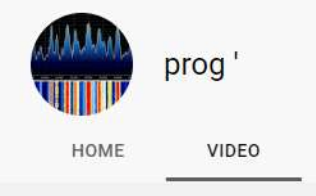
With the help of some friends, I have collected and summarized some impressions in the following tables.

Control	Scale range	Features
Audio Threshold IF Threshold	da -100 a -20 dBFS da -160 a 0 dBFS	User-controlled threshold setting. At "aggressive" settings, signals may have partially compromised quality.
Boost SNR *(in IF Threshold)*		Do not use Boost SNR with the AM detector. The instrument does not "boost" the carrier as before, and if you boost the sidebands to a higher level than the carriers, you will get poor AM demodulation with the conventional demodulation technique. Instead, use DSB with lock. *Try to use it only on very weak radio signals, such as with SNR <5 dB.*
Depth	0 to 100 dB	Depth of intervention: little change in quality was noted across the 0 to 50 dB range, with 50 dB applying the greatest effect.
Slope	0 to 100 %	New additional parameter initially introduced by v1859 and modified with v1897.
Frequency Smoothing	0 to 100 %	From v1897 this control replaces the previous "Smoothing" control. This control consumes a little more CPU at high percentages.
Time Smoothing	0 to 100 ms	From v1897 this control replaces the previous "Attack/Decay" controls. Most of the effect occurs at the beginning of the scale.
FFT Size	1024 to 32768 bins	Bins (or spectrum samples) define the frequency resolution of the window.

The only settings that can affect CPU consumption are "Smoothing" and "FFT size," the rest seem to have no appreciable effect on CPU utilization.

AM	The IF N.R. eliminates RF noise that can be anywhere in the signal. This has no effect in WFM or NFM with high modulation index because the signal is distributed over a large bandwidth, but with linear modulations such as AM and SSB, it can dramatically improve SNR by identifying the parts of the spectrum that do not contain signal and attenuating them. *Using the "Hi-Fi" profile on an AM broadcast, I forgot that the filter was on until I turned it off and all signals returned to the usual local noise...*
CW	The "Narrow Band" profile gives the best results with CW. It can also be used by selecting the "Custom" profile.

	It really works very well and a weak morse signal becomes good. One was able to pull out a couple of regional NDBs that otherwise would have been practically submerged in noise.
NFM	Joint use of both IF and AF Noise Reduction is necessary. The IF N.R. can still work for NFM signals, but it is mainly intended to be used with the other modes. I did some limited testing of AF Noise Reduction on NFM by tuning some stations in the 160/170 MHz range where the electrical noise is very high and annoying. *The AF N.R. audio is better for FM modes because it eliminates the hiss, which has mostly high frequency components in the audio spectrum.*
SSB	The new algorithm definitely helps the S/N on weak signals in SSB that are immersed in noise and thus become much more intelligible. *The IF N.R. eliminates RF noise that can be anywhere in the signal. This has no effect in WFM or NFM with high modulation index because the signal is distributed over a large bandwidth, but with linear modulations such as AM and SSB, it can drastically improve SNR by identifying the parts of the spectrum that contain no signal and attenuating them.*
WFM	The Audio N.R. option is recommended for wideband FM. *AF N.R. audio is better for FM modes because it eliminates hiss, which has mostly high-frequency components in the audio spectrum.*



VIDEO COLLECTION

Extreme de-noising using State of The Art DSP Technology from Airspy

https://www.youtube.com/watch?v=L5C3RpL9tXc&feature=youtu.be

Advanced Radio Signal Restoration Technology by Airspy

https://www.youtube.com/watch?v=WHeAKY2IpgA&feature=youtu.be

Airspy SDR# (rev 1888) with dramatic effects of the new DSP:

https://www.youtube.com/watch?v=hevMQblTQAc

SDR# (SDRSharp rev 1860) NINR Noise Reduction on US 20 kHz AM Broadcast:

https://www.youtube.com/watch?v=5IwV2BW_Mp0

Airspy HF+ Discovery / SDR# Daytime RX of Medi1 from Amsterdam:

https://www.youtube.com/watch?v=1WqNR9e_G3s

Or to play an IQ file (296 MB):

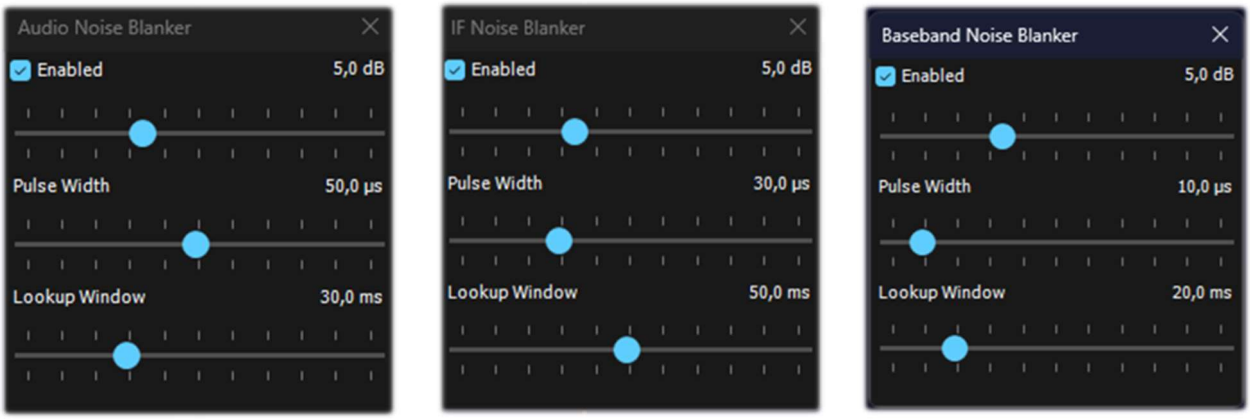
https://Airspy.com/downloads/IQ_Training_27-Sep-2017 203114.151_305000Hz 000.wavzip

In general, the whole PROG video collection:

https://www.youtube.com/channel/UCLxV5qQH52VcN6HfXEWC83Q/videos

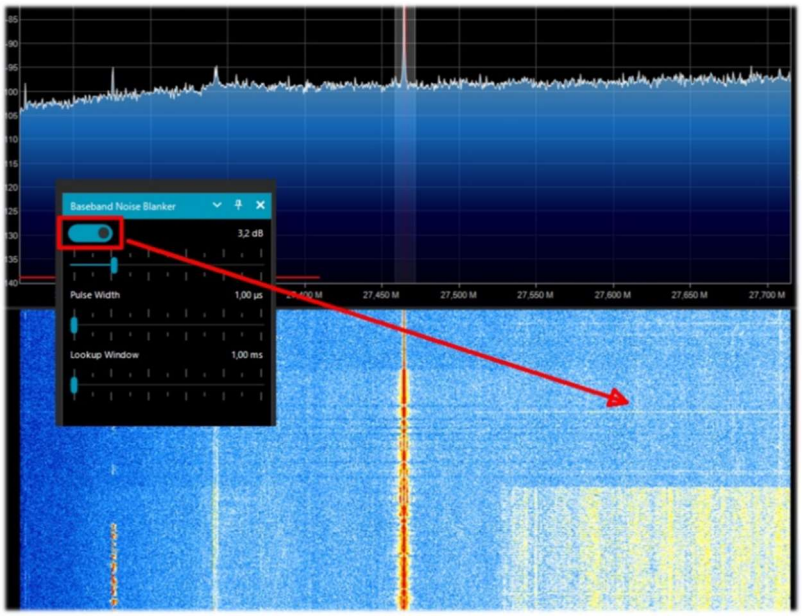
Audio/IF/Baseband Noise Blanker

The Noise Blanker is a function that can be activated to try to reduce impulsive and pulsating noises such as that coming from sources such as some motors, power lines, power supplies of various kinds.



This function can really make a difference, especially in the HF band, when receiving weak signals immersed in noise.
The algorithm tries to remove those tracks that have large pulses inside of them. In SDR# there are three different types: “Audio: Noise Blanker” operates within the tuned area, “IF: Noise Blanker” operates on the IF signal, “BaseBand: Noise Blanker” operates over the entire RF spectrum and removes pulses from the FFT and Waterfall.

The pulsing noise can appear in different forms at different stages. It’s important to know that at the Baseband stage, you have more opportunity to eliminate very short pulses without affecting the rest of the processing. If the pulses are “fat”, ie. have a longtime window, you can eliminate them at the IF stage with better results, but it’s less optimal than the Baseband NB with short pulses. Finally, at the Audio stage, the pulses will definitely take something from your signal, but that’s the last resort when everything else fails. Consider there is more averaging effect happening between the Baseband, IF and Audio stages, which will spread the pulses in time as the processing goes. The earlier you can cut the offending pulses, the better.

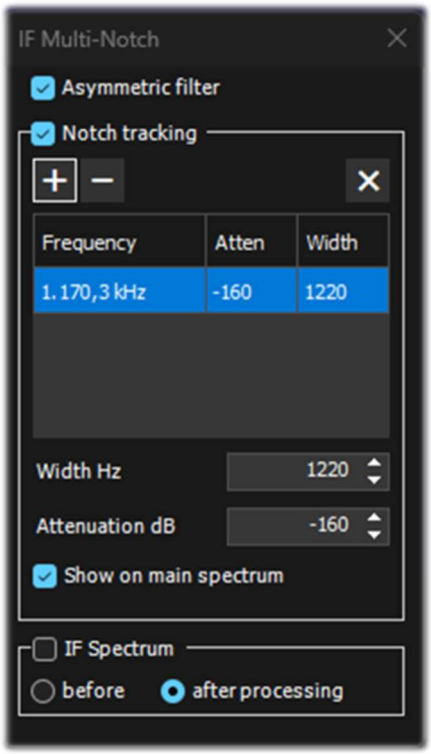


There are obviously no preset values or thresholds, *so you need to gradually move the various sliders until the pulsing noise disappears or reduces without distorting too much the received audio (for this pay close attention).*

In the example, you can immediately see the difference of activating the "Baseband Noise Blanker" in HF where the noise can be mitigated by removing the noise pulses as seen in the waterfall...

IF Multi-Notch

In a nutshell, how to eliminate a great many annoying and unwanted signals at will!

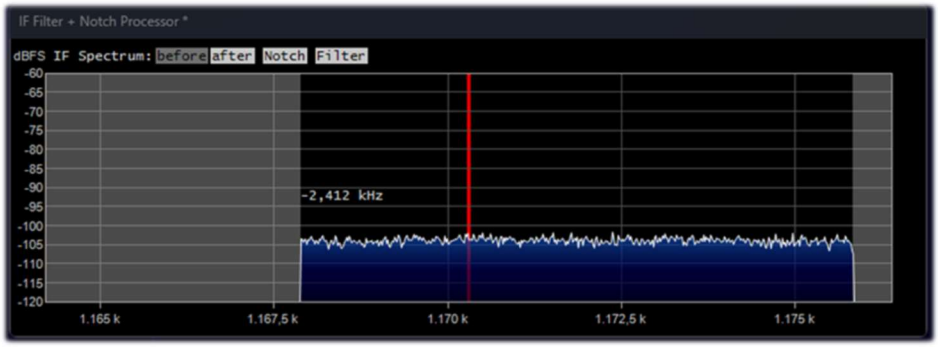


Just for memory's sake and as many will remember, in previous versions of SDR# the plugin was called (**"IF Notch + Filter"** and before that **"IF Processor"** (*)). Through multiple functions it allows us to eliminate entire portions of frequencies that can create serious listening problems for us in some circumstances.

*(*) Until v1899 the plugin "**SDRSharp.DigitalIfProcessor.dll**" in the plugins folder made the IF Processor available. Since v1900 this service is rendered internally and therefore the presence of that DLL only creates anomalies including not closing the SDR# itself among the active processes. It is recommended do not installing or to delete if it is present.*

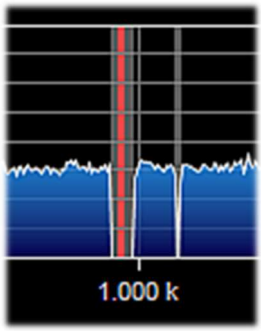
"Notch tracking" data (Frequency, Atten, Width) are automatically saved in the "notches.xml" file in the program directory.

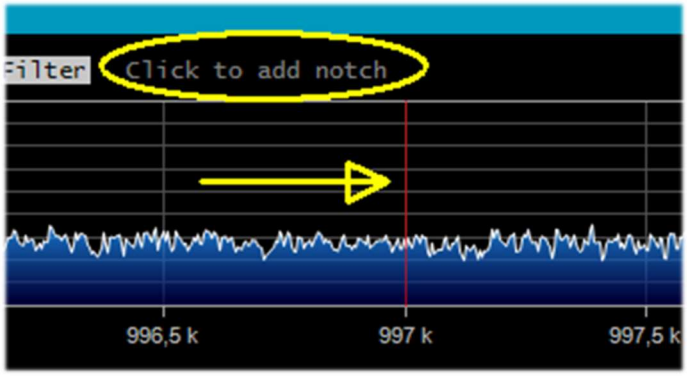
The power of its **"Asymmetric filter"** (also enabling the flag on "IF Spectrum") allows you to select which side of the signal to operate on in the "IF Filter + Notch Processor" window. In the following screen, I resized only the left side of the IF spectrum with the mouse, reducing it by -2,4 kHz.



"Notch tracking" allows us to configure various notches each with its own bandwidth (Width in Hz) and attenuation (in dB) characteristics.
In the example opposite we can see two active notches: the first at 999 kHz very pronounced as bandwidth and the second at 1002 kHz of only 300 Hz.

The + button inserts a new notch, while the − button deletes the highlighted one.

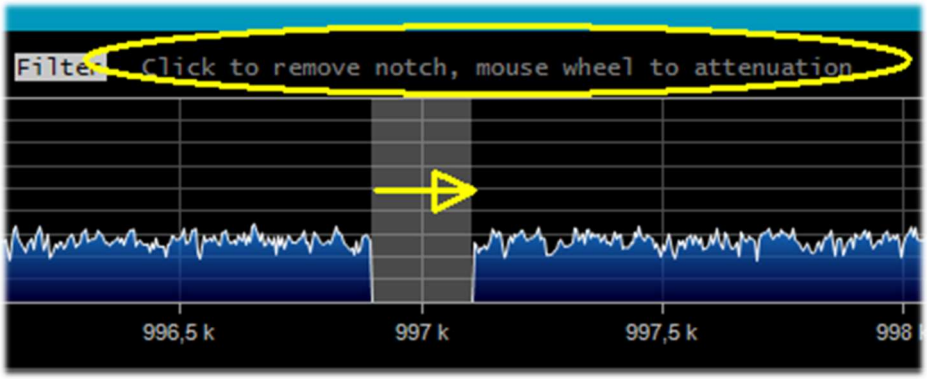




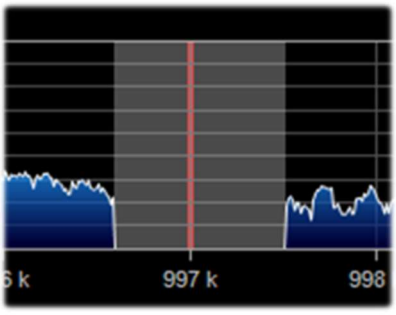
In an even faster and more practical way, you can do the same thing directly on the **"IF Filter + Notch Processor"** window as suggested by the yellow highlighted message.

In this example, wanting to insert a notch filter at 997 kHz, we will simply click directly on the IF Spectrum window at the thin red vertical line...

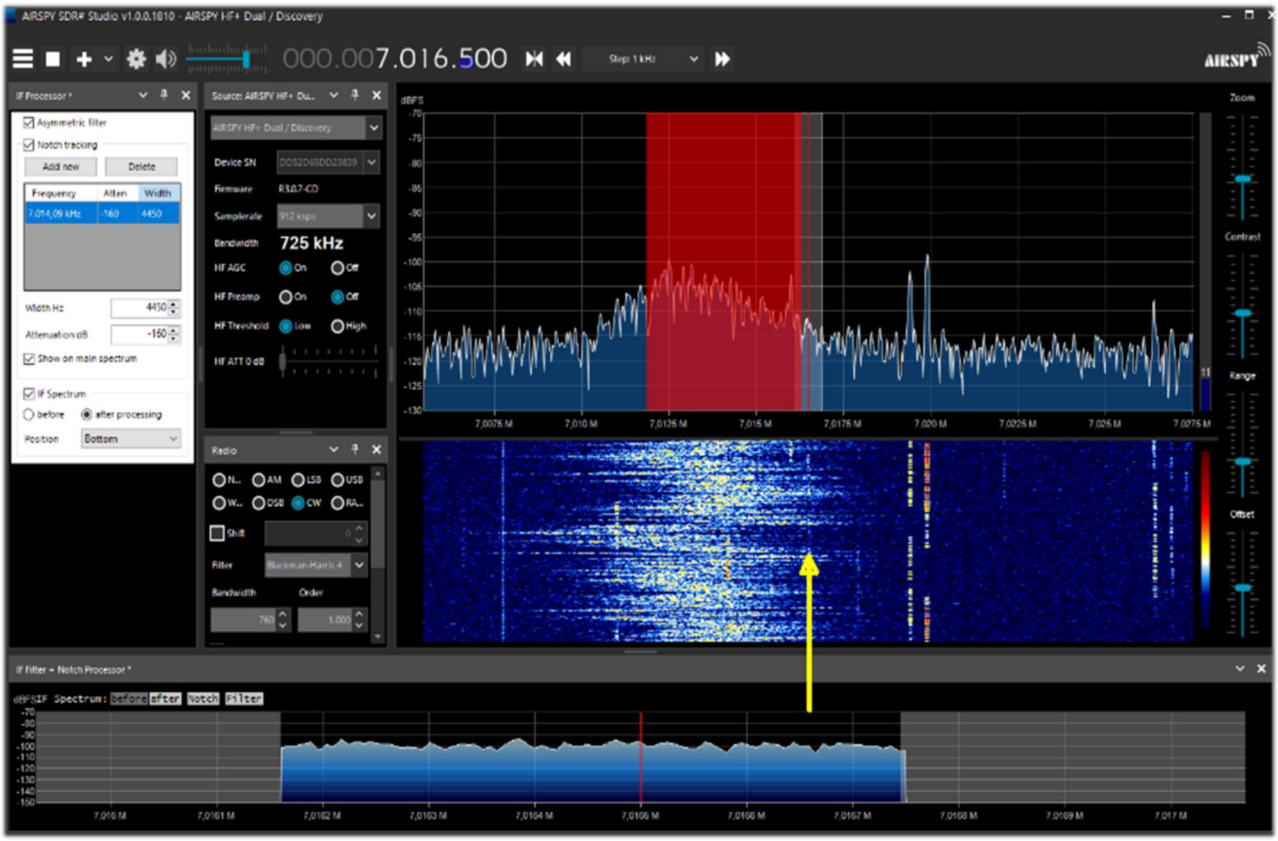
In the same way, you can eliminate the notch filter by clicking on it or changing its bandwidth with the mouse.



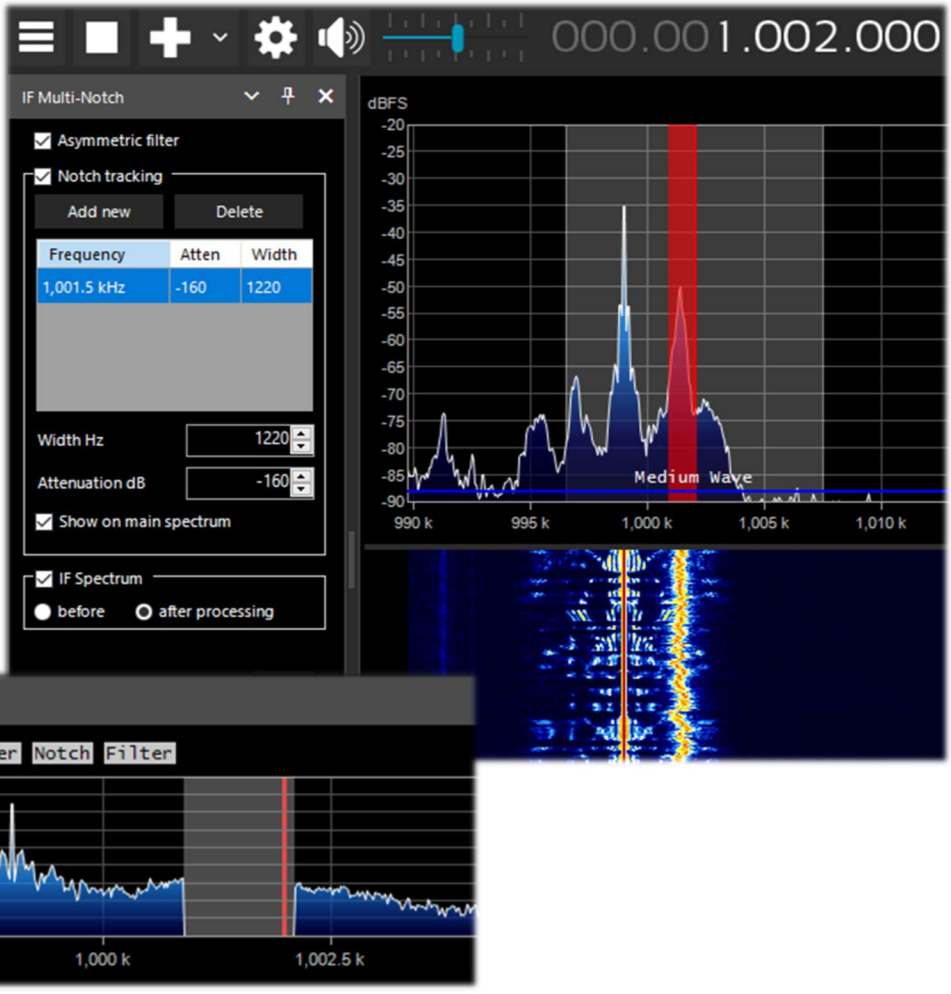
Instead, turning the mouse wheel, varies the attenuation until to -160 dB.



On the next page we will look at some practical examples of applying the Multi Notch filter in some different situations experienced in the past on HF and Medium Waves.
In this other example, the red portion of the filter, several kHz wide, where the extreme variable noise made it difficult to receive the very weak CW signal at 7016.5 kHz (shown with yellow arrow)



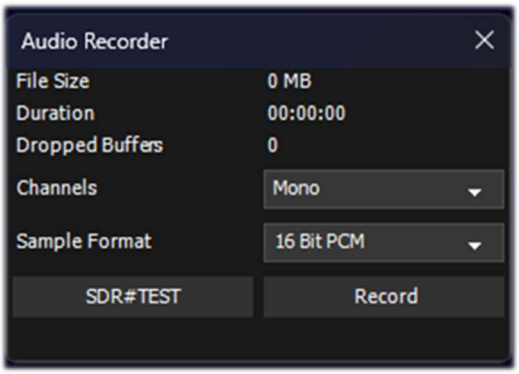
Another significant example where the filter allows me to listen perfectly to the local RAI station on Medium Wave at 999 kHz, which is heavily disturbed by heavy variable noise just a little higher in frequency.
With a notch set at 1001.5 kHz, with strong attenuation and wide BW solved the problem in no time!!!



...

Audio Recorders

To record audio samples of what we are listening to and playable later with any player we have as many as two possibilities, seemingly similar but with somewhat different characteristics.

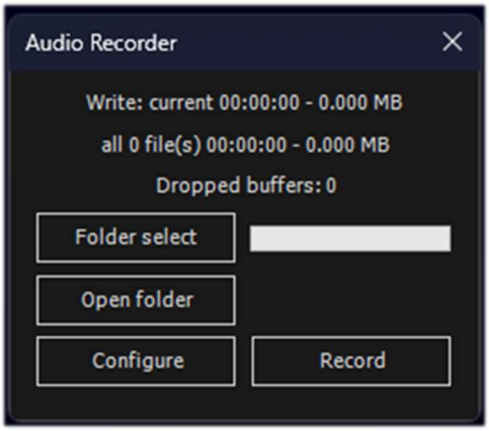


Audio Simple Recorder: is the one built into SDR# from the v189x.

You initially configure the destination directory for the audio files by clicking on the button at the bottom left ("audio" in my case).

Then you can choose between Mono and Stereo and the format between 8 and 16 bit PCM or 32 bit IEEE Float.

Very practical and fast! However, it is not connected to the squelch...

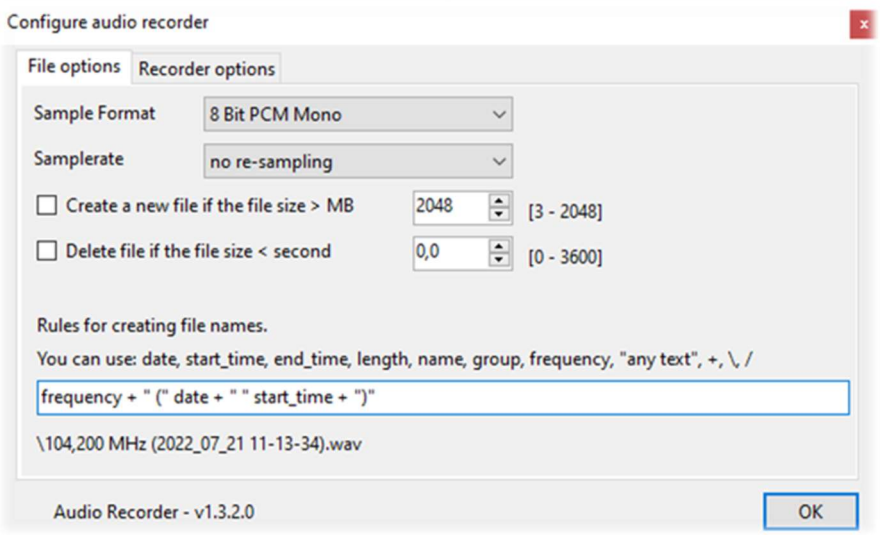


This other one, on the other hand, by author "Thewraith2008". I personally use a lot, it allows you to make recordings on the fly in the most congenial audio format with a wealth of options and parameterizations.

The latest update is v1.3.10.0.

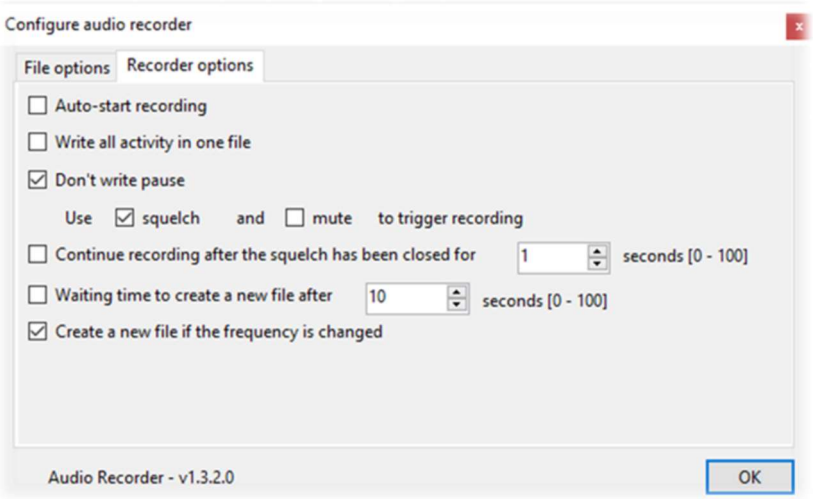
With the "Folder select" button you initially determine where the files will be saved, while with the "Configure" button you can customize a lot of things, some really important...

In "File options" for example you can decide for the quality of the WAV file and the rules for automatic file name creation (really useful!).

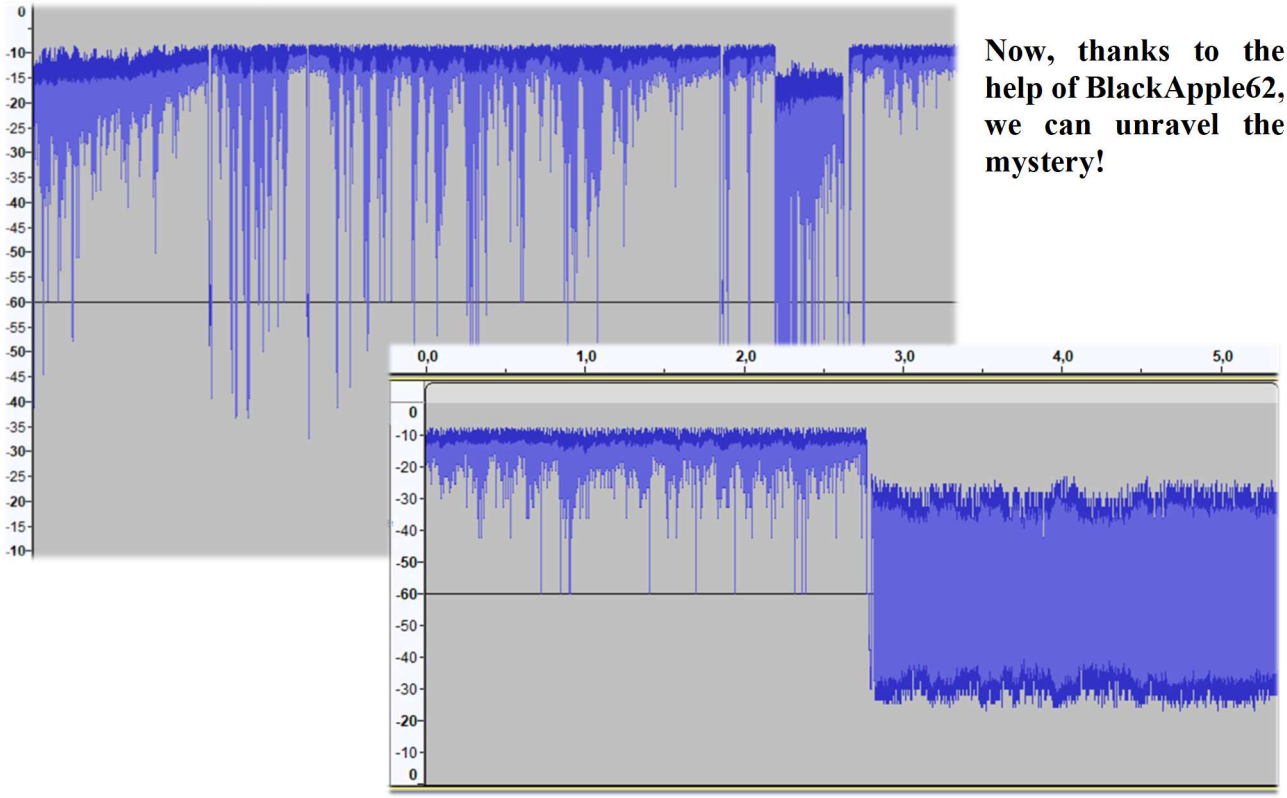


In "Recorder options" you can configure many other parameters.

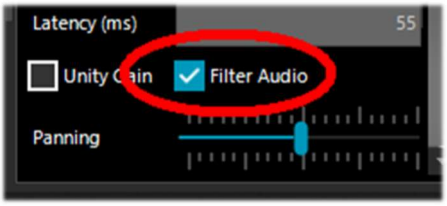
Personally, I consider very useful the options "Don't write pause / Use squelch" to make recordings only when is active the audio and "Create a new file if the frequency is changed"...



Some friends over time have reported to me an "oddity" in the recording of AM signals where the waveform did not appear to be absolutely centered on the horizontal axis, as can be seen from the screen (taken from Audacity), being all centered at the top...



Now, thanks to the help of BlackApple62, we can unravel the mystery!



Just flag, in the Audio section of SDR#, the "Filter Audio" to have perfectly symmetrical audio in our recordings.

In the above screen you can clearly see the difference with an audio sample recorded in the two modes.

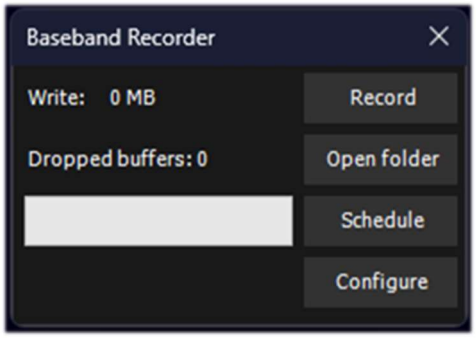
Same thing is to be done to have correct audio recordings and to avoid that distortion called "clipping" when using "Audio Recorder" plugins: the use of "Filter Audio" is highly recommended!

Baseband Recorders

To record IQ files instead, it is necessary to use a Baseband Recorder other than the Audio Recorder. Note that these are not traditional "WAV" audio files (despite the same extension) because they contain very large binary data from the stream exchanged between USB and SDR..

The "Sample Format" allows us to choose the quality level of the recording. However, we must keep in mind the following basic aspects:
* We recommend using 16-bit or 32-bit (example WAV-64RF format), but not 8-bit which would penalize our Airspy. The 8-bit PCM is good only with RTL-SDRs and to save HD storage space.
* ***Warning: when saving low-resolution IQ data, you must be sure that the signals are strong enough to overcome the quantization noise of the target resolution. For 8-bit, you need to have a noise floor close to -80 dBFS: so you need to increase the RF gain until you reach that level, then you can safely quantize the data.***
* In the case of criticality with unusable or partially good IQ records, it should be kept in mind that:
    1) if the buffer fills up it could be because of recordings made on slow/very slow or even very full disks. Different streams accumulate on the USB ports and together with the SDR itself can create criticality.
    2) dated USBs, especially the older 2.x, have limited bandwidth. If you receive on one port and record IQ on an external disk attached to another and perhaps via a HUB everything might suddenly crash... Same thing with a mix of USB 2.x and 3.x ports. You can always do a simple test by connecting two USB disks: save a somewhat large file on one and try using the other disk at the same time...
* To summarize: much depends by the computer, the version of USB/controller used, and the quality of the disks themselves (NTFS formatted, not FAT32!!).

Recordings can be started manually or by simple scheduling (called "Schedule"). Recording can be single or continuous, up to the maximum limit defined for each file format (see below), up to a set limit, or until there is no more available space on the harddisk.



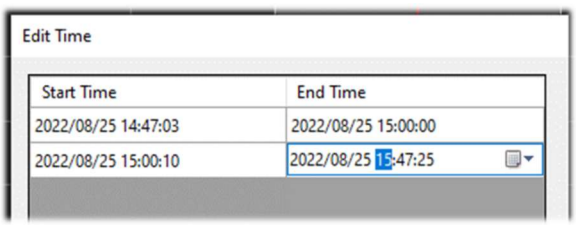
Baseband Simple Recorder: is the built-in SDR# from the v189x.

The "Configure" button allows you to choose the following types of files:

- WAV RF64 (size header 64 bit, for very large files)
- WAV FULL (size header 32 bit, up to max 4,095 GB)
- WAV STRICT (size header 32 bit, up to max 2,047 GB)

The formats are between 8 and 16 bits PCM IQ or 32 bits IEEE Float IQ.

The "Schedule" button allows the scheduling of one or more recording schedules. By clicking on the cell with the mouse, it is possible to change the start and end for each day and time...



Note: the now built-in baseband recorder locks the center frequency.

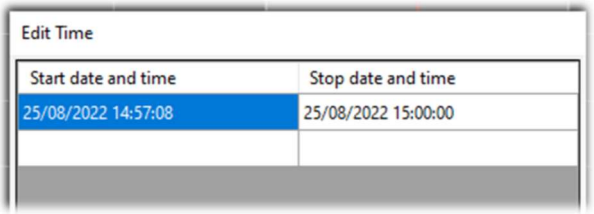
This other is by author "Thewraiht2008", with the latest update to v1.4.6.0 (visible in the bottom left corner in the "Configure").

The "Configure" button allows you to choose the following types of files:

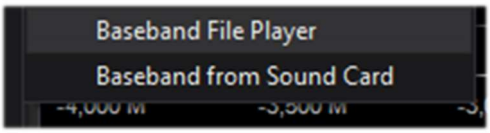
- WAV RF64 (for files up to 1 TB)
- WAV FULL (for files up to 4 GB)
- WAV SDSR# Compatible (up to a 2 GB)

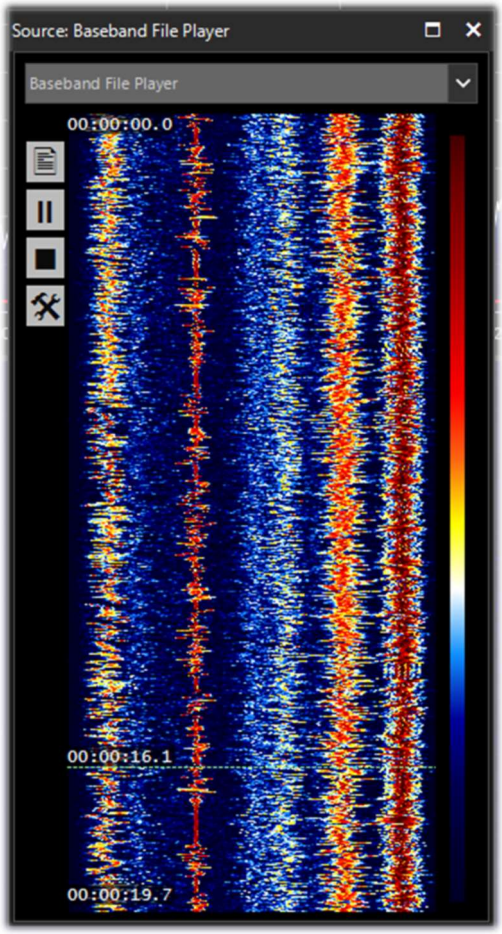
The formats are between 8 and 16 bits PCM IQ or 32 bits IEEE Float IQ.

The "Schedule" button allows management of one or more recording schedules. By clicking on each cell with the mouse, it is possible to change the start and end by day and time...



Next, to play any I/Q recording, one must use the "Source" panel by selecting toward the bottom "Baseband File Player"

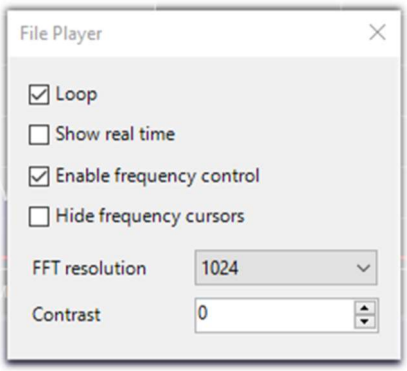




To search the HD and upload an IQ file, simply click on the icon  (see item 4 "Device configuration")

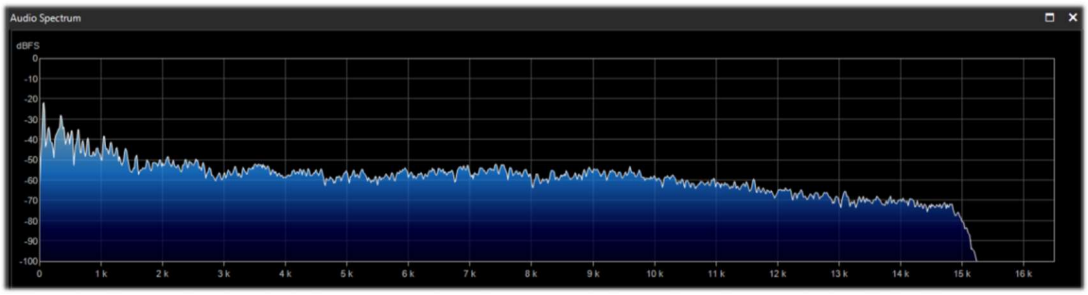
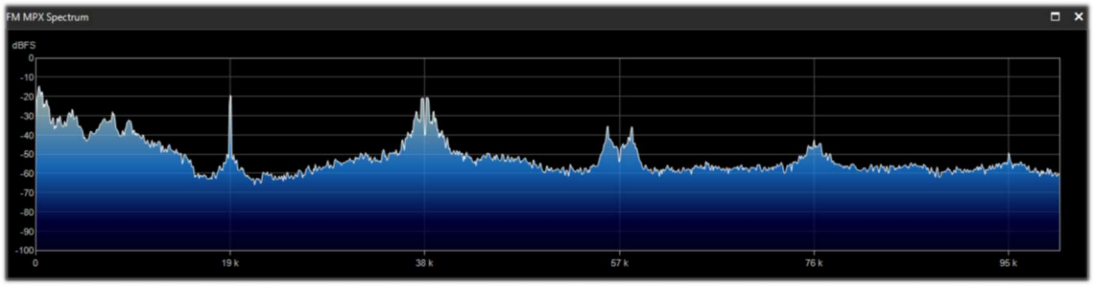
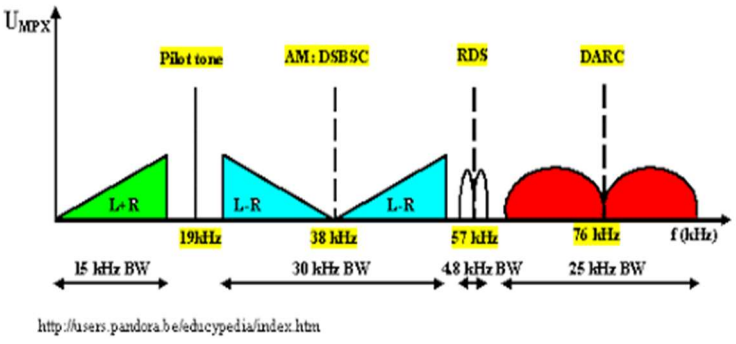
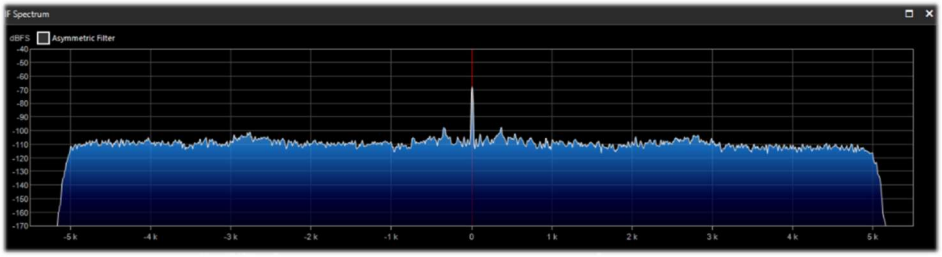
This side window will open, allowing you to move within the recorded file.

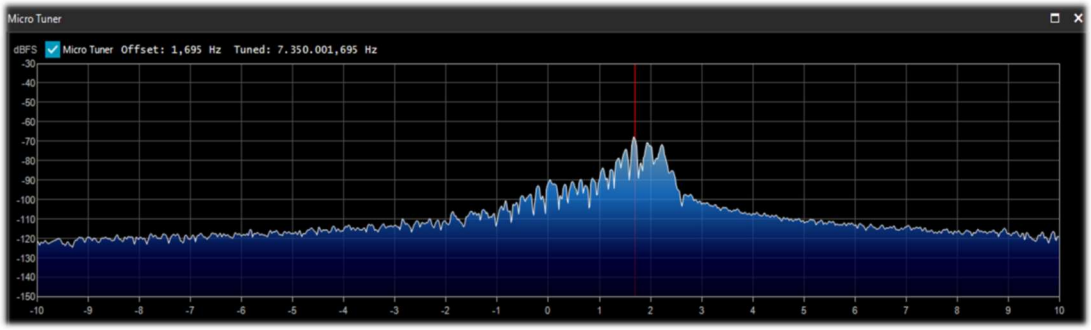
Click on the icon  to choose another file, and on the icon  to open a window with further customizations.



FFT Spectrum panels

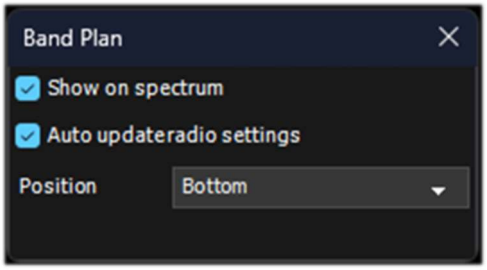
With these specific panels, some options can be displayed zoomed in. *Recently they have been optimized: the names and positions of individual options differ from previous releases.*

Key	Feature
Audio Spectrum	Allows you to see the audio spectrum in baseband. 
FM MPX Spectrum	Allows you to see the MPX spectrum, that is the baseband audio of an FM station.  *Can be activated only on WFM signals (in the 88-108 MHz band).*  http://users.pandora.be/educypedia/index.htm The spectrum contains, on the x-axis, from 0 kHz the monophonic audio section, then a pilot tone (at 19 kHz), the stereo section (centered at 38 kHz), then the RDS data subcarrier (57 kHz) or even other special services such as DARC visible in this screen...
IF Spectrum	Opens a zoomed window of the RF spectrum around the area of the tuned IF bandwidth.  *It allows you to view the signal structure with better resolution, and by enabling "Asymmetric filter" it allows you to select which side of the signal to act on.*

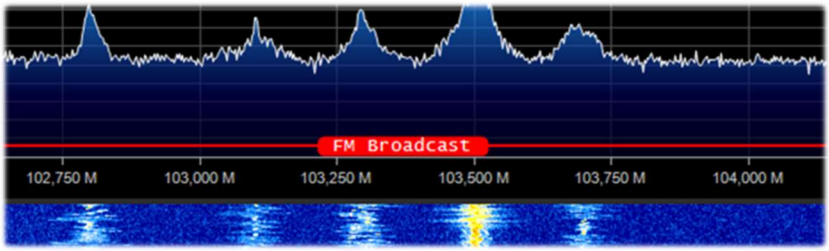
Micro Tuner	With v1886 (and later) this option was added which activates the panel called "Micro Tuner" where the main purpose of micro tuning is to "help" the CCC algorithm to better analyze the signal to be removed.  *This superb function, which is unmatched by any other SDR software of my knowledge, gives its maximum usefulness just together with the "AM Co-Channel Canceller" to achieve a very deep rejection of the selected station thanks to a brand new algorithm with a native implementation.* *Its combination with HF+ Discovery is the best MW-dx listening system, unbeatable for its price...*

Band Plan

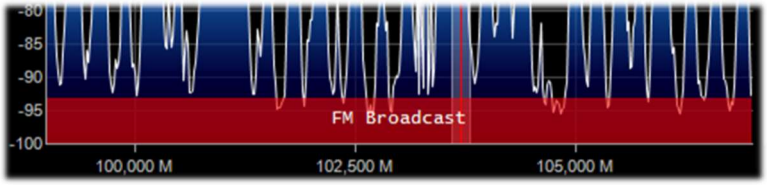
The Band Plan panel is very useful to view the many services that use radio frequencies across the spectrum in an organized way from different countries (in the following screen the "FM Broadcast").



Subsequently, the visualization changed, for the same information provided, it is now a very thin line and less intrusive to the UI, this is to make room for other plugins. XML files are now explicitly indented to facilitate offline editing and modification.



Until v1834 it appeared like this (with a very thick coloured band). In the following example it is displayed in "Bottom" position)



Key	Default	Feature
Show on spectrum		Enabling this option will display a rectangular color bar with the bandplan in the RF Spectrum window at the position indicated by the "Position" option.
Auto update radio settings		Enabling this option will automatically detect the emission mode/step and set in the VFO. *So if in the bandplan, in certain portions of the HF band is provided the USB mode and 0,5 kHz step, it will be applied immediately only typing the frequency!*
Position	Bottom	It allows you to choose between three different positions for the display of the Bandplan: Top, Bottom, Full (over the whole RF Spectrum window).

The support "BandPlan.xml" file, present in the program directory, must be modified with the information of your national knowledge by inserting the appropriate lines of text and respecting the format syntax. This must be the format of each "RangeEntry" unique for each frequency group:

<RangeEntry minFrequency="87500000" maxFrequency="108000000" color="90FF0000" mode="WFM" step="12500">FM Broadcast</RangeEntry>

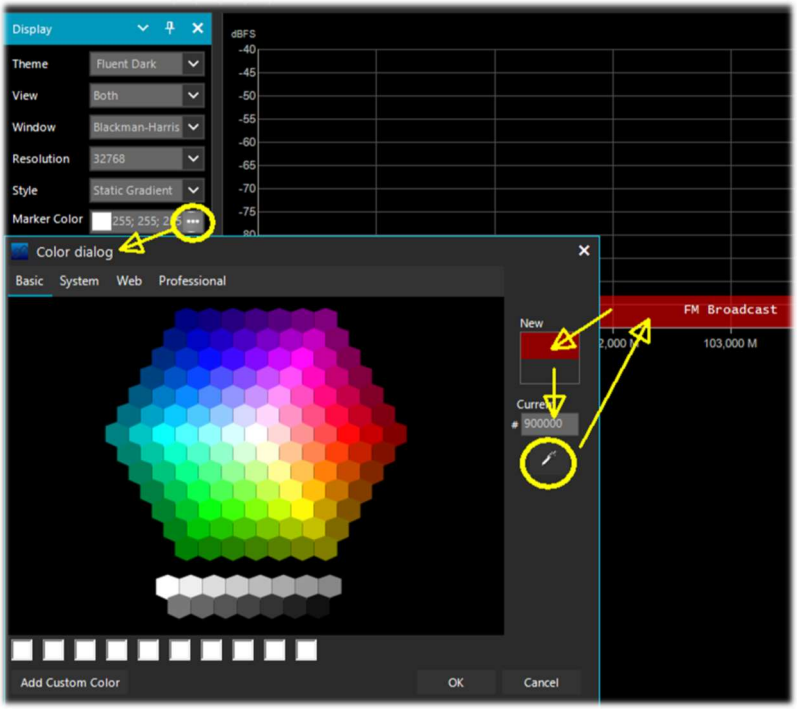
Each band can be divided into individual areas with different coloring, except for the overlapping of a subgroup (but not altogether).

Colors are defined as T-RGB, where T=Transparency (in values from 0 to 99 as a percentage, from almost completely transparent to full color) R=Red, G=Green, B=Blue in blocks of 2-digit hexadecimal values (indifferent to uppercase or lowercase letters).

To define the colors you can use the internal tool named "Color dialog" that you can reach from the panel “Display” → “Marker Color” → 


In the BASIC menu, through this icon you can select a color on the screen to have immediately, in the "Current" window, the hexadecimal countervalue.

Or by entering a value you can immediately see the result in the field. In the example below, the red band of FM broadcast appears as "900000". Or you can use the "Professional" menu to have all possible color palettes available.

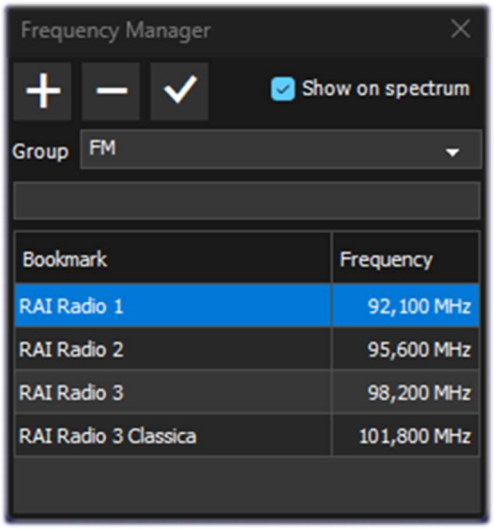


Or at these links among the many available on the net:
http://www.w3schools.com/colors/colors_names.asp
https://toolset.mrw.it/html/colori-del-web.html
http://www.colorihtml.it/
https://encycolorpedia.it/d0417e

The "mode" must be set between: WFM, NFM, AM, USB, LSB, CW. The "step" will automatically set the receiver VFO to the preset value for each band. The final field allows you to enter a text label that will appear as a name in the bandplan. *Be careful not to enter particular or special characters that could block the interpretation of the XML file, so it is recommended to use only alphanumeric characters.*
This plugin is very useful and allows you to divide the various bands of service assignment in automatic mode by simply clicking on the RF Spectrum*, but be careful because some bands with multiple assignment in emission modes make the correct mode pre-selection impractical (e.g. the articulated V-UHF bandplans of radioamateurs). In this case, deselect the "Auto update radio settings" option in the Band Plan panel.*
Any formatting errors in the file or the use of special characters will prevent the plugin from loading when the program starts!

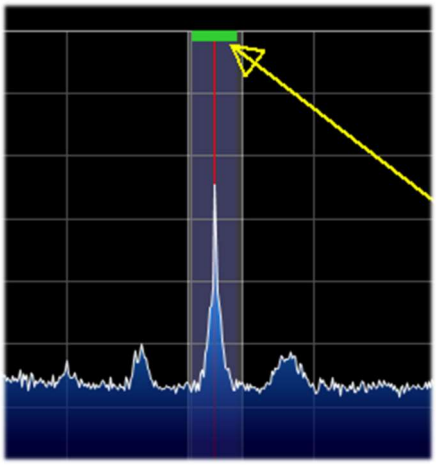
Frequency Manager

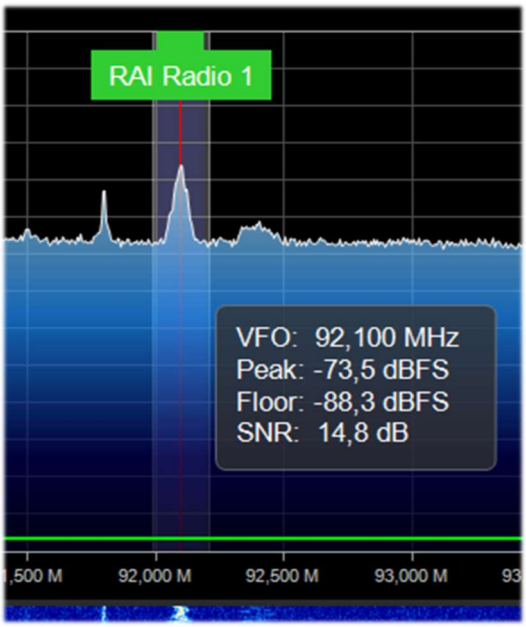
The Frequency Manager panel allows you to catalogue a large database of all the frequencies of interest. A new frequency can be added directly by clicking on the “New” button. A small data-entry opens, where all you have to do is add the name of the Group (if any), the name of the station and confirm all the other data already automatically acquired. Then a double click on a record will tune SDR# to that frequency, automatically setting the emission mode and its bandwidth.



.

If you check the "Show on spectrum" box, a colored horizontal indicator will appear at the top of the RF spectrum, and if you mouse over the BW column, the label shown in the bookmark will appear.

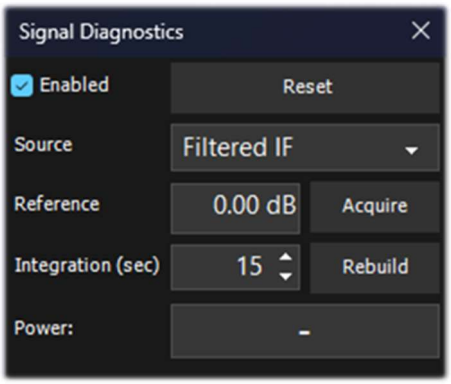




See also the optional "Frequency Manager (FreqMan) & Frequency Scanner" plugin....

Signal Diagnostics

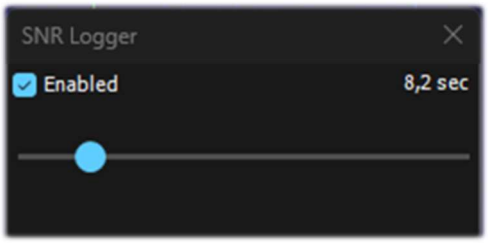
This diagnostic plugin is useful for determining the power levels (dB) of signals.



On YouTube, the radioamateur Leif Asbrink (SM5BSZ) has uploaded some very interesting and technical videos, where he shows how the Airspy HF+ can be used as an accurate power meter for RF signals. He points out that if the noise figure (NF) or minimum distinguishable signal (MDS) of a device is known, then it is possible to use this device as a power meter by calibrating it with a resistor (dummy load) at room temperature.

I suggest viewing at:
https://www.youtube.com/watch?v=ipwWayemCSQ&feature=youtu.be

SNR Logger

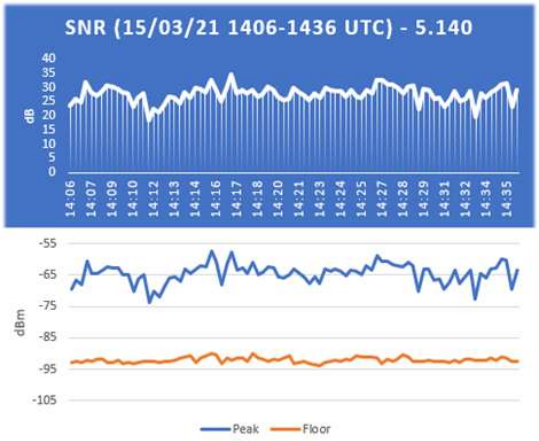


The SNR Logger has been implemented in the latest v18xx to include Peak and Floor in addition to SNR, making it truly unique in the SDR arena.

The signal strength is the height of the peak shown in the Waterfall while the noise level is simply the strength of the noise at frequencies where no signals are emitted. The absolute value of the difference between the two is called SNR and is expressed in dB.

	Timestamp	Frequency	SNR	Peak	Floor
2	2021-03-15 14:06:31.866	5140000	23.44	-69.57	-93.01
3	2021-03-15 14:06:52.479	5140000	26.02	-66.63	-92.65
4	2021-03-15 14:07:13.089	5140000	24.84	-67.92	-92.76

Once the flag has been enabled in the panel and a time interval has been selected with the cursor (up to 60 seconds), a text file with a name similar to this one will be created in a directory of your choice: "SDRSharp_20210315_140603Z_SNR.csv" inside are written the values in dB for SNR, and dBm for Peak and Floor detected by the active frequency of the VFO.



The small CSV file can be imported into MS Excel for further analysis and, using a suitable graphical representation, it will be possible to report the Timestamp data (date/time) on the x-axis and the values of the received signals on the y-axis. In the example the reception of R. Charleston at 5.140 kHz on 15 March 2021).

Plugins

In this section I will describe, in alphabetical order, some "Plugins" that compared to the default "Panels" are options developed specifically for SDR# that expand or extend the original functionality. This is in fact another peculiarity of the software, unique in its genre, that allows to API developers to make it for all specific needs...

A lot of plugins can be found on the net but recently SDR# has been updated to the latest technical knowledge regarding the internal DSP, audio and GUI: therefore with this in mind, developers are encouraged to review their plugins to have full compatibility.
In the my examples, however, I have still left indication of plugins widely used in previous versions of SDRsharp...

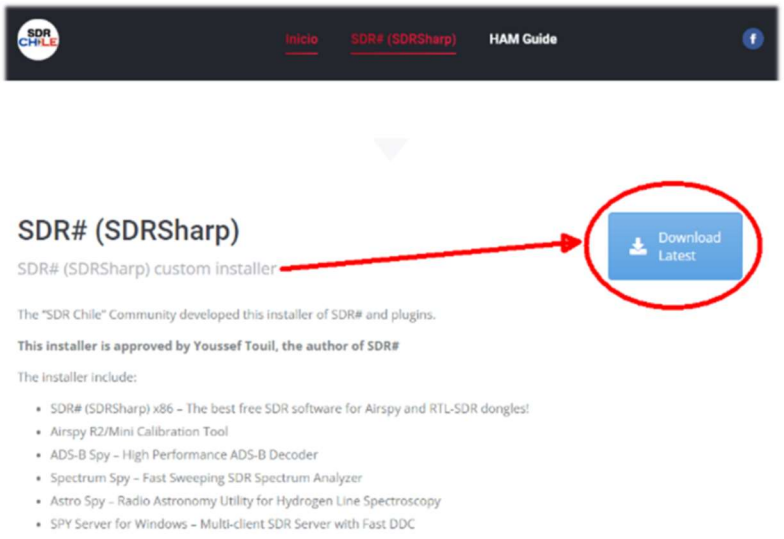
C:\RADIO\SDR#
airspy.dll
airspyhf.dll
api-ms-win-core-winrt-l1-1-0.dll
BandPlan.xml
D3DCompiler_47_cor3.dll
...
C:\RADIO\SDR#\Plugins
dmr_full.dll
SDRSharp.AudioRecorder.dll
SDRSharp.AudioRecorder.dll
SDRSharp.FreqMan.dll
...

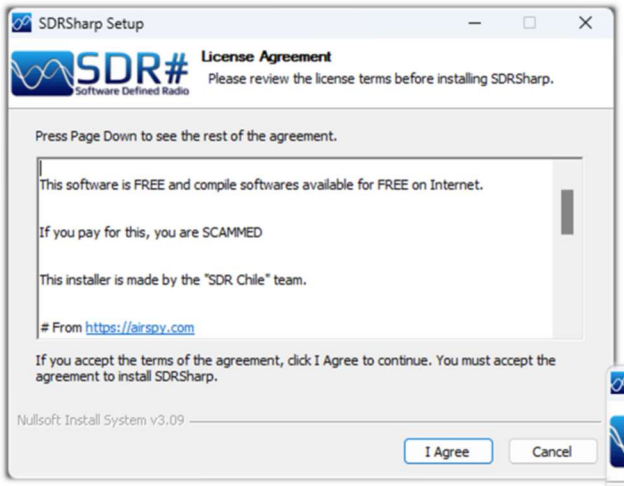
Since v1801 much has changed! Since then for their use it is sufficient to create a subdirectory "Plugins" and put the related DLLs in it. Loading at SDR# startup will be automatic, and the Plugins.xml file and MagicLine are no longer required.

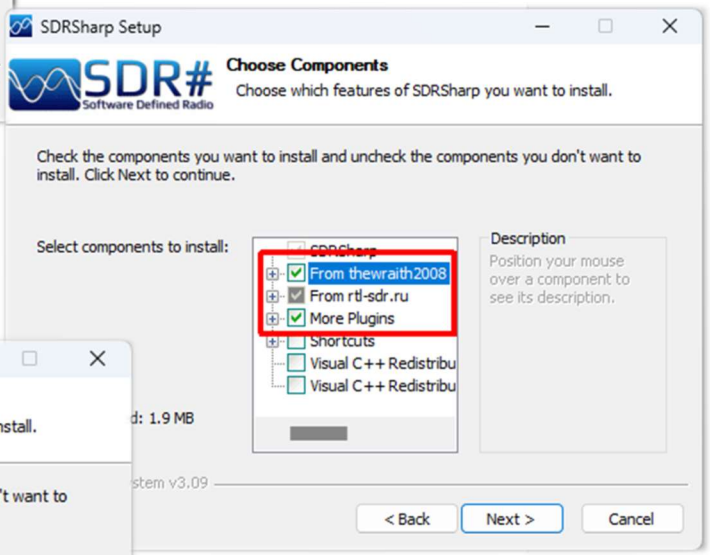
You can also decide to use another custom directory by editing the "core.pluginsDirectory" statement in the SDRSharp.config file. To disable the loading of a specific DLL (or directory) simply rename it so that it begins with the underscore character "_".
In case of some error in loading the plugin you will be able to find indication of it in the log file named "PluginError.log".

PLUGINS INSTALLATION

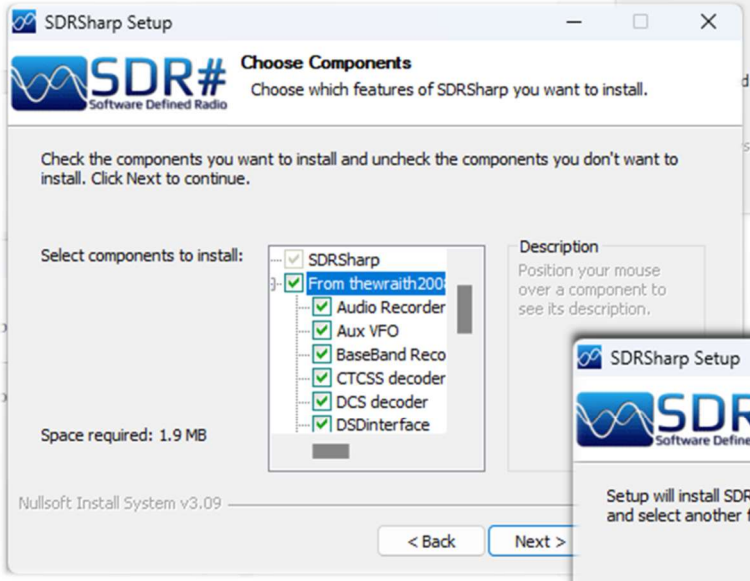
Plugins can be loaded manually by taking them from the developers' sites or via the flexible and updated **"Community Package Installer"** developed by Rodrigo Pérez. These are the necessary steps...
Connect to: ***https://sdrchile.cl/en/*** to download the file **SDRSharp-Installer.exe**



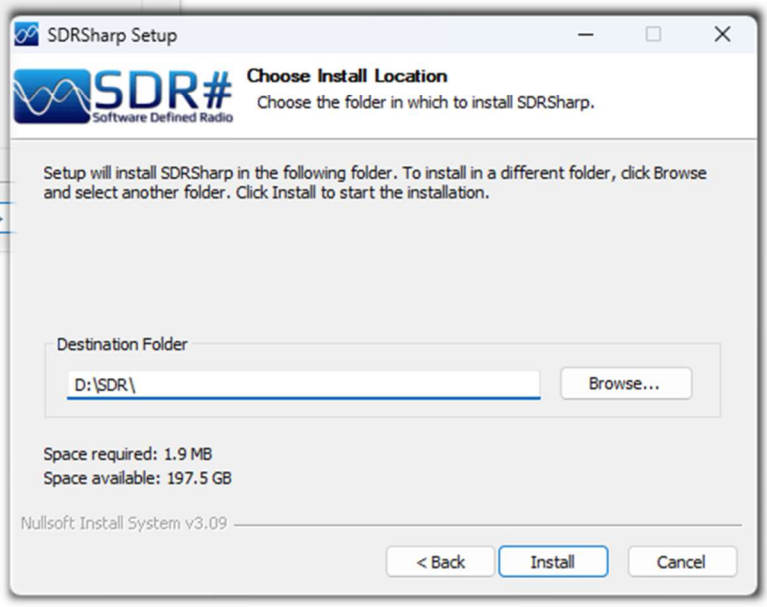




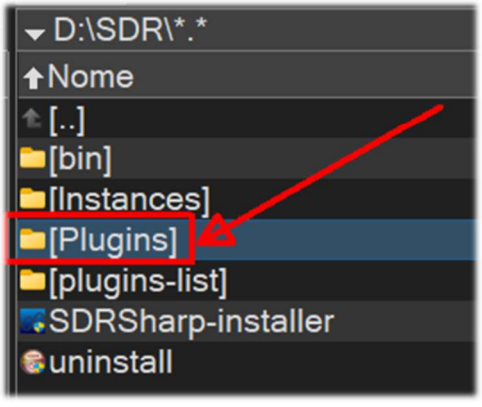
Select only the plugin of your interest



and finally choose the Install Location



To be absolutely avoided, however:
"C:\program files"
"C:\program files (x86)"

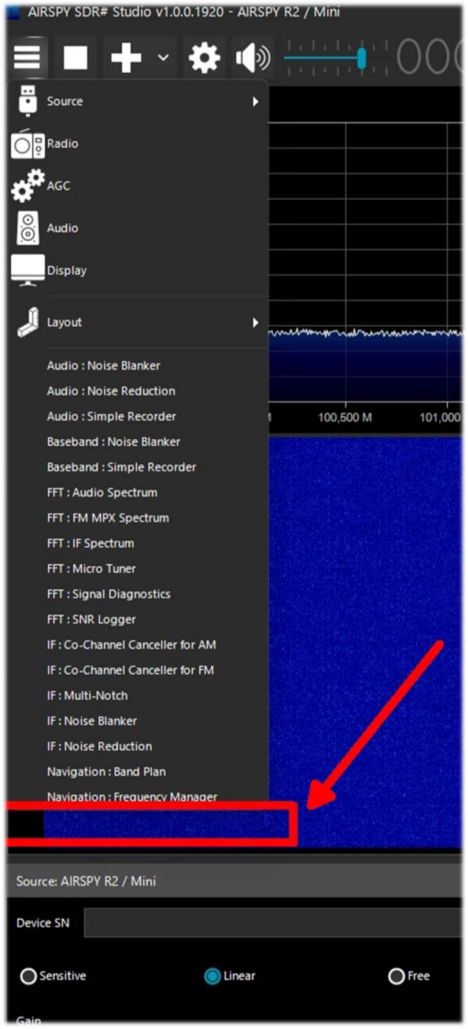


Eventually we will have such a thing with the directory "Plugins" containing the necessary.

In my case, I then copy the plugins manually to the respective folder of my SDRsharp.

WARNING: Since v1919/1920 some improvements in the audio chain have made it essential to patch some files directly from the "Community Package Installer" for the following plugins: Simple APCO, Simple DMR, Simple DPMR and Tetra.

So in case of audio problems you have to download and install the new DLLs fixed…



In the event that the usual icon for the plugins group is not present when SDRsharp is started (left image)

can mean two things:
1) there are no plugins in the appropriate directory.
2) or they are not found and need to check the path and what appears in the file "SDRSharp.config" at the following line:
<add key="core.pluginsDirectory" value="Plugins" />

WARNING! SOME OF THESE SYSTEMS IT COULD BE ILLEGAL IN YOUR COUNTRIES!

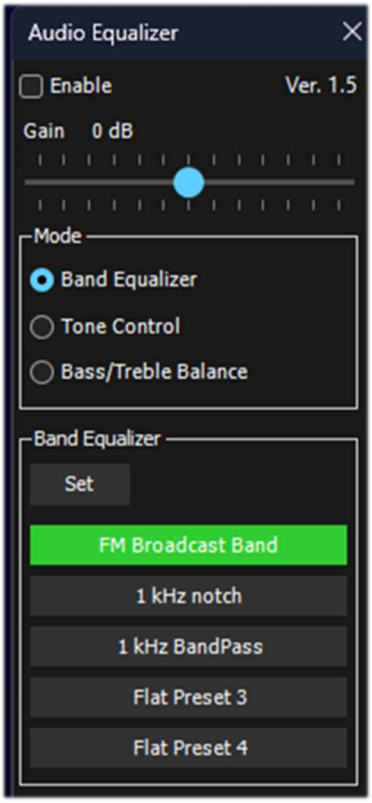
Check carefully and thoroughly the regulations in force in your country. Some of this radio system was specifically designed for use by government, emergency services, for public safety networks, etc etc. who all share spectrum allocated to a city, county, or other entity.

Audio Equalizer v1.5

Already author of the "Magic Eye" plugin (see below), Marco Melandri (BlackApple62) makes available freeware its ever-updated "Audio Equalizer":

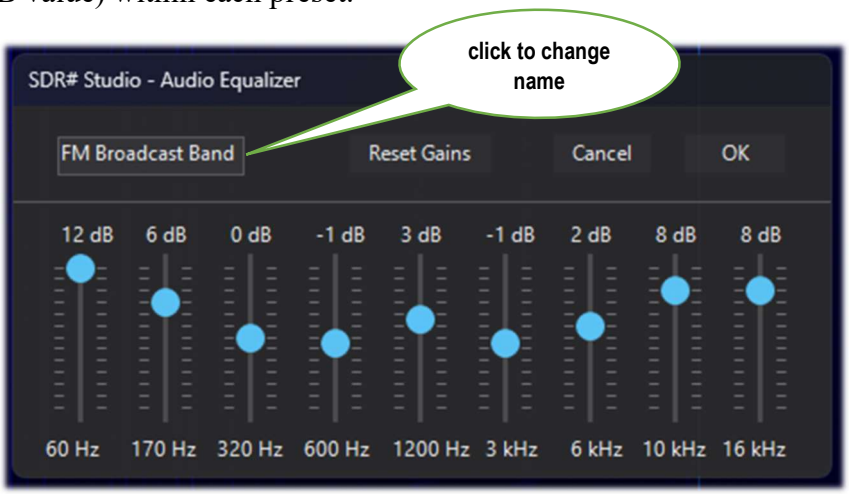
https://github.com/BlackApple62/SDRSharp-Audio-Equalizer-Plugin

The plugin provides a panoramic equalizer, tone and bass/treble balance control, now compatible with the latest SDR# Studio 32bit updated to the new .Net.

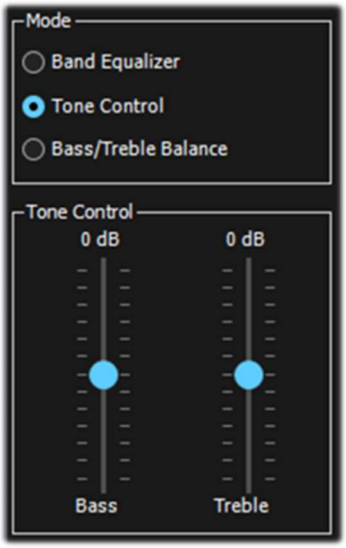


The "Enable" flag activates the plugin and with "Gain" slider sets the relative gain.
The "Set" button accesses the configuration of the five presets starting from the ability to assign a name (as shown below) at will and then set the nine bands (60 Hz to 16 kHz) in the range +/- 12 dB.
The panoramic equalizer set window allows to control SDR# Studio while is active.

The "Reset Gains" button resets the gain of the nine bands to flat (0 dB value) within each preset.

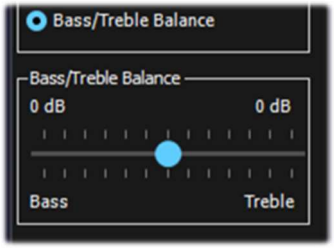


The equalizer configuration data are automatically saved in the "SDRSharp.config" file.



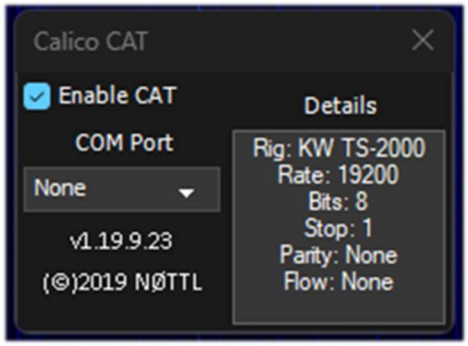
These are the two additional handy panels:

"Tone Control" and "Bass/Treble Balance"

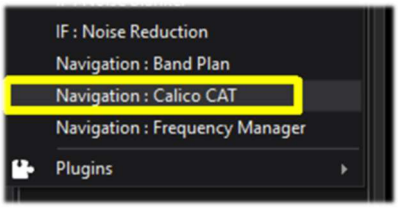


CalicoCat v1.19.9.23

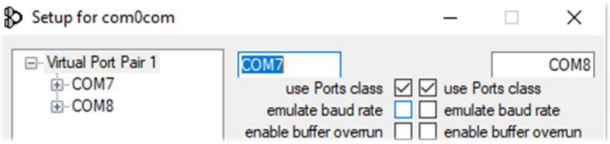
CalicoCat, created by Stephen Loomis (N0TTL) as of 2018, allows SDR# to interface with other amateur radio software through a virtual serial port via CAT protocol commands. The plugin supports a subset of the Kenwood TS-2000 command set and therefore any software used must be set up to communicate with this rtx at 19,200 baud rate, 8 bits, 1 stop bit, no parity, and flow.



In February 2024 the author donated the source code which then became part of the standard "Navigation" plugins from v1920 (april 2024).



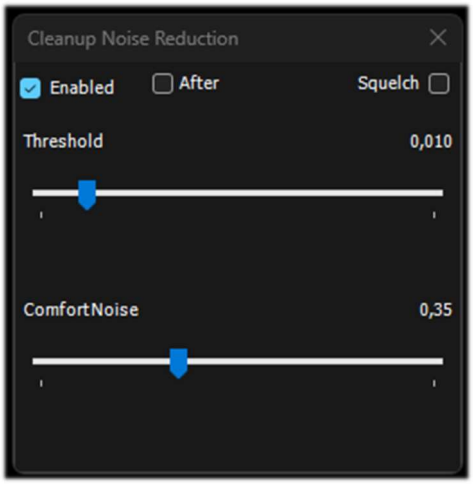
CalicoCat will use two virtual serial ports previously created by software such as "com0com". The actual numbers that will be assigned to the COM ports depend on the configuration of your system (in my case COM7 and COM8). I therefore selected COM7 in SDR# and COM8 in the other software with which I wanted to interface. Since CAT is a bidirectional protocol, changes made in SDR# will be immediately sent to the other software and vice versa...
In the chapter "Listening Recipes," we will see this at work with the Fldigi software.



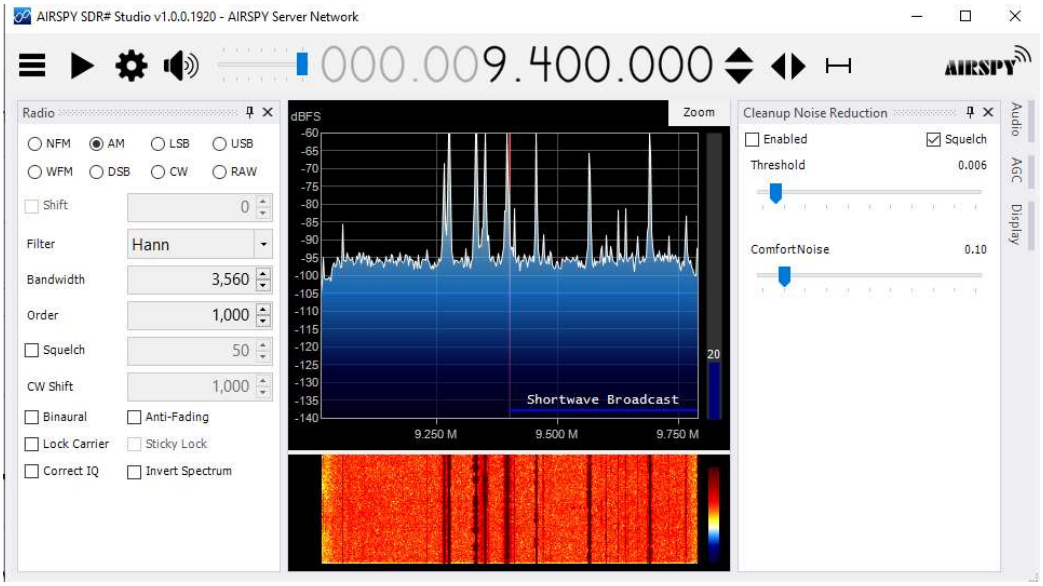
The previous plugin release, for compatibility with .NET7, is still present here:
https://Airspy.com/downloads/SDRSharp.Calico7.zip

Cleanup v1.0.6

The Cleanup plugin is a new freeware DSP algorithm for denoising on speech developed by Joshuah Rainstar. Designed to work mainly on shortwave (similar to NINR), but also on FM, VHF/UHF with narrow and wideband signals.



It is compiled for SDR# 32bit, with .NET 8 (SDR# Studio release >= 1920) and will not work on earlier versions.
It is designed to run at an input frequency of 48 kHz, but the basic algorithm can be adapted for use at frequencies of 6, 12, and 24k.
By hovering the mouse over the "Threshold" slider, the "minus: more signale, plus: less noise" popup appears, while "adjust for preferred level" appears on the "ComfortNoise" slider.
The use of a notch filter (for carrier suppression) and AGC are recommended before taking advantage of this algorithm.



The developer suggests these steps:

- Unzip the contents of the zipper file into SDR#'s "Plugins" folder.
- Select the Hann filter and use a bandwidth of about 3,2 or slightly more.
- Tune Cleanup: for AM and FM and for very faint signals, turn squelch off. Use the threshold to tune the squelch. Use ComfortNoise settings to tune it to a point where noise is minimal. The IF denoiser disrupts the noise estimation process.
- Turn the plugin off and tune any denoisers. Tune them to minimize any attenuation. The option "After" option should normally be enabled, but can be experimented with to further refine the results.

https://github.com/falseywinchnet/SDRSharp-Cleanup-Plugin

CSVUserlistBrowser v4.25

A software that I have been using since its inception is the very powerful "CSVUserlistBrowser" (or CSVUB) by radioamateur Henry DF8RY.

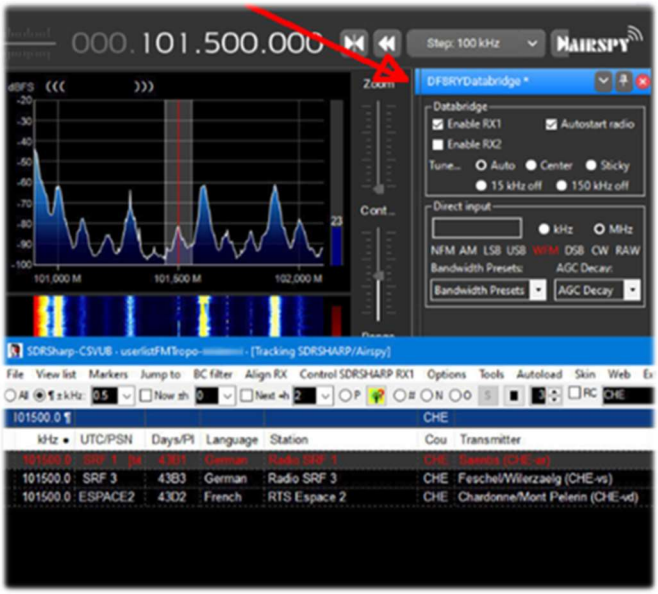
CSVUB is a Windows application that interfaces to SDR#, to manage numerous databases (or lists) of radio frequencies of long, medium, short and WFM broadcasting stations. It displays the lists in the following formats: AOKI, EIBI, HFCC, FMSCAN, numeric stations, "ITU monitoring", ClassAxe (for NDB), DX cluster, etc. etc. as well as Personal Userlists.

These are the steps for its installation:

- Download the file **https://www.df8ry.de/htmlen/csvub/CSVUserlistBrowser.zip**
- Extract the files to a directory on the HD with full write privileges
- When CSVUserlistBrowser.exe starts, you are prompted for the name of the receiver or receivers you wish to check. Select "SDRSHARP." (This step is only needed once at the beginning)
- From the zip file copy only the correct plugin for your release of SDR#!!! There are in fact five different versions of plugins in the zipper that should not be confused... For SDR# releases prior to rev1801: Copy the file SDRSharp.DF8RYDatabridge.dll into the SDR# directory. Open the file "Plugins.xml" in your SDR# directory with a word processor and add the following line in the <sharpPlugins> section:

  <add key = "DF8RYDatabridge" value = "SDRSharp.DF8RYDatabridge.DF8RYDatabridgePlugin, SDRSharp.DF8RYDatabridge" />
- For SDR# versions after rev1801: Copy the file SDRSharp.DF8RYDatabridge.dll into the Plugins folder of SDR#. No additional line is needed in Plugins.xml!
- Start SDRSHARP-CSVUserlistBrowser.exe (automatically created in the previous step in the CSVUserlistBrowser folder)
- If you have never downloaded databases/schedules with CSVUserlistBrowser, follow the instructions on the site on the "First Steps" page under Overview / First Steps.
- In the SDR# software, open the DF8RYDatabridge plugin and verify that "Enable RX" is selected.

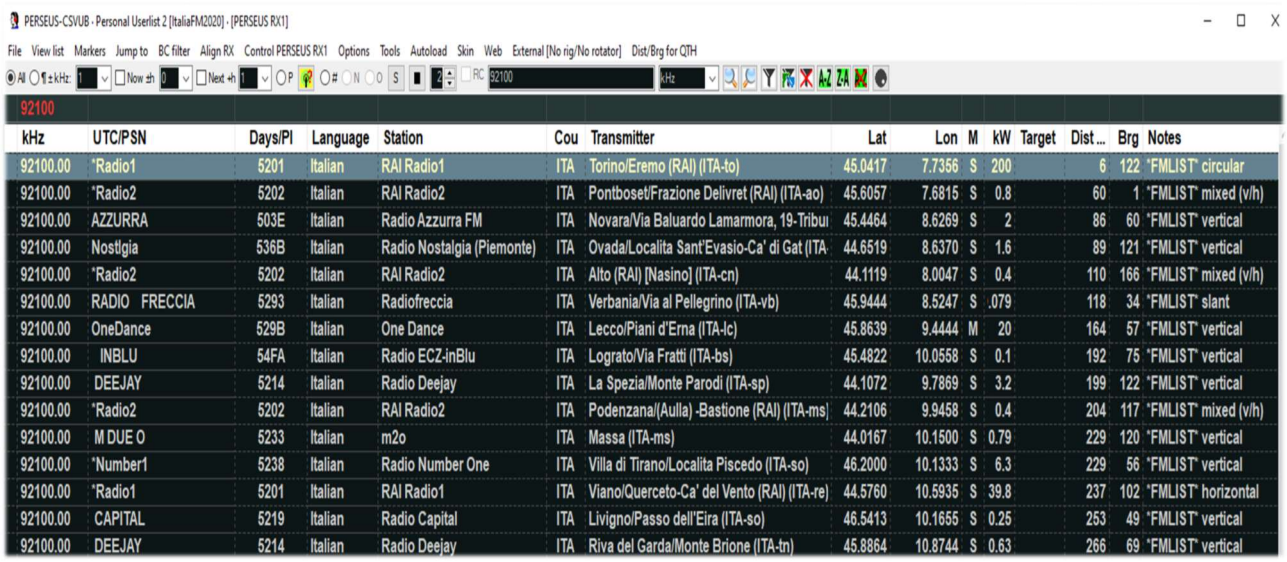
  You can start two instances of SDR# and control them with CSVUserlistBrowser (see the "Control SDRSHARP RX" menu).



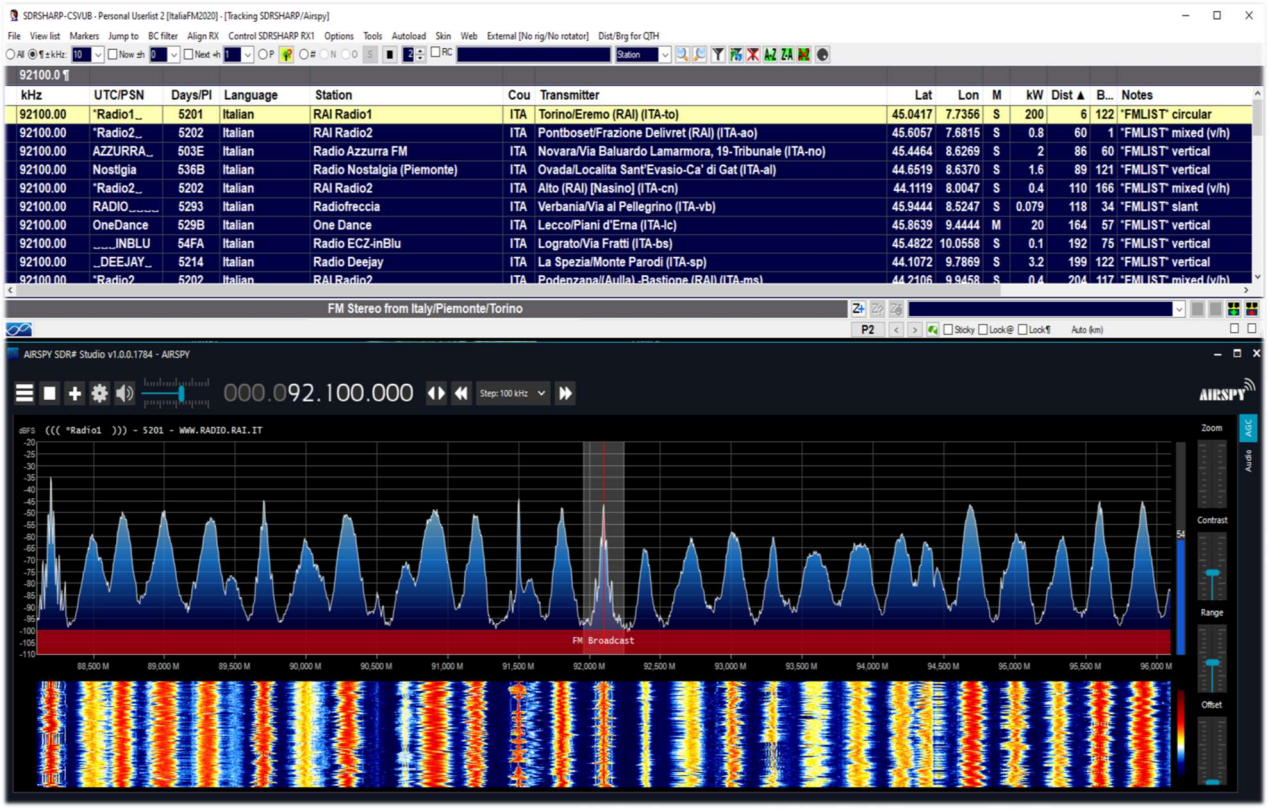
For its countless features and functions I invite you to consult here:
https://www.df8ry.de/htmlen/csvub/%F0%9F%91%93features.htm

CSVUB tunes the receiver with a single mouse click in the proper emission mode, showing the station name, time, language, transmitter position, distance and bearing, as well as other information automatically updated by the respective servers! It also contains Hamlib and Omnirig control for external receivers, also analog receivers that can be connected via RS-232. The plugin allows you to

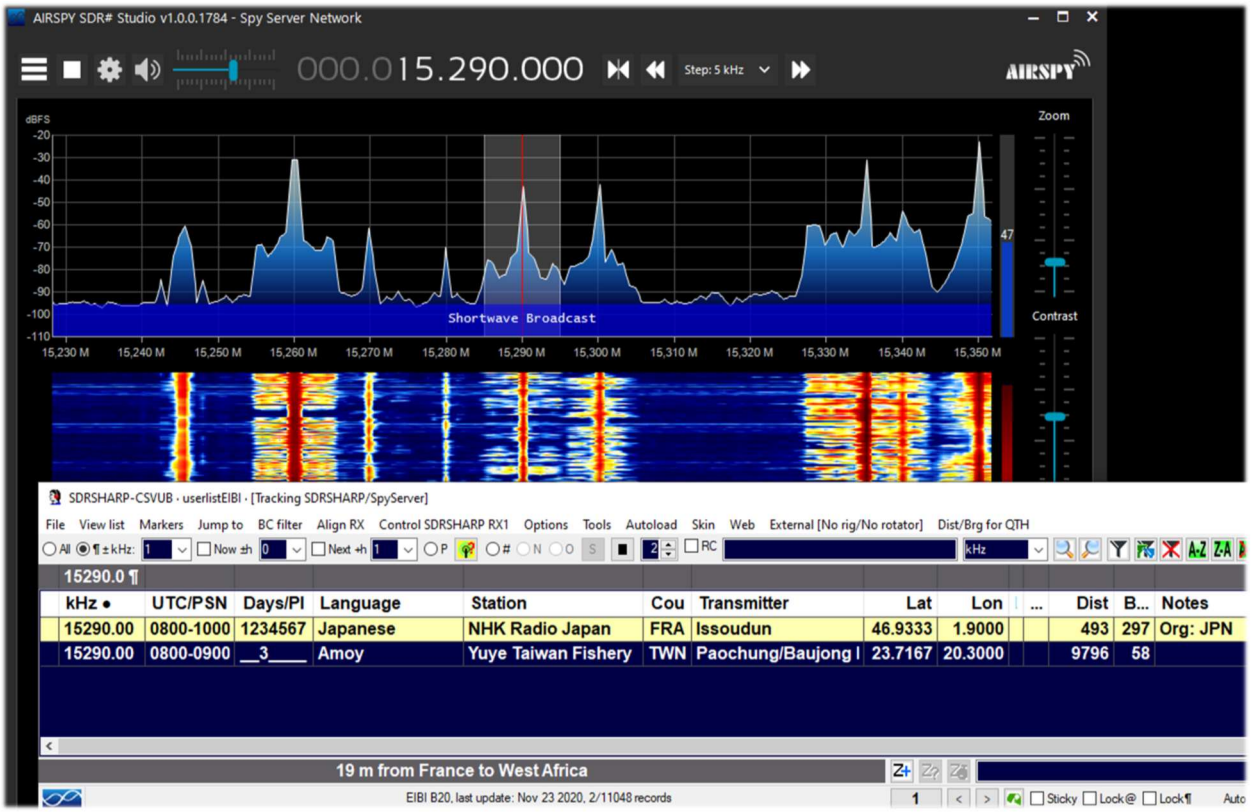
interface SDR# in a fast and non-invasive way, unlike other much slower and uncomfortable to use. The CSVUB window is external, dimensionable and positionable at will.



The CSVUB window is external, sizable and can be placed wherever you want. I prefer to keep it above SDR# to see all the frequencies and information immediately.

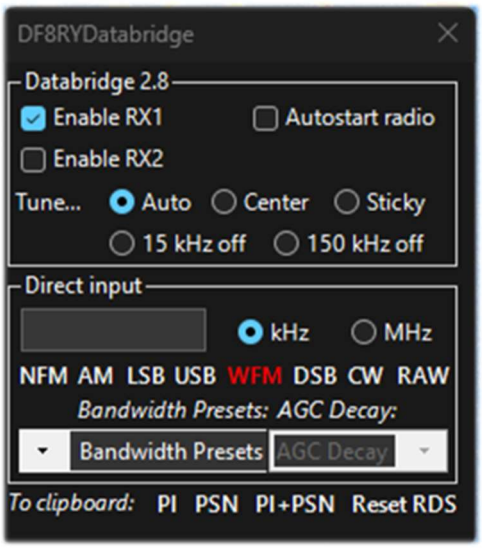


In the screen above, SDR# is tuned in full screen at 92,100 kHz WFM, the plugin sends the information to CSVUB, which displays it in tabular form, showing in the first line of different color the identified broadcaster. Everything is configurable in font and size, as well as having a customizable "skin" for the color scheme (in the example the skin used is the "SDRsharp"!). It can also work in reverse, you click on a frequency in the CSVUB table and the receiver will immediately tune in the correct emission mode and specific bandwidth preset.



In this screenshot, a Spy Server Network is used to verify a broadcaster on 19-meter. By enabling the "Track mode" option, the first line appears in CSVUB with its transmission highlighted. For very busy slots, you could also use the "Now" option which immediately filters the transmission at the current time.

Let's see in detail the possible plugin configurations through the “DF8RYDatabridge” v2.8.



Key	
Enable RX1 / RX2	Enable or disable SDR#-CSVUB control. There are two instances of connection to SDR#, for example one with an Airspy and the other with an RTL-SDR dongle.
Autostart radio	The plugin automatically starts the radio it finds connected. In case of problems when the radio is not connected, it is preferable to disable the option and start it manually.

	The receiver only starts automatically when RX1 is enabled. The autostart is blocked for a second instance of SDR# with RX2, otherwise it would start the same radio twice and cause confusion.
Tune… Auto	The frequency position, tunes in the RF spectrum, is controlled by SDR#.
Tune… Center	The tuned frequency always appears in the center of the SDR# RF spectrum *(see Tuning types).*
Tune… Sticky	Use SDR# Sticky tuning mode *(see Tuning types).*
Tune… 15 kHz off	The frequency is tuned to 15 kHz from the center. This avoids collisions with the typical peak I/Q that some RTL-SDR/sound cards produce in the center of the RF spectrum RF.
Tune… 150 kHz off	As in the previous point, but for reception in WFM. The frontend must have sufficient RF bandwidth (at least 300 kHz).
Direct input kHz or MHz	Here you can directly type a frequency in kHz or MHz and press Enter for tuning: *really very convenient and fast!* Or, when with the mouse, you have the "focus" on this field, the Pag Up/Down keys or Up/Down arrows tune the VFO gradually with the Step Size selected in SDR#.
NFM … RAW	Eight buttons for immediate setting of the various modes.
Bandwidth Presets and AGC Decay	These are some default snapshot settings for SDR# that may sometimes be useful. Not related to CSVUB.
To clipboard: PI, PSN, PI+PSN	*In WFM mode, the following buttons at the bottom of the plugin are activated and usable.* When a WFM station is received with the RDS decoded by SDR# it is possible to copy the code PI, PSN or both to the clipboard, to be used to compose its own Personal Userlist.
Reset RDS	The button activates a new RDS decoding in SDR# (it is basically a reset of the RDS).

This is the reference link (the SDR# plugin is inside the zipper file):
https://www.df8ry.de/htmlen/csvub/%F0%9F%93%BBsdrsharp.htm

It has so many options and features that it is impossible to treat them all here even minimally. I recommend that you download and consult the relevant manual.

WARNING:
In the last quarter of 2023, the creators of the "fmscan.org" website decided to provide RDS PI codes only to regularly registered users anymore. Therefore, those who do not have an account, the Downloader in CSVUB will no longer download PI codes.
In order to continue to receive the lists with PI codes (for strictly personal use and not transferable to third parties and/or unauthorized persons) this is the procedure to follow.
All users are also invited to cooperate by sending their listening logs to keep the common database up to date...

- Register with Email/Password and/or go to the site: ***www.fmlist.org***
- Go to the site: ***www.fmscan.org***
- Set up your QTH if not already done.

- Go down and click under "Tools (userlists etc.)"
- Click on "Userlists for Perseus/ELAD/Winradio/Stationlist/SDR Console".
- Click on the desired list (FM+Tropo, FM+ Meteor Scatter, Sporadic E).
- Choose "CSV format" and to the right "CSV Separator: Tab."
- Left-click on "DOWNLOAD userlist1.csv" and wait.
- Save the file by renaming it, for example, as "userlist-FMtropo.txt."
- Now you can load it into CSVUB with "Open CSV userlist (1...30)."
- To prevent this list from being inadvertently overwritten by a new version without PI codes, the ASD (AutoStart Download) flag next to "Download FMLIST" in the Downloader must be removed.

This procedure must be performed manually each time a new list is available.

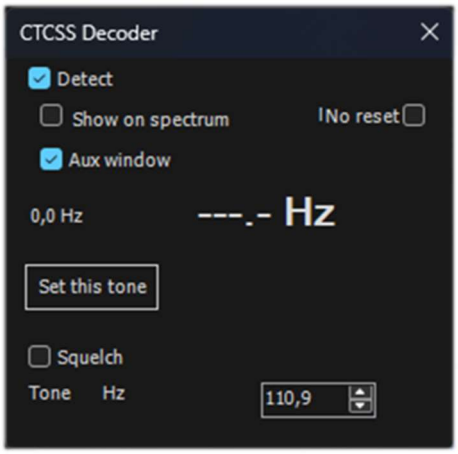
CTCSS & DCS v1.3.5.0

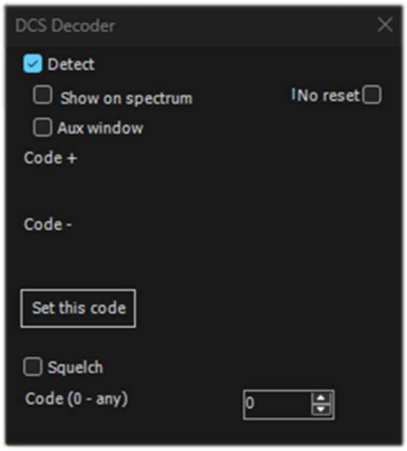
These two plugins, for analogue CTCSS and digital DCS detection *(only for NFM mode),* have recently been updated by the tireless work of "Thewraith2008" (already the author of other plugins from the initial TSSDR Vasili work).

Important note: you cannot use CTCSS and DCS at the same time.

Versions 1.3.2.0 of both have recently introduced a new feature: the value of the detected CTCSS or DCS will be sent to the "Frequency Scanner" plugin for displaying/recording (obviously the latter must be updated).

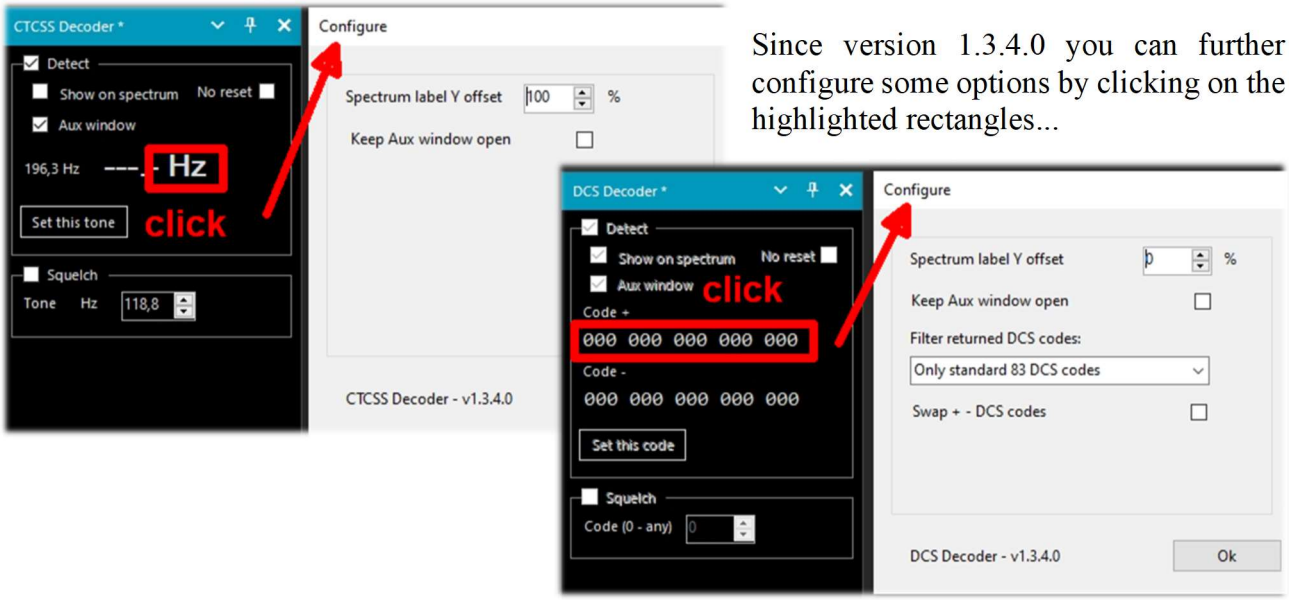
Before even with versions 1818/1822, the size of the audio buffer having been greatly reduced, the previous plugins no longer worked, as they required the presence of some "zero crossings" in the audio buffer (i.e. points of zero crossing, just before the change of sign, in the alternation of the signal between positive and negative values) to detect the tones, and therefore have been updated again.
They can be downloaded, together with other plugins (Auto Start, FreqMan, Frequency Scanner, ScopeView and Short-wave info), from the site forum: ***https://www.radioreference.com***





Option	
Detect	"Detect" enables/disables plugins. The tones/codes detected and decoded by the audio buffer are made visible in the following ways: in the body of the plugin, in the spectrum window and/or in an auxiliary window. ***Read the NOTES below...***
Show on spectum	 **CTCSS:** With this parameter enabled, you can display the detected tone directly on the RF Spectrum, on the right side of the VFO mark, (or on the left side if the margin is at the edge of the screen). *See below for new options introduced by v1.3.4.0 especially for positioning on the Y-axis of the spectrum.*  **DCS:** The detected codes (positive or negative) will be displayed at the top of the RF Spectrum and next to the VFO marker.

No reset (on change of frequency)	Option recently added with release 1.3.0.0. Keeps the last detected CTCSS/DCS visible on the panel and on the external window but will delete the one on the RF Spectrum. *Can be useful during a scan to keep the last detected tone/code visible.*
Aux windows	 With this parameter enabled, it is also possible to display the measured data in auxiliary windows that can be positioned anywhere on the screen and always in the foreground with respect to all other open windows.
Squelch / Set this tone	Enables/disables Squelch to operate with the detected tone/code.



Since version 1.3.4.0 you can further configure some options by clicking on the highlighted rectangles...

NOTE (1) - CTCSS: *The developer informs us that the plugin may have some difficulty in detecting subtones at a lower frequency in Hz and therefore recommends (SDR# v1810 and lower) to increase the Latency value to 60 (mS) in the "Audio" panel.*

NOTE (2) - DCS*: An option has been introduced in the "Configure" to use only those DCSs that exist in the table, thus reducing the list of DCSs. The three options are:*
0 = Default - No need to vary anything.
1 = To use only the 83 standard DCS codes (those ETSI TS 103 236 v1.1.1-Table 2)
2 = As point "1" but in addition the 21 extended DCS codes.
To use the option choose the item from the drop-down menu or manually add the following line in the file "SDRSharp.exe.config" with the preference value, at the end of the block of the other entries starting with "DCS.xxxxx":

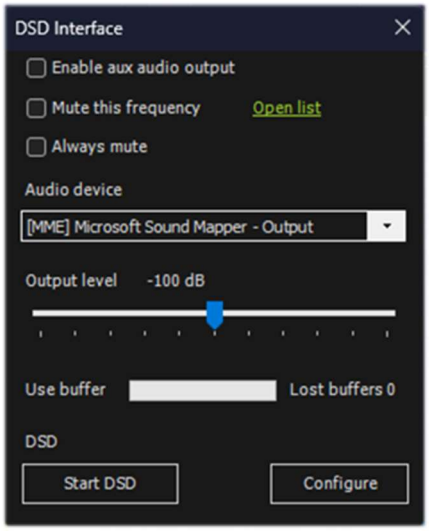
<add key="DCS.OnlyUseDcsCodesInTable" value="1" />

NOTE (3) - DCS*: Introduced an additional option to eventually switch the display of DCS codes between 'Normal' and 'Inverted'.*
To use this option, the following line must be added to the "'SDRSharp.exe.config" file:

<add key="DCS.SwapNormalInvertedDcsCodes" value="True" />

DSD Interface v1.0.9.0

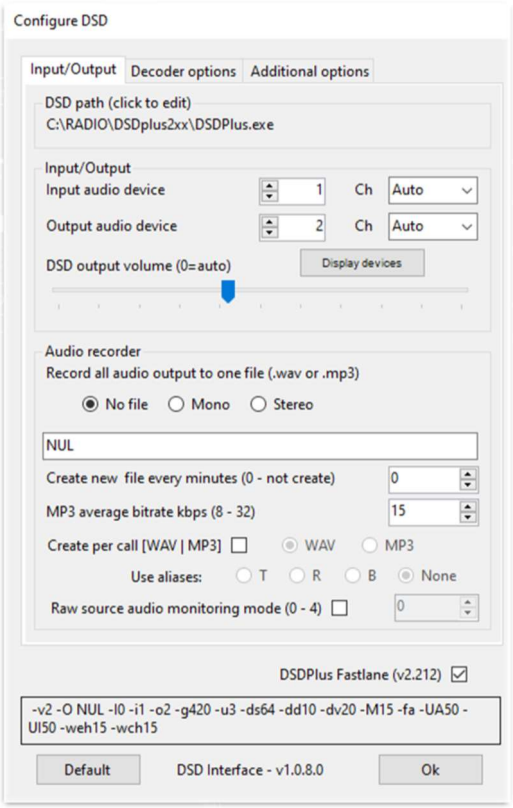
This plugin, updated as of July 2022 by author "Thewraith2008" (who has already published several others covered in this chapter) allows you to use SDR# as a radio source by providing a convenient graphical interface to DSD+ or Digital Speech Decoder software that allows you to decode audio of digital signals in the DRM standard, D-STAR...



The plugin supports:

DSD+ v1.101 public version

DSD+ Fastlane (with limitations specifically v2.212). The plugin version has been updated to support DSD+ Fastlane v2.212 (from v2.183 to v2.212). Future versions may also not work given the many changes made to DSD+ since then (v2.390 at the time of the author's writing).



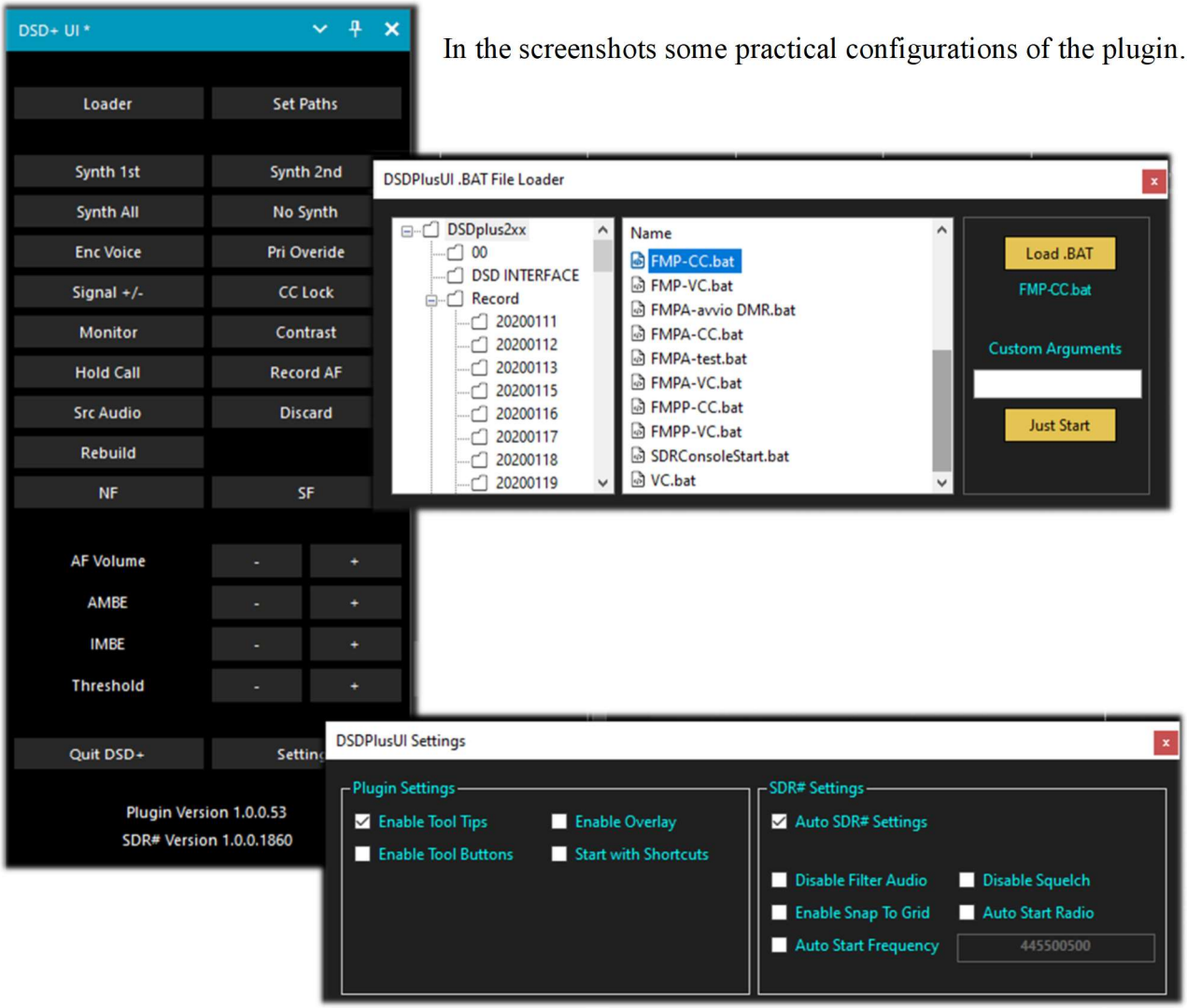
NOTE (1) from the author. This plugin will no longer be suitable for and will no longer support future versions of DSD+ Fastlane, as many features of DSD+ Fastlane will only work when used with its tuner front-end (FMP24, FMPA, FMPP). **NOTE (2)** Not all functions of DSD+ Fastlane are available with this plugin.

DSD+ UI v1.0.0.54

DSDPlusUI is a handy free GUI created for DSD+ especially to facilitate the use of FastLane releases (paid versions of DSD+) on x86/64 desktop and tablet platforms.

In fact, there is also a "MainLine" version on the developer's site, but we will deal here with the plugin for SDR# that pernects to configure and launch DSD+ directly from SDR#, via the usual DLL to be copied into the Plugins directory

DMR Slot1 and Slot2 emissions are processed and listened to simultaneously.

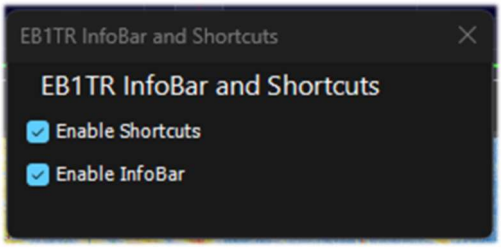


In the screenshots some practical configurations of the plugin.

https://dsdplusui.com/download.php?download_file=DSDPlusUI_SDRSharp_Plugin_v1.0.0.54.zip

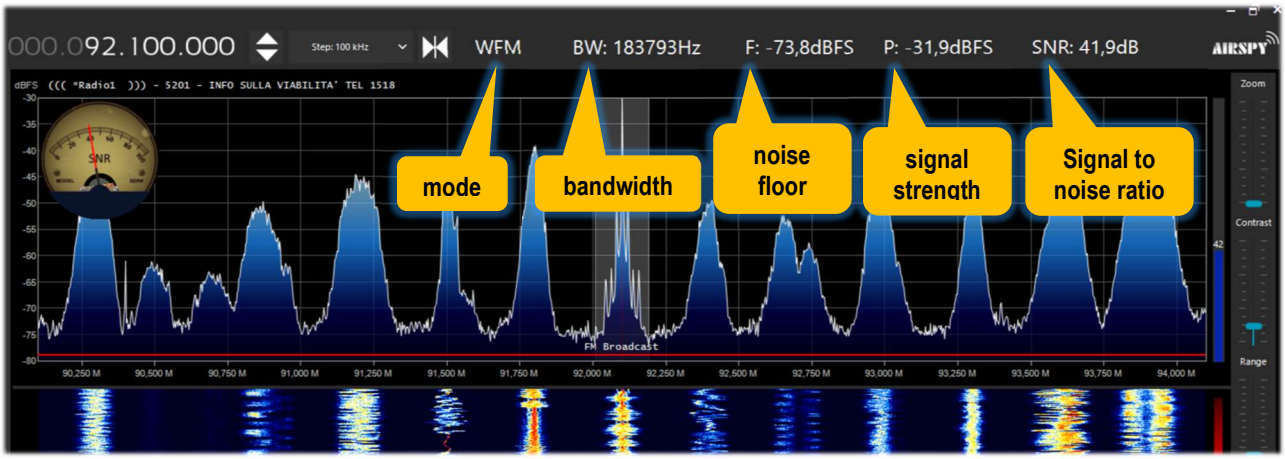
EB1TR Infobar and Shortcuts v1.1.1.0

Simple and effective Fabian's EB1TR plugin!
https://www.eb1tr.com/eb1tr-infobar-and-shortcuts/



It is divided into two groups of functions that can be turned on/off from the main panel.

Infobar
This is an information bar located on the right side of the VFO with the following information:



Shortcuts
This is a series of keyboard shortcuts that allow you to change the step, bandwidth, emission modes, RF Spectrum/waterfall zoom, etc., quickly and easily. The list is as follows:

Keys	Receiver control	Display
CTRL+left arrow	Decrease step	
CTRL+right arrow	Increase step	
CTRL+arrow up	Increase BW linearly (1 Hz)	
CTRL+arrow down	Decrease the BW linearly (1 Hz)	
CTRL+(+)	Increase BW exponentially (+10%)	
CTRL+(-)	Decrease BW exponentially (-10%)	
A	Mode AM	
D	Mode DSB	
F	Mode NFM	
W	Mode WFM	
C	Mode CW	
L	Mode LSB	
U	Mode USB	
R	Mode RAW	
M	Cycles between emission modes	
S	Cycles between the different steps	
Q	Enable/disable Squelch	
ALT+arrow up		Zoom in
ALT+arrow down		Zooms out
ALT (o CTRL)+Enter		Center the signal

Shortcuts EB1TR - Print, cut out and keep

FMS-Frequency Manager Suite v2.3.4

Writing “plugin” may be very reductive, in fact Jeff Knapp's is a freeware "suite" consisting of several modules. The reference link is: ***http://www.freqmgrsuite.com/***

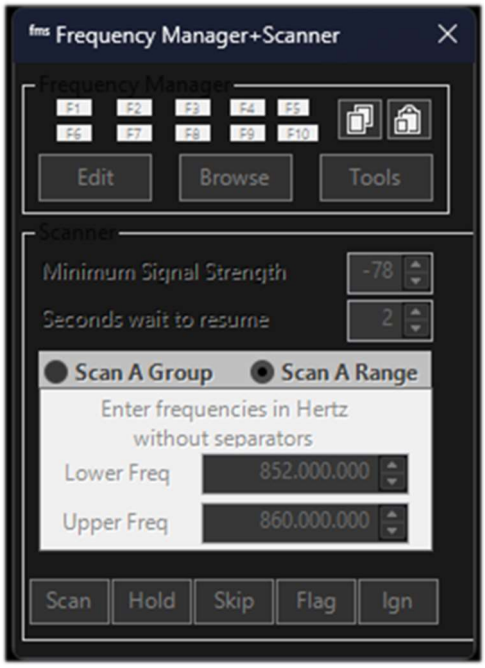
This has recently been updated to make it fully compatible with the newer versions of SDRsharp (1893 and later), particularly in what concerns:

- Themes have been improved, within the constraints of the Microsoft UI controls.
- All plugins now request SDR#/Telerik scrollbar support.
- Minor changes to captions of various controls for greater clarity.
- Many internal changes to improve speed and reliability.

I refer, of course, to all the PDF documentation, which is very rich and comprehensive in every respect, available in the directory at the path: **C:\xxx\FMSuite\FMSuite.Documentation**

Frequency Manager+Scanner

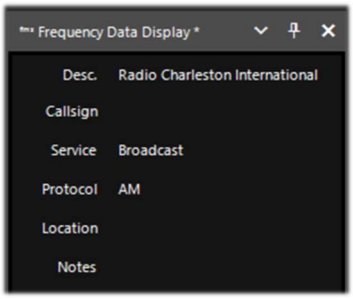
It is the real heart of the Suite, written in C#, it offers a frequency management tool with display of information directly on the RF Spectrum. It makes it easy to search and edit frequencies, scan previously defined intervals or groups of frequencies. These are some other notable features:



- With release 2.3 the scanner runs up to 5 times faster than previous versions, (depending on the computer in use).
- A scanner specifically for the VHF air/ground communication band 118-137 MHz is now included. It correctly identifies and tunes ICAO frequencies with channel spacing of 8.333 and 25 kHz.
- When there are multiple stations on a frequency in your database and the option "Show a signal's description, strength, and timers above the spectrum" is enabled in the Preferences, the letter "M" is displayed next to the dBFS value to indicate the presence of multiple stations.
- You can now selectively choose whether to have the description of a specific station appear in the spectrum window when "Show descriptions of frequencies in the spectrum" is enabled in Preferences. This allows all those less important records in the database to be removed from the RF spectrum, thus reducing the visual impact that occurs when you have thousands of records in the database and use a high sampling rate in SDR#.
- The description of the tuned frequency, if present in the database, is now displayed above the spectrum RF even when the scanner is not operating.

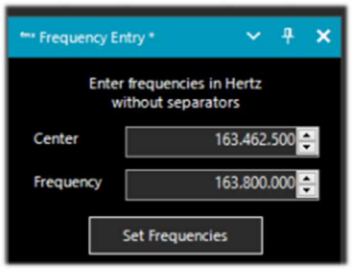
Frequency Data Display

Panel showing the main information if present in the database. In case more than one isofrequency stations are present and the option "Show a frequency's description on the spectrum" is enabled in the Preferences, the legend "Multiple stations are on this frequency" is displayed next to the frequency description".



Frequency Entry

The plugin has been adapted to make it easier to change the frequency and center frequency.

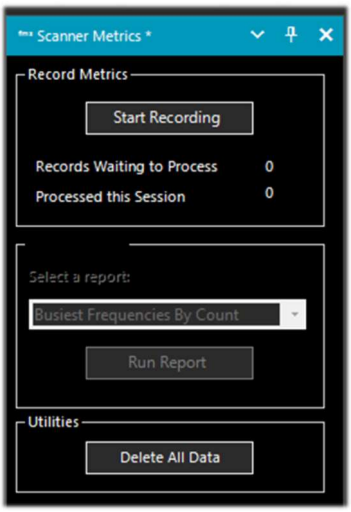


Scanner Decisions

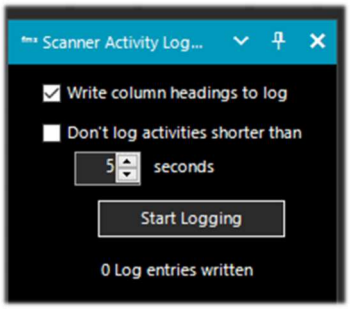
To reduce the amount of informations in the decision list, the frequency appears only on the first row of the reception detail. Tooltips now appear on decision information when the plugin window is too narrow to show the full row. Simply hover the mouse over the row you wish to read. The font size is maintained between SDR# sessions.

Scanner Metrics

It is an accessory of the Frequency Manager+Scanner module. It allows you to record scanner activity in a database and then run analysis on this information. Report graphs are now in colors suitable for color-blind people.



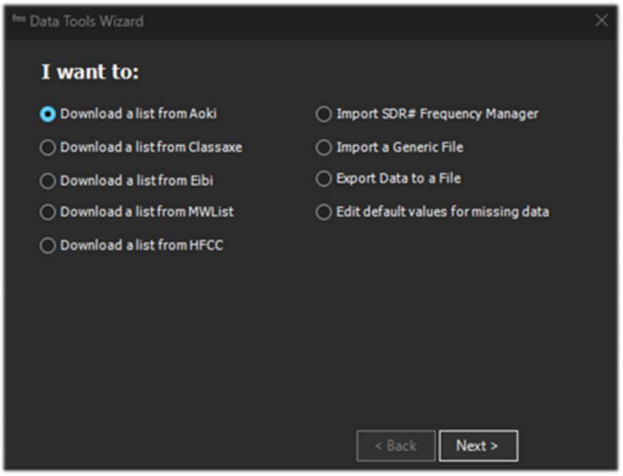
Activity Logger



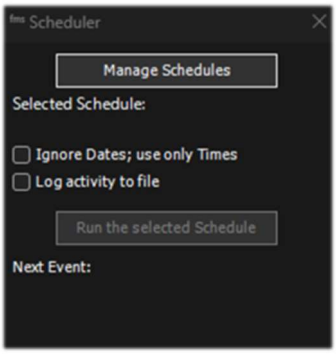
Records the scanner activity created by the "Frequency Manager+Scanner" module implemented in the file name in addition to the date, also with the time of day when the log was created.

Data Tools Wizard

Really useful accessory now that it has become a plugin (previously it was an external program): it is designed to download and import various frequency databases available via the Internet: AOKI, CLASSAXE, EIBI, MWLIST, HFCC.
Import from the SDR# Frequency Manager and generic text files is provided.
The number of imported records is now displayed when the import is completed.
Imported data now also support frequencies beyond Gigahertz.

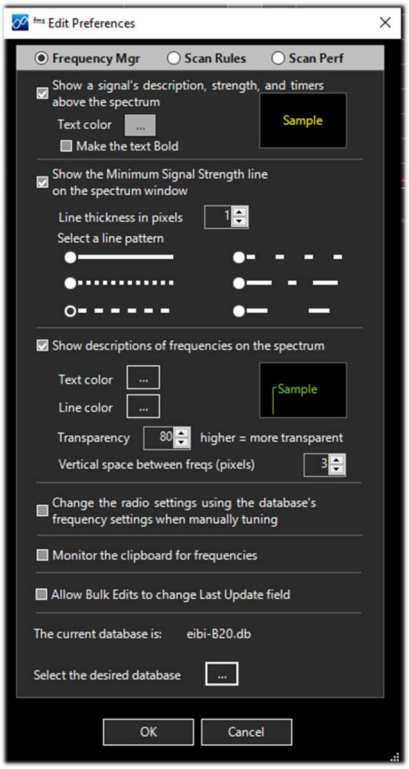


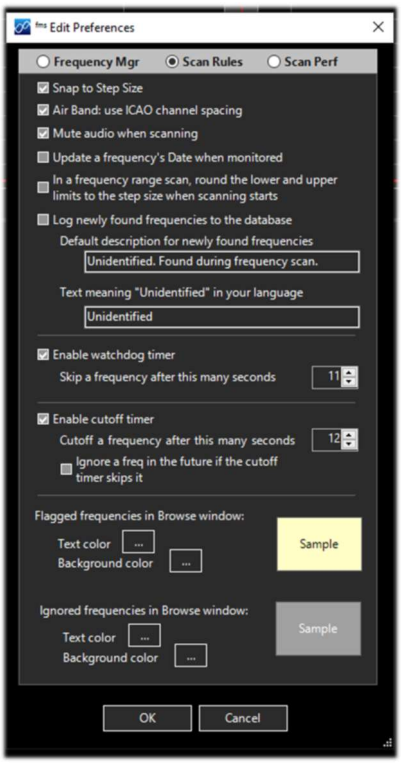
Scheduler

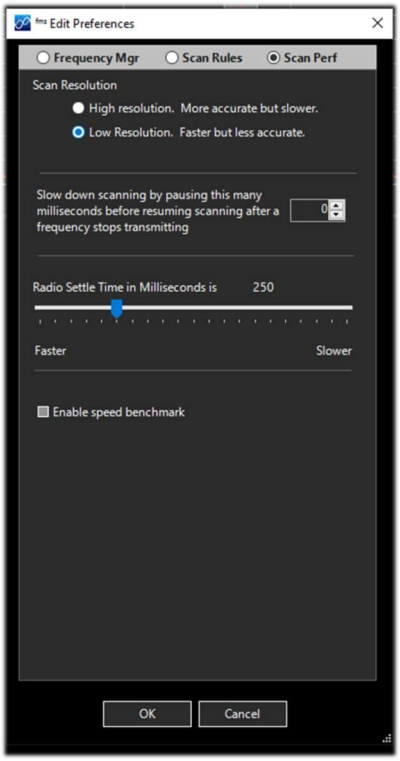


Provides the means to create, use and reuse schedules and manage their frequencies.
Multiple operational schedules can be defined; when a schedule is activated, the Scheduler will activate the frequency on the specified dates and times.

In Tools / Edit Preferences you can see the amount of customizations and consfigurations you can make to the Suite for the "Frequency Manager," "Scan Rules," and "Scan Performances".







In the following screenshot you can see the FMS Frequency Manager+Scanner in conjunction with the newly imported EIBI A21 database. By tuning, for example, the 5,140 kHz frequency in HF, the database, in "Live Track" mode, will be placed at the corresponding frequency, and the stations found isofrequency are highlighted in blue rows.
A label customizable in font and color is also graphically displayed at the spectrum RF.
It is thus possible to create one's own personal archives for frequencies in HF and VHF/UHF, easily importing perhaps those that one has long since entered into one's standard SDR# Frequency Manager.

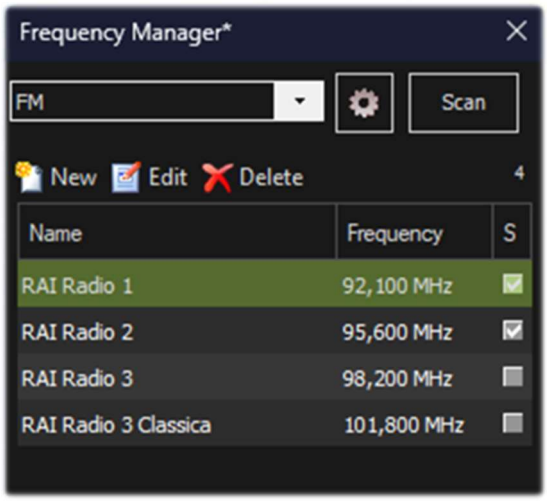
The following screenshots are of editing a frequency in the full-bodied SQLite database of "Basic Info" or "Extended Info" that you can enhance for various archives that are then useful for making specific filters and searches.

In case some problems arise during the installation or use of FMSuite please also refer to the "Troubleshooting" section of the Guides. Also keep in mind that:

- Windows often blocks applications downloaded from the Internet. During software installation, was step 4 in " New Installation" in the "Read Me First" document performed? If not, delete the FMSuite folder and start over with a new installation and be sure to carefully follow the instructions to remove the Windows block.
- The database file could be used by Windows or some other program (antivirus, backup program, etc.). Restart the computer and start SDR# again.
- The FMSuite folder contains a subfolder named "x86." Some people mistakenly delete this folder thinking that, using Windows 64 bit this folder is not necessary, but this is not the case: it is indispensable. **Do not delete any folder from FMSuite!!!**

FreqMan v1.1.9.0 & Fast Scanner v2.4.4.0

These plugins, taken from TSSDR's (Vasili) initials, are now maintained and updated thanks to "Thewraith2008". They are downloadable, with others (Audio Recorder, Aux-VFO, CTCSS/DCS, DSD interface, etc), from the site forum: ***https://www.radioreference.com***



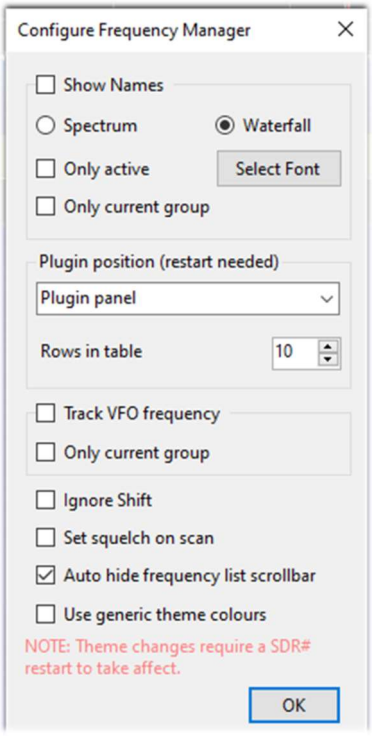
With the "**Frequency Manager**" (or **FreqMan** to distinguish it a bit from the previous one) you can create different groups to store any frequency by assigning a name.
Other parameters such as emission mode, BW filter, centre and shift are automatically detected by the current VFO of SDRsharp.

The very useful thing is that FreqMan uses the same archive as Frequency Manager (i.e. the file 'frequencies.xml' in the program directory). So both plugins will conveniently see the same groups and frequencies…

For those who want to experiment, MS Excel conveniently edits the "frequencies.xml" file, thus expanding the possibilities for better managing and cataloging the various stored frequencies.

You can create many different groups according to your needs: HF, VHF/UHF or by genre, for example FM broadcasters, amateur radio, satellites, etc. ...

On the right, in this version of "FreqMan", there is a very useful checkboxes "S" that allows you to mark each record and then scan it by pressing the "Scan" button. Below the latter is a handy counter that indicates the memories currently saved in the group ("FM" in our example).



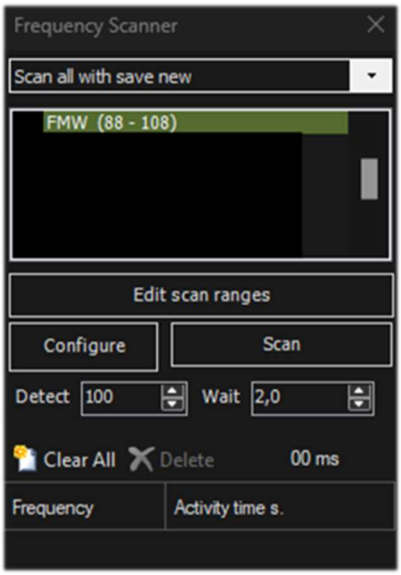
By pressing the wheel button  instead you access the configuration panel where you can customise other options including the possibility of displaying a label on the Waterfall or Spectrum (with a font of your choice), of the only active frequency or of the current group, etc. etc...

With the "**Frequency Scanner**" (also known as "Fast Scanner") it is possible to search in a wide range and with impressive scan speeds that cannot be achieved with any other scanner, even analogue!
There are two modes: the more immediate mode of searching in the current spectrum window using the "Screen" preset, or defining a scan range in priority by pressing the "Edit scan range" button for example with this data:

Edit Range

Name	Start (Hz)	End (Hz)	Detector	Bandwidth	Step size	Group
FMW (88 - 108)	88.000.000	108.000.000	WFM	130.000	100.000	fmw

You can take advantage of as many as 5 different scanning modes: **Scan all with save new, Scan all without save new, Scan only memorized exclude new, Scan only new exclude memorized, Scan only enabled in Manager.**



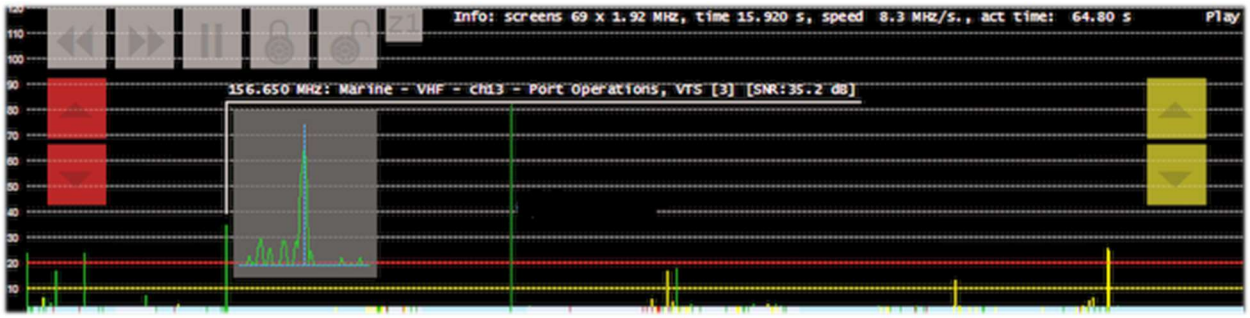
The "**Configure**" button allows you to set every possible parameter of the Scanner, Channel Analyzer and Logging to file in detail.

The "**Detect**" button allows you to vary the scanning speed allowing the best detection of an active signal. *The default value is 100.*

The "**Wait**" button allows you to vary (in seconds) the delay with which to resume scanning. *You can start testing with a value of 5 seconds.*

At this point you are ready to press the "**Scan**" button to see and appreciate the extreme speed of scanning (still improved in version 2.2.1x for CPU and scan speed!) and the wealth of supporting information.

In this author example, the nautical band is being scanned. The Channel Analyzer window will appear



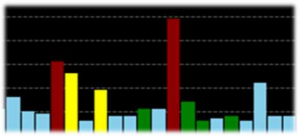
with a rich set of indications and operational buttons. Let's see how to use them:

- The buttons << >> control the scanning direction or to skip the current active frequency
- With | | to pause or resume scanning
- Use “locks” to lock/unlock one or more frequencies
- The Z1/Z2 buttons toggle the zoom type in the channel analyser window

While the following buttons control interrupting and resuming the scan:

- The red ones adjust the level of the "trigger" (red horizontal line). *When the signal goes above the red line the scan stops and you can listen.*
- The yellow ones adjust the "hysteresis" level (yellow horizontal line). *When a signal goes below the yellow line, the countdown (for waiting) starts. When the time is up, scanning resumes. If in the meantime the signal goes above the red line again, during the waiting period, the counter will be reset and the scanner will remain on the current frequency.*

The colours at the bottom of the Channel Analyser have these meanings:

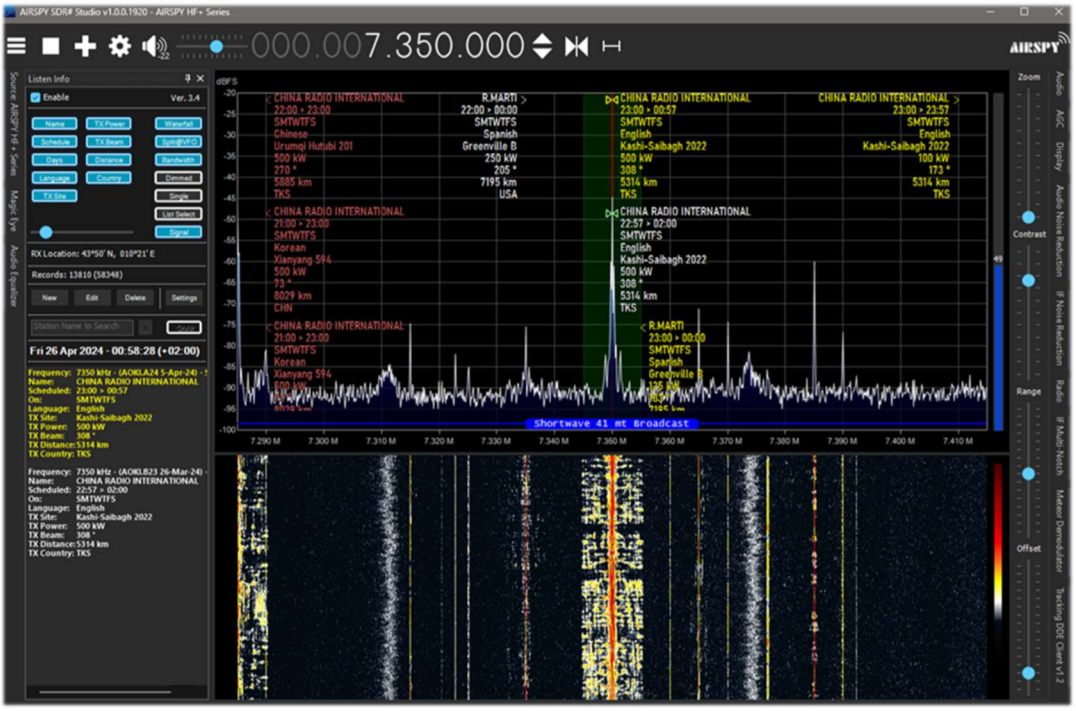


BLUE = The frequency is not present in the associated Frequency Manager database and is not locked.
DARK RED = The frequency is not present in the Frequency Manager database but is locked.
YELLOW = The frequency is present in the Frequency Manager database but is locked.
GREEN = The frequency is in the Frequency Manager database and is not locked.

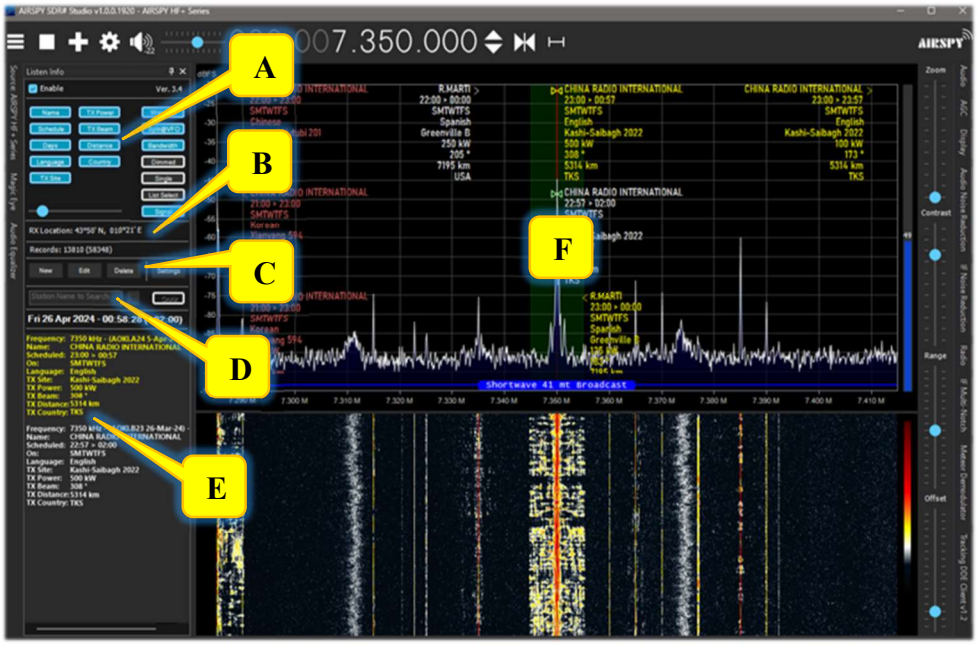
***Note**: Be sure to turn off radio Squelch since this function does essentially the same thing. Simply put, if I want to hear weak signals, I set the red line just above the background noise. However, this will result in more false stops and then it will raise the line a little bit...*

Listen Info v3.4

With SDR# we make use of hearing since its best is revealed precisely through sound and many of its special features involve suppressing noises and emphasizing audio. This allows us to pay attention only to what we are interested in, somewhat like a skilled musician manages to separate and balance the sounds and tracks of a song to be mixed...



Now, thanks to Listen Info for .NET7 (abbreviated "LI") by Marco Melandri (BlackApple62), an energetic visual alchemy will also be established: one simultaneously listens to and visualizes on the RF Spectrum / Waterfall the valuable information related to the broadcaster: without this aid, the signals would remain only completely evanescent graphic signs...



It looks like this: on the left side I opened the plugin panel and on the RF Spectrum (or waterfall) all the information we have ever dreamed of are immediately available, actually more!

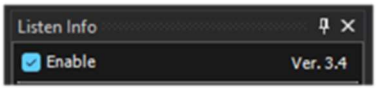
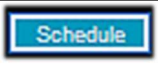
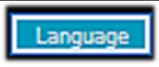
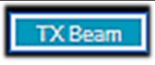
Following my motto "From black and white to color... now Listen Info takes us to a higher level of perfection."

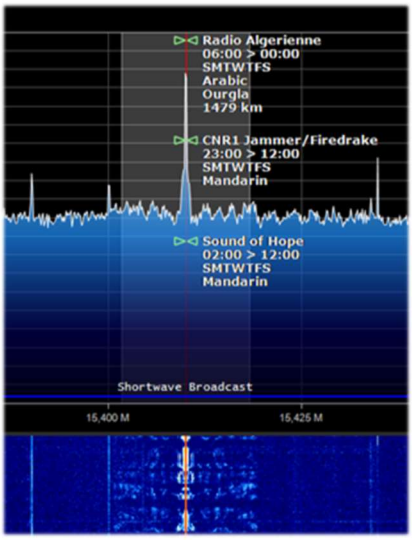
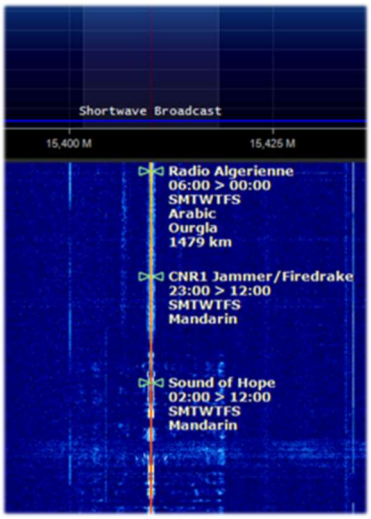
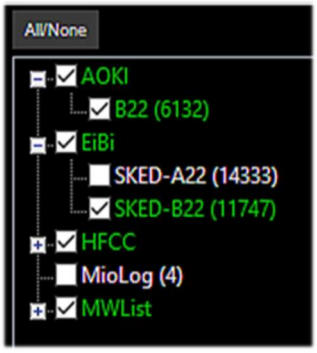
It has been my absolute honor and privilege to be a betatester, and with LI now I have radically changed the way I do radio listening.

It covers any band: LW, MW, SW, VHF and UHF. Imports the following lists directly from the web: AOKI, EIBI, HFCC, MWList while acquiring from local files: SDR# Frequency Manager, ILGRadio and SWSKEDS, providing on the Spectrum/Waterfall customizable, dynamic, and colorful font informations about the broadcaster you are receiving, also allowing you to do targeted searches within the multiple supporting databases and all in real time!

Reference link: ***https://github.com/BlackApple62/SDRSharp-ListenInfo-Plugin***

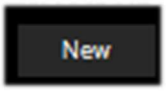
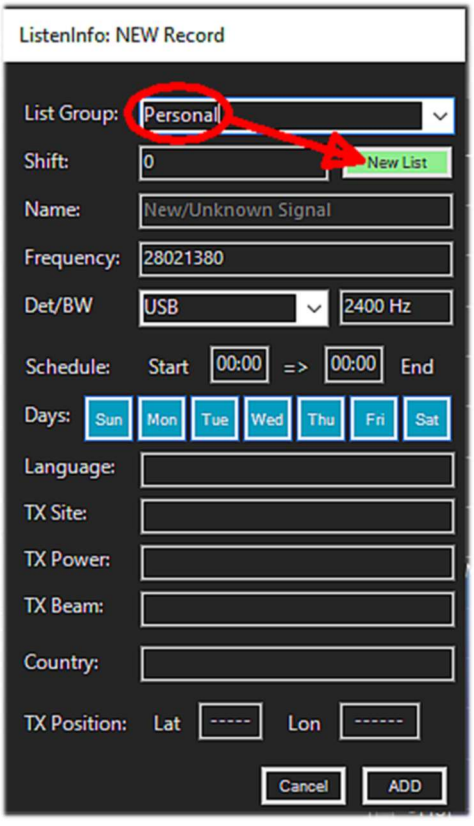
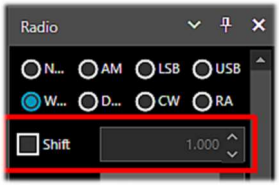
Let us analyze in detail the various features of the individual "A / F" panels.

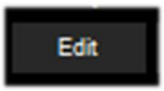
.Panel "A"	Feature
	The "Enable" flag activates the plugin. On the **right** appears the version number *and clicking on it takes you directly to the plugin download link.*
The following buttons, once activated, will take on the color blue **These allow the respective labels to be displayed on the RF Spectrum/Waterfall and table.**	
	Displays the name of the station (or stations if there is more than one).
	Displays the UTC time of the broadcast.
	Makes the days of the week visible (S M T W T F S format).
	Displays the language of the current program.
	Displays the site of the transmitting station.
	Displays power in kW of the transmitter (if listed).
	Displays the degrees where the transmission beam is oriented.
	Displays the distance in kilometers from the transmitter (calculated from your own geographical coordinates, see "RX position" and those of the transmitter site).
	Displays the country of the transmitter.

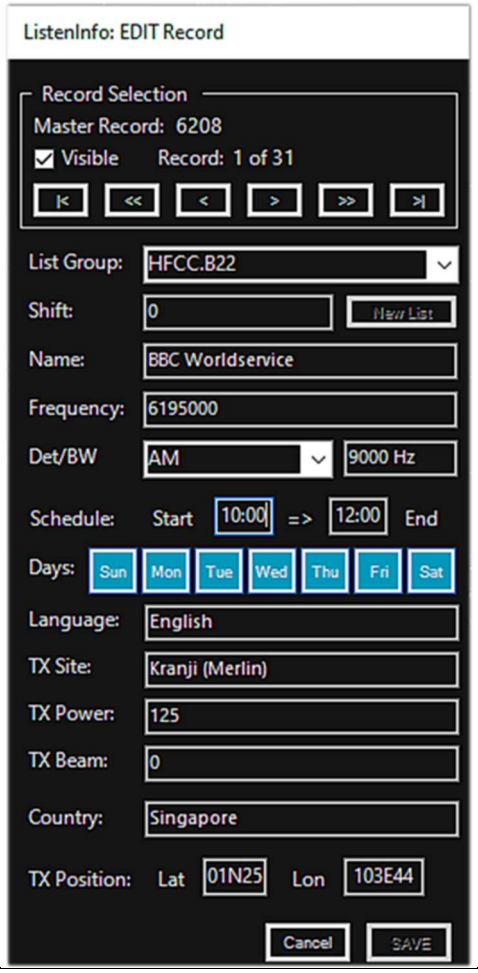
While these have the following features.	
	Toggles the display of RF Spectrum or Waterfall information.  
	If enabled, prioritize the visualization of the tuned frequency records, against nearby.
	Allows to display labels related to the current bandwidth.
	Makes the labels a very muted color. Used in conjunction with the "Bandwidth" button displays broadcasters in the current bandwidth.
	Makes visible only the active broadcasters transmitting on the frequency tuned by the VFO. If disabled on the RF Spectrum, the info of all active broadcasters visible in the RF Spectrum window appears.
	Opens in the tabular part of the plugin (see point "E") the complete list of loaded lists with the relative number of records next to it (in parentheses). The "All/None" button allows you to mark or unmark all lists in the table. Removing the flag from a list also updates the corresponding number of records in the "C" panel.  In hierarchical lists, the green color indicates when a sublist is available (e.g., different schedule A22/B22 and area in the MWList) and at least one of the sublists is active. In the example the lists "EIBI SKED-A22" and "MioLog" appear white since they are not active. The + sign also indicates the presence of a sublist.

	Only records of broadcasters for which the signal (S) can be estimated to be receivable (with the cursor from highest to lowest) are filtered and displayed. It is currently calculated only by distance and power of the transmitting site for those lists that show such indications. 

Panel "B"	Feature
	Indication of one's receiving location. *See "Settings".*

Panel "C"	Feature
	Indication of records in the database. The first number is for those filtered and therefore visible, the second (in parentheses) is the number of total records in the database.
	 With this panel, you can enter a new record, related to the VFO frequency tuned, in the list you want (for example with the name "Personal" and then clicking the "New List" button). The emission mode and filter bandwidth are automatically detected by SDR#. The "Shift" field allows you to value the equivalent field of SDR# in the use with the SpyVerter or any converters..  All other fields are freely editable respecting only the relative adequacy. The "ADD" button will complete the action or "Cancel" to exit without any entry.

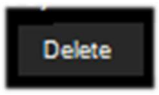


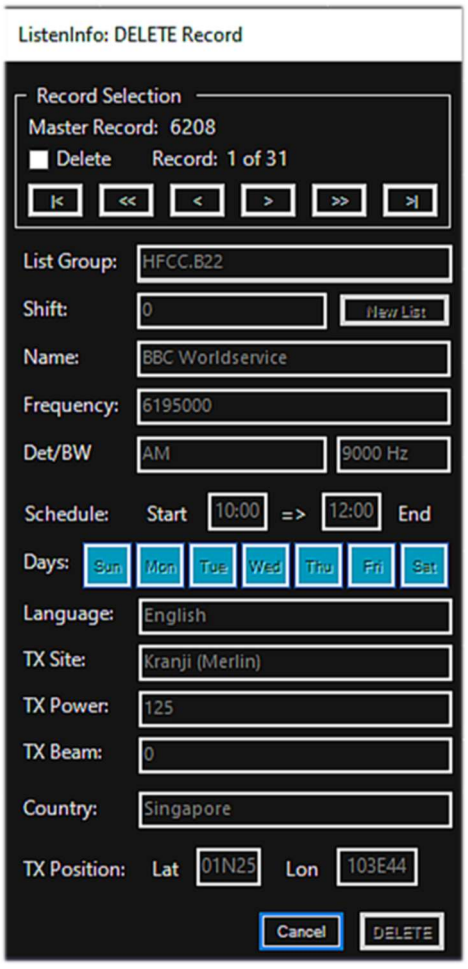


With this panel you can change the content of any record in the list, related to the frequency tuned in the VFO.
The "Visible" flag makes the record active or invisible during normal use.
With the following buttons you will be able to select:
the first record, back by 10

records, back by 1 record, forward by 1, forward by 10, last record.

With the "SAVE" button you will complete the action or with "Cancel" to exit without any change.

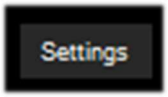
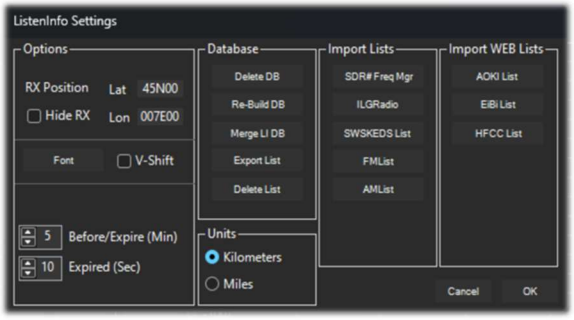




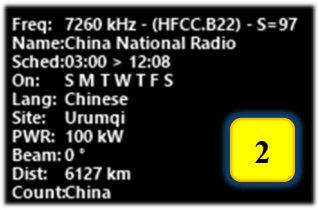
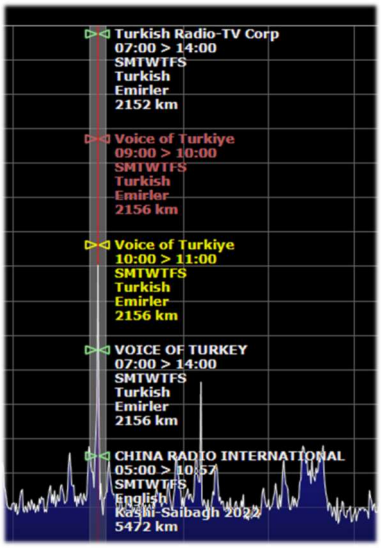
With this panel it is possible to delete a record.
The "Delete" flag is used to mark, among any several records, the one selected for deletion.
With the following buttons you will be able to select:
the first record, back by 10

records, back by 1 record, forward by 1, forward by 10, last record.

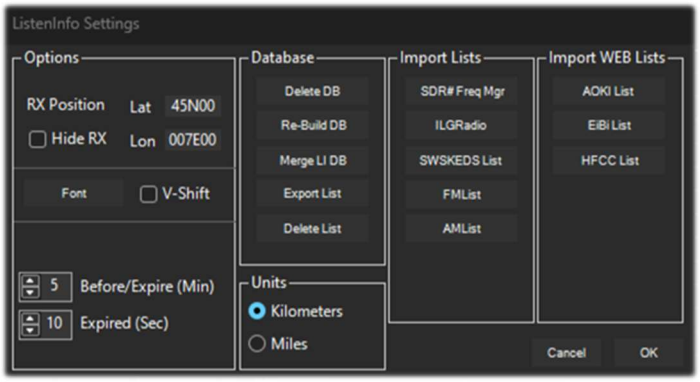
With the "DELETE" button you will complete the action or with "Cancel" to exit without any deletion.

	Various settings of the plugin can be configured with this panel.  *I write a separate paragraph to address the various topics more exhaustively.*

Panel "D"	Feature
	It only takes three letters, regardless of the position of the string (at the beginning, middle, or end of the name) to search for the occurrence in the loaded lists. The number of records found also appears in the title.  *Example by typing only "radio" it may appear either "Adventist World Radio" or "China Radio Int." or "Radio Farda" and all the others... Really very performant and convenient!!* The tabular part shows, on the right and bottom, sliders to move more conveniently. The "x" flag is used to reset the search field. *Another unique feature of LI is that clicking on any record in the table centers the VFO of SDR# on the chosen frequency by tuning it with the exact emission mode and bandwidth present for that record in the database.*
	In the search, it filters only active broadcasters from the lists according to the current UTC time.
	Indication of the day, month, year and UTC time taken from your own computer.

Panels "E" and "F"	Feature
   	The informations that can be displayed, on the side of the vertical tuning bar beside the Spectrum pointer (*), for panel "F", are as follows:  • Station name • Active schedule (start > end transmission) • Days of the week • Program language • Transmitter site • Power (in kW) • Bearing antenna • Distance (in km) • Country In the tabular part (panel "E"), which you can scroll through with the side cursor, we will have in addition to the previous information: • Frequency - (reference list) - Signal Clicking on the table, with the right mouse button, loops between the three modes provided in order to optimize the available space (see alongside screenshots 1/2/3). The different color codes, active only in HF where schedules are present, have this meaning (see also "Settings"): white color (default) – Transmission in progress yellow color – Notice next start of broadcast (5 minutes before). Salmon color – Advance notice of end of broadcast (duration 5 minutes). Red color – End of transmission. The label will disappear after a few moments assuming some new transmission still follows, otherwise nothing will be displayed (default 10 seconds). ***Note**: If the VFO is set to "Free tuning" and we are toward the right edge of the screen the respective LI indications will appear on the left side of the vertical tuning bar: no small feat for those who designed and built it!!*  *(*) Spectrum pointer in the first row in the Spectrum or the Waterfall. It allows a very precise pointer with respect to the frequencies detected in the database. In fact, it takes only 1 Hz difference that such pointer changes graphically in the < or > symbols.* 

Settings



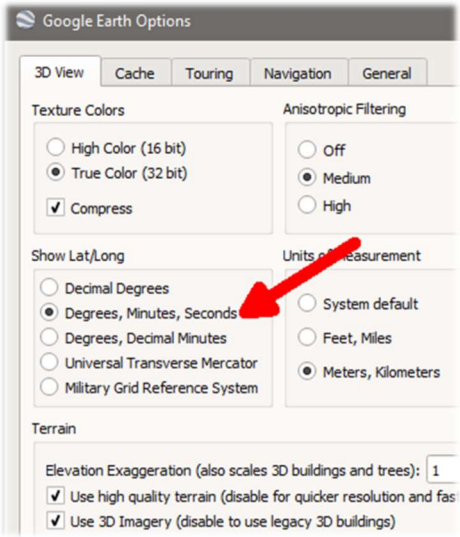
OPTIONS: RX POSITION

Fill these fields with your receiver's Geographic Coordinates is quite important, as LI will use them to estimate stations distance and Signal fields. With the "Hide RX" option, you can hide the coordinate display from the main panel.

If you wish to change these values you will need to provide "Re-Build DB" in the DATABASE field (see below).

The format used is as follows: **36N09 - 139E48 or 32S58 - 071W30**

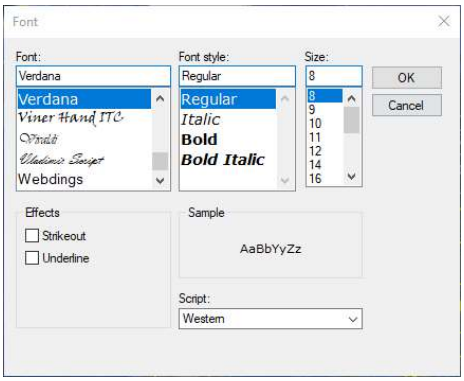
The latitude ranges from 0 to ±90 degrees, and longitude, from 0 to ±180 degrees. The sign is indicated by (0=Equator, or Greenwich Meridian): N, E giving positive values while S, W give negative values.
For Longitude the format is aaa(E|W)bb with 0<=aaa<+179 and 0<=bb<=59 aaa in degrees, bb in 60ths.
So Longitude of -8 degrees and 6 sixtieths should be written as 008W06.
Similarly Latitude has format cc(S|N)dd where 0<=cc<89 and 0=<dd<59, same units.



Those who want to verify exactly their geographic coordinates in the correct Listen Info annotation I suggest taking advantage of Google Earth (from the menu: Tools / Options / 3D View and using Degrees and Minutes).

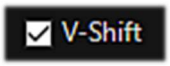
OPTIONS: FONT

It is possible to customize the fonts that appear on the RF Spectrum by Font Type, Style, and Size Points.



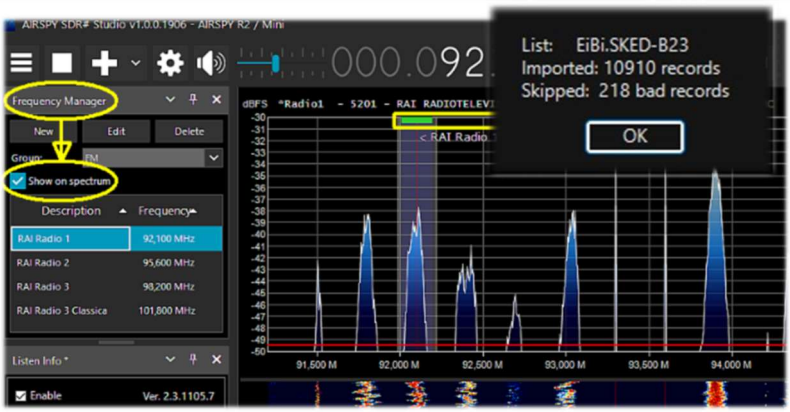
On the other hand, it is not possible to change the color of the font since it is already used internally with specific meanings (see panels "E" and "F" above).

OPTIONS: V-Shift

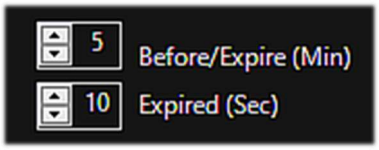


The control allows the content of the information appearing on the RF Spectrum to be moved vertically down (by one line).

It can be useful if used in conjunction with the "Frequency Manager" plugin with the "Show on spectrum" option enabled.

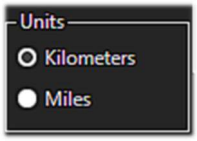


OPTIONS:



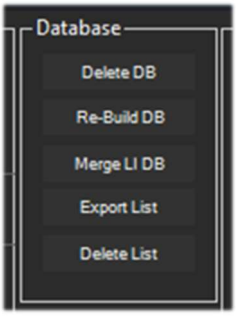
These two controls allow customizing the exposure time on the RF spectrum of the colored labels in minutes (for yellow/salmon colors-"Before/Expire") and seconds (for red-"Expired").

UNITS:



This control is for displaying distances in kilometers or miles.

DATABASE



Delete DB - with this command you delete the entire database.

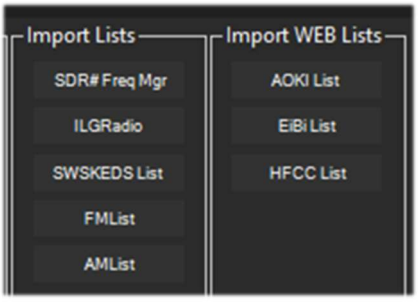
Re-Build DB - with this command you re-index the database after acquisition/deletion operations and/or change geographic coordinates.

Merge LI DB - with this command you can add a custom database to the current database.

Export List - this command allows you to select the lists to be exported. The exported files are saved in the subdirectory "LI-Data\DBExports" with format "list name.csv" and are of course re-importable.

Delete List - with this command you can select the lists to be deleted.

IMPORT LISTS/WEB LISTS



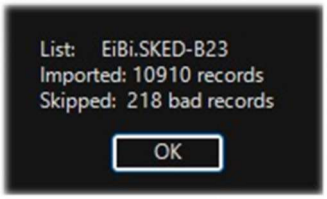
The databases or frequency lists currently managed and importable are as follows depending on whether they are acquired from local files:

- SDR# Frequency Manager (selecting the frequencies.xml file from SDRsharp)
- ILGRadio (from “ilgaacsvzip” without unzipping it!)
- SWSKEDS List, from BC schedule of Dan Ferguson (directly from the zipped file without unpacking it)
- FM/AM Lists

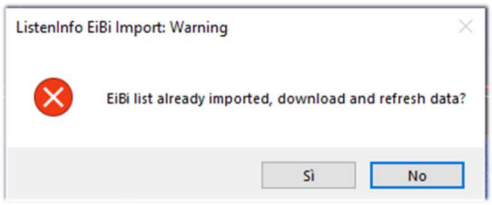
Or directly via internet:

- AOKI List
- EIBI List
- HFCC List

Note: For some lists from the Internet it can sometimes happen that some records are discarded in the import phase. Nothing of concern or critical, but the procedure discards and flags all those records that have content that does not conform from what is expected in Listen Info.



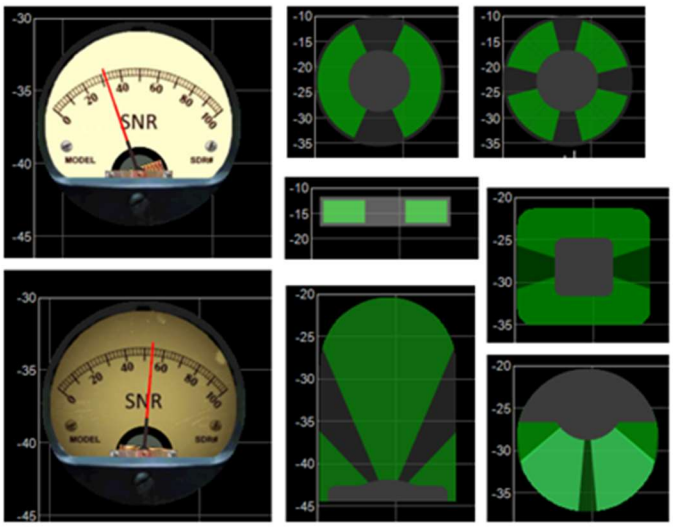
If a List has been previously imported, you will be prompted to select keeping or refreshing existing data.



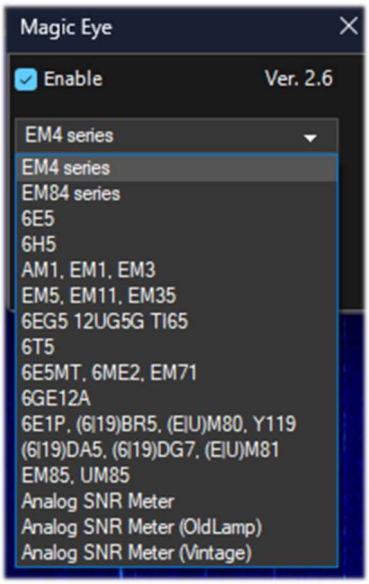
Magic-Eye v2.6

On an old-style note, in this world of ultratechnological software, you might like to try the freeware plugin "Magic eye" of old memory, by the author Marco Melandri (BlackApple62):

https://github.com/blackapple62/SDRSharp-Magic-Eye-Plugin

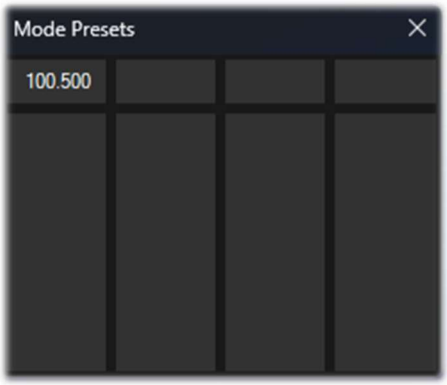


Once installed and activated one of sixteen available patterns, customizable in size and transparency relative to the background, including three analog-type SNR meters, will appear in the upper left corner of the RF Spectrum window.



Mode Presets v1.4.6.0

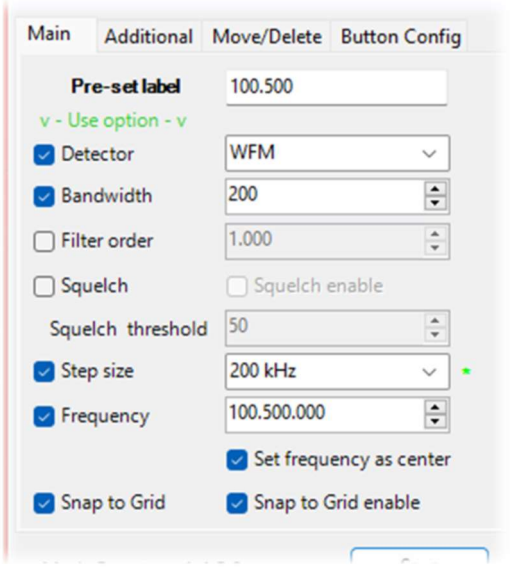
The plugin by "Thewraith2008," already cited several times for his numerous works, allows the user to store some of the most common SDR# settings in convenient preset and customizable buttons (in rows and columns).



In the example on the side, I stored the frequency "100,500" in WFM in the first button.

By right-clicking on each button, you can customize many more parameters.

In my case you can see those enabled in the "Main" tab with the Detector, Bandwidth, Filter, Step size etc. etc.



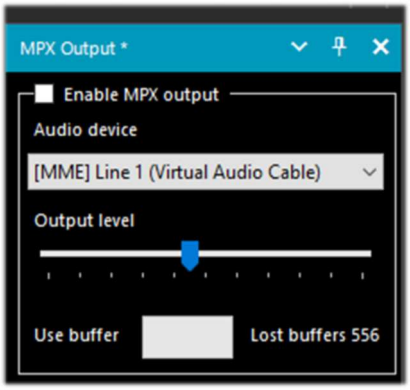
Up to a maximum of 50 rows and 10 columns are allowed to be managed…

.

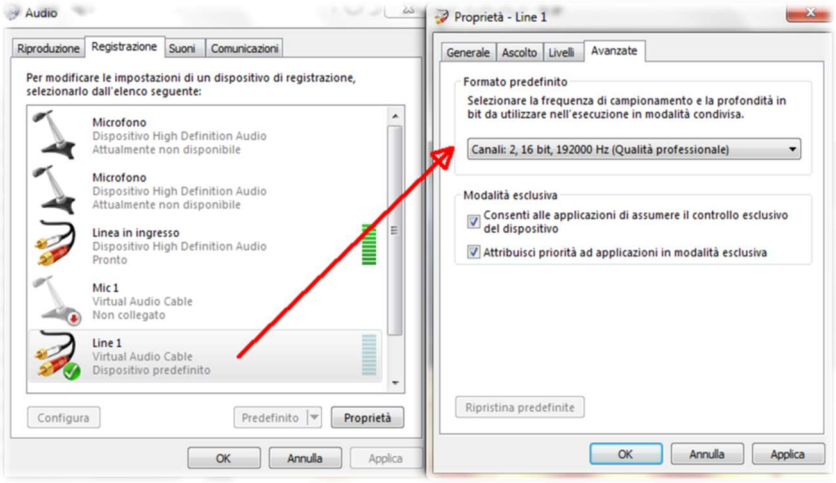
NOTE: The BandPlan plugin option "Automatically update settings" must be turned off for this plugin to work properly.

MPX Output v0.2.1 and RDS-Spy

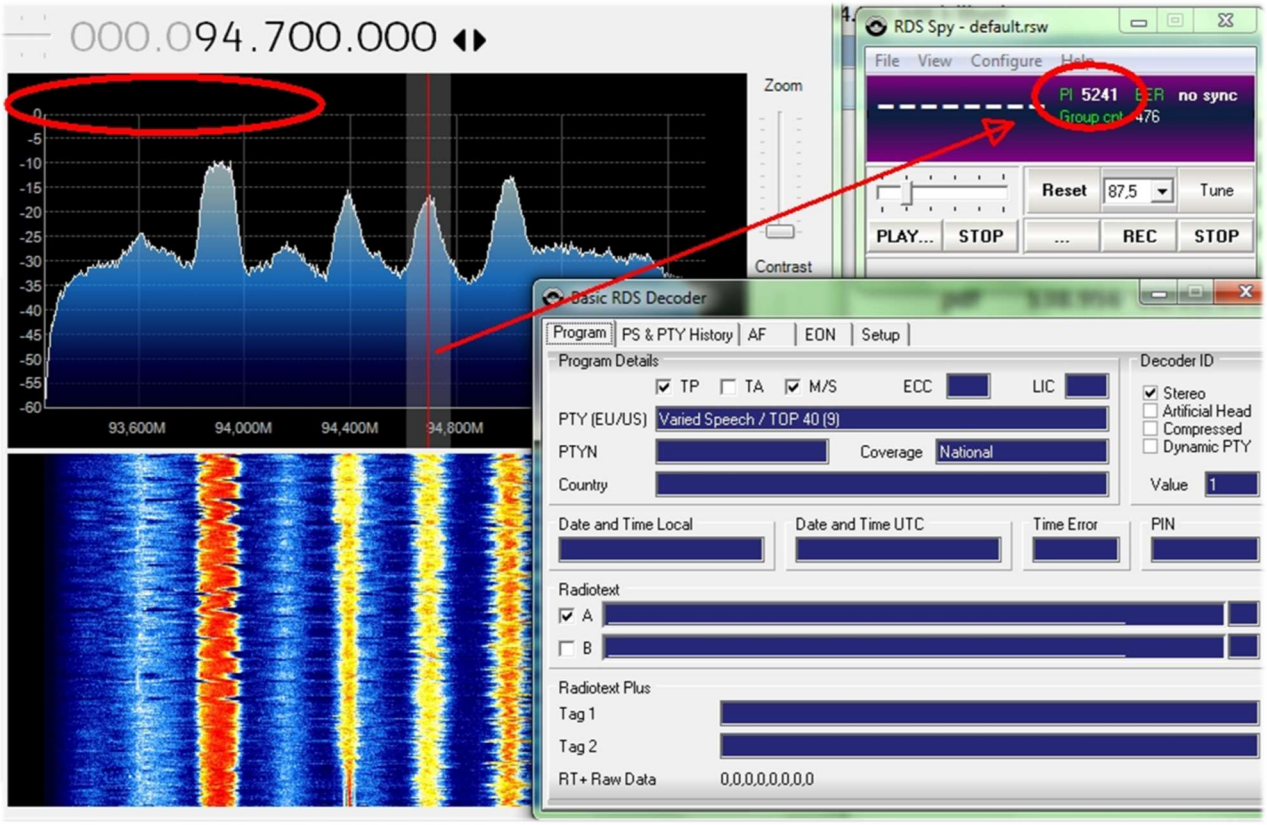
For a friend who is interested in FM-DX I tried the new plugin "MPX Output" in combination with the professional decoder "RDS-Spy" which allows you to discover and highlight all, but really all, the "secrets" hidden inside the RDS: ***https://rdsspy.com/downloads/***



The system is really performant and sensitive, *and before the v186x of SDR# that changed things a lot,* often it hooks the PI even before they are detected by the SDR# integrated RDS decoder (see below the picture with the immediate PI detection). *For this, however, it is necessary that your sound card supports 192 kHz sampling in recording and that this is enabled in the audio panel (as shown below), only then you can have the decoding of the RDS.*

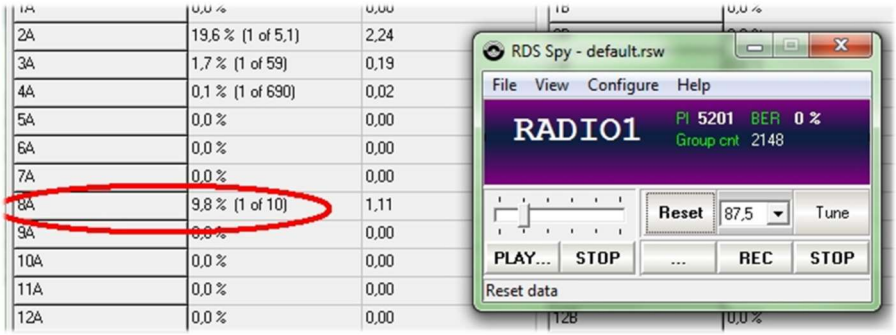


By enabling the checkbox "Enable MPX output" the multiplexed audio stream will be routed to the indicated device and from this to the RDS-Spy decoder which will be configured in the panel "Configure / Select RDS Source / Sound Card / Input Mode "Direct RDS/MPX (192 kHz)" with the same device selected in the MPX plugin.

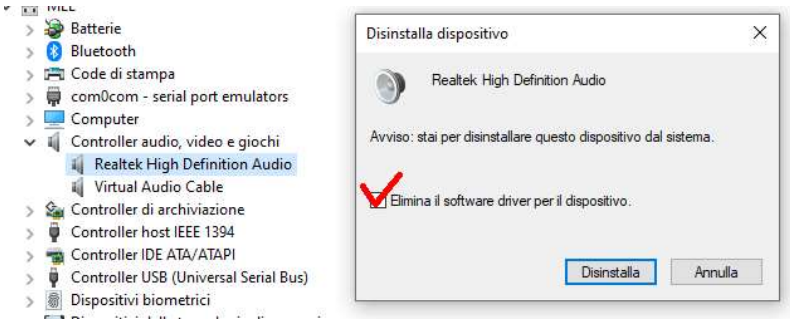


At this point the decoder panel will animate with all the RDS information and in "View / Basic RDS services" you can appreciate the multiple indications "Program Details, PS & PTY, AF, EON". In the "Group Analyzer" all the active groups will be checked with their percentage of diffusion in time...

In the following example I found for the RADIO1 broadcaster the presence of the TMC service - Traffic Message Channel on block 8A

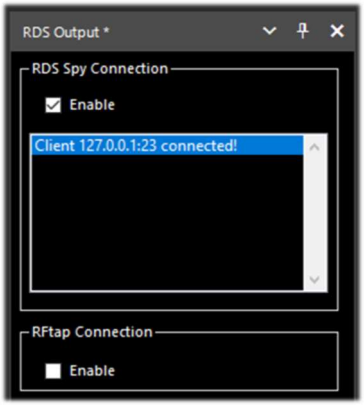


In the my tests I found some difficulties to set the 192 kHz sampling that was not present in my W10 operating system despite the drivers were updated, then reading a thread on the net, someone suggested to uninstall the drivers of device also marking the highlighted field.



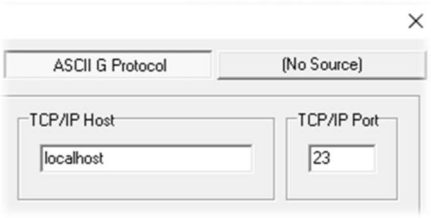
Upon restarting Windows the system was correct…

Only on a laptop I didn't succeed and so, on a friend's suggestion, I tried another way with the plugin "SDRsharp RDSOutput" that allows to use RDS-Spy but without MPX, Virtual Audio Cable and the whole issue of sampling and configuration for 192 kHz.

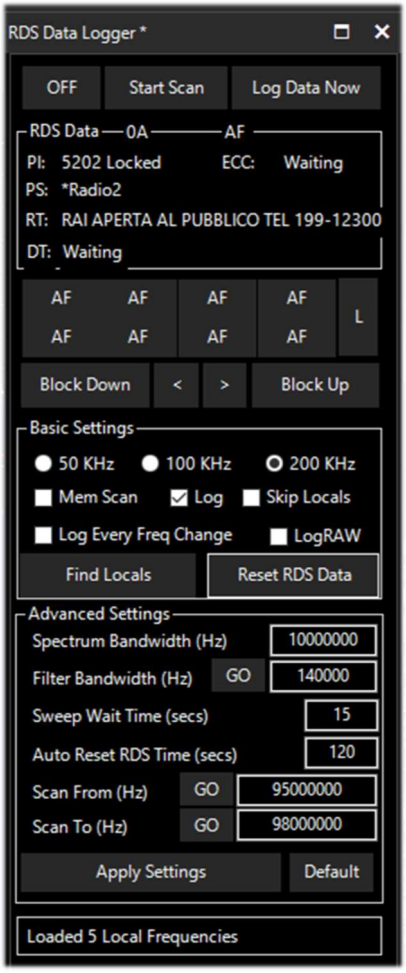


The "trick" is done by using the TCP/IP protocol and taking SDR#'s internal RDS decoding. The RadarFolf plugin is available here: ***https://github.com/RadarFolf/RDSOutput***

After unpacking the DLL, in the usual SDR# directory, configure RDS-Spy in the "Source / ASCII G Protocol" menu with these settings: localhost, port 23. Then click in RDS-Spy on File / Play Stream...



RDS Data Logger



This is not a new plugin, in fact the last update was in 2018, but it is fully functional even with the latest SDR# releases. As you can easily guess from the name, the plugin by Andy (M0CYP) is an RDS data logger useful in the skilled art of FM-dx.

Free download at: https://www.apritch.co.uk/sdrsharp_plugins.htm

It allows recording with date/time of RDS data (PI, PS, RT, etc.) detected on single frequency or with sweep function from 87.5 to 108 MHz (or individual portions). Parameters such as filter selectivity (Bandwidth) and sweep time (Sweep time) can also be specified to operate best under all conditions.

Another interesting function is the one under the "Find Locals" button, which I will explain in more detail later. Basically, all stations with RDS are logged and only local frequencies are entered into the RDSDataLogger-Locals file.

This list could then be used to do new scans (perhaps during ES, Tropo or scatter activities) skipping precisely the local frequencies to devote only to finding DX ones!

The important thing during the installation phase is to respect the location of the following files in these paths:

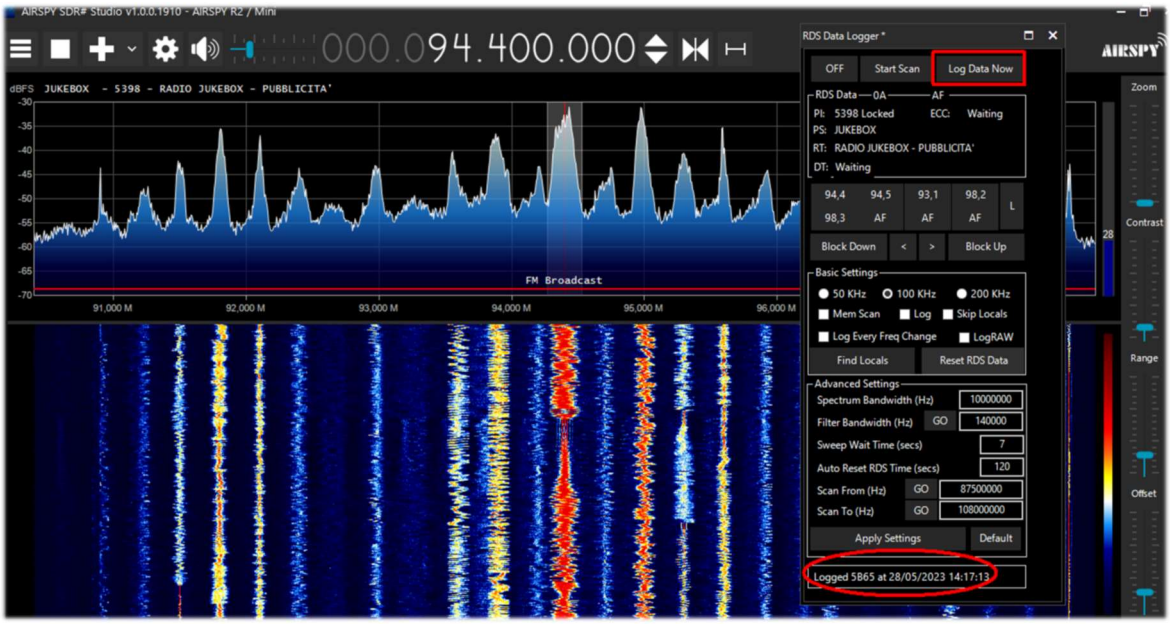
SDRSharp\Plugins\SDRSharp.RDSDataLogger.dll

SDRSharp\RDSDataLogger\RDSDataLogger-MemBank.CSV

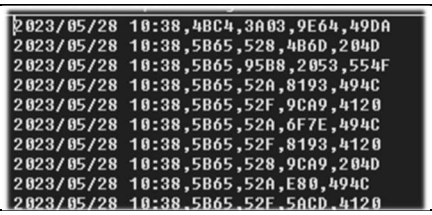
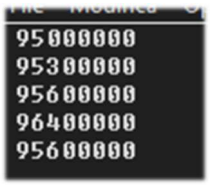
Then edit the "plugins.xml" file by adding the following line before the </sharpPlugins>:

<add key="RDS Data Logger" value="SDRSharp.RDSDataLogger.RDSDataLogger,SDRSharp.RDSDataLogger" />

In the following screen is enabled the "Log Data Now" function on the single station of my interest.



These are the detailed functions of the plugin:

ON / OFF	Plugin startup/shutdown.
Start scan	Start the scanning.
Log Data Now	Saves the current RDS data.
Block Down / Block Up	Shifting the tuning down or up.
< / >	Tuning shift by steps (50/100/200).
50/100/200 kHz	Tuning steps.
Mem Scan	To scan for some desired frequencies, simply edit the file RDSDataLogger-MemBank.csv
Log	To automatically save with local date/time all RDS data in the RDSDataLogger-yyyy-mm-dd.csv file (in the RDSDataLogger directory).
Skip Locals	To activate DX scanning and disregard Locals stations, you click the "Start Scan" button after flagging the "Skip Locals" button. See also "Find Locals"...
LogRAW	 To automatically save RDS data in RAW format.
Find Locals	 By starting a scan, the active stations with RDS, are automatically inserted into the RDSDataLogger-Locals file but only with the frequency (screen alongside). The list should then be used to make new "band scans" during FM-dx sessions skipping precisely the local frequencies not useful for the purpose.
Reset RDS Data	To reset the decoding of RDS data displayed at the top of the plugin.
Spectrum Bandwidth (Hz)	Using 10 MSPS or 2.5 MSPS sampling, frequencies approaching the band extremes (right/left) may lose quality and RDS data may not be reported correctly. This window then allows you to select a lower portion of the spectrum than that displayed in the Waterfall/Spectrum.
Filter Bandwidth (Hz)	To change the selectivity at will.
Sweep Wait Time (secs)	To set the time for which the scan stops on each frequency in to analyze the data. By default it is 7 seconds.
Auto reset RDS time (secs)	To set every how many seconds the RDS is reset when you are not scanning. By default, the time between each reset is 120 seconds.
Scan From (Hz) / Scan To (Hz)	To set the start/end frequencies of the scan with the previously set step. When the highest frequency is reached, the scan will automatically restart in a continuous loop.
Apply Settings	After any changes in settings this button should be pressed.
Default	To restore values to their original conditions.

This is an example of the data collected in the RDSDataLogger-yyyy-mm-dd.csv file, which can then be opened in Excel to do further sorting, filtering, selection, and analysis as desired...

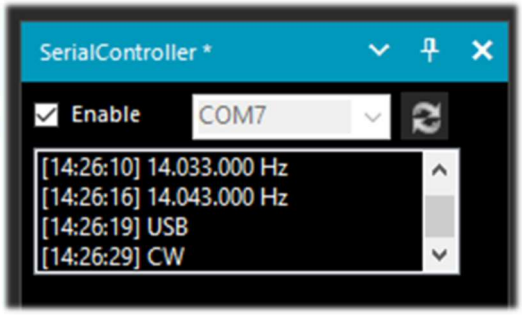
```
2023/05/28 10:25,95000000,5000,Locked,MAN0 A,2DI           E LE    NA Q,100,3 / 100,4 / 103,1 / 105,1 / 91,9 / 91,3 / 95,3 / ,Waiting
2023/05/28 10:27,95300000,5000,Locked,R. +RINO,,95,3 / 91,9 / 91,3 / 100,3 / 100,4 / 103,1 / 105,1 / ,Waiting
2023/05/28 10:28,95600000,5202,Locked,*Radio2,RAI APERTA AL PUBBLICO TEL 199-123000,96,5 / 93,4 / 93,9 / 104,9 / 97,1 / 93,5 / 92,7 / 93,8 /
2023/05/28 10:29,95900000,5299,Locked,e,NG9,Waiting,Waiting
2023/05/28 10:30,96400000,5348,Locked,H     *,TOLD     SO     ` J     4   (202     SM     I,Waiting,Waiting
2023/05/28 10:32,97300000,5219,Locked,CAPITAL,Save Your Tears   AP  WEEK,90,1 / 93,1 / 95,8 / ,Waiting
2023/05/28 10:34,97900000,5B68,Locked,* RTB *,la vera alternativa,Waiting,Waiting
2023/05/28 10:34,95000000,5B65,Locked,MANILA,O  (NILA    E LE        UANDO VUOI,103,1 / 104,4 / 104,5 / 105,1 / 88,9 / 100,3 / 89,4 / 95 / 9
2023/05/28 10:35,95300000,5000,Locked,R.TO* NO,,95,3 / 91,9 / ,Waiting
2023/05/28 10:36,95600000,5202,Locked,*Radio2,. "         LITA,93,3 / 97,5 / 95,6 / 96,5 / 93,4 / 93,9 / 104,9 / 96,9 / 92,5 / 91,9 / 92,4 /
2023/05/28 10:36,95900000,5299,Locked,ZETA   \,NI<`0,Waiting,Waiting
2023/05/28 10:38,96400000,5348,Locked,*DISCO*,X (2         ICKY JAM FT.,Waiting,Waiting
2023/05/28 10:38,95000000,5B65,Locked,MANraA,RADIO MANILA              UANDO VUOI,95 / 98,4 / 100,4 / 102,2 / 103,1 / 104,4 / 98,6 / 100,1
2023/05/28 10:43,95300000,5000,Locked,R.TORINO,,103,1 / 105,1 / 95,3 / 91,9 / 91,3 / 100,3 / 100,4 / ,Waiting
2023/05/28 10:43,95600000,5202,Locked,*Radio2,RAI RADIOTELEVISIONE ITALIANA NUOVI SERVIZI RDS  EON  TMC,91,9 / 92,4 / 94,5 / 96,8 / 92,9 / 9
2023/05/28 10:45,96400000,5348,Locked,*DISCO*,OLE                  22              LA,Waiting,Waiting
2023/05/28 10:46,95600000,5202,Locked,*Radio2,RAI RADIOTELEVISIONE ITALIANA NUOVI SERVIZI RDS  EON  TMC,,Waiting
```

However, the exposure part of PS data, which does not always appear complete in the log, could be more improved..

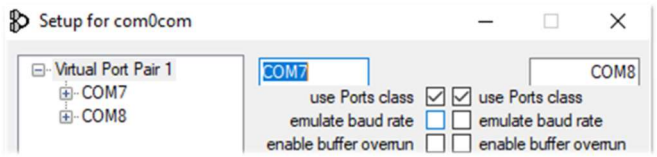
A video can be viewed here: **https://twitter.com/i/status/1662193387615793155**

SerialController

As an alternative to the CalicoCat plugin I point out this additional software "SerialController" which allows SDR# to control via virtual COM ports a set of commands proper to the rtx Kenwood TS-50.



Supported commands:
IF - sets frequency and mode
FA - sets the frequency
MD - sets the mode (AM, CW, FM, USB, LSB)
Serial port parameters: **9600 baud rate, 8 data bits, 1 stop bit, no parity.**
Installation is very minimal: copy the file "SDRSharp.SerialController.dll" to the Plugins directory, then start SDR# and flag the "Enable" checkbox.
"SerialController" will use two virtual serial ports previously created by software such as "com0com". The actual numbers that will be assigned to the COM ports depend on the configuration of your own system (in my case COM7 and COM8). I therefore selected COM7 in SDR# and COM8 in the other software with which I wanted to interface. Since CAT is a bidirectional protocol, the changes made in SDR# will be immediately sent to the other software and vice versa, and in the body of the plugin you will be able to see as you go along the execution of various commands: for example, changing the frequency of the VFO or changing the emission mode.



Freeware downloadable: ***https://github.com/UzixLS/sdrsharp-catcontroller***

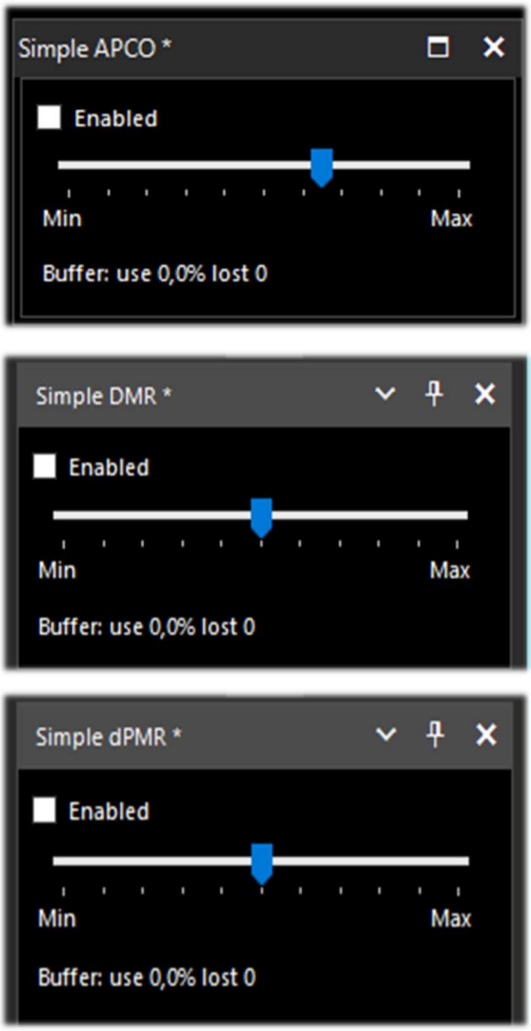
Simple APCO / DMR / dPMR v1.1.1.0

Digital fans and radioamateurs will find these simple plugins, which are fully integrated with SDRsharp, very practical and immediate.

As the title says, they are “simple”, perhaps even too simple, with no indication of the various information that the DMR can carry, such as Colour Code, talkgroup, network type, etc., etc., but for this very reason they are fast and ultra-practical!

You extract the two DLLs into the appropriate Plugins directory and running SDR# it only remains to enable the plugin (in the appropriate check in the top left corner) and possibly adjust the volume slider.

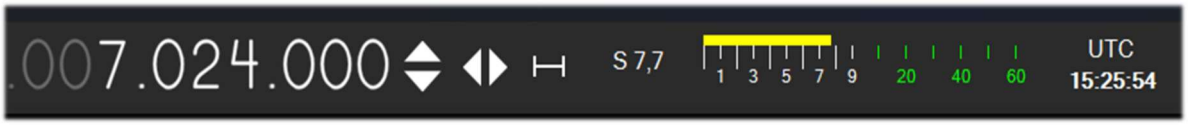
As soon as one of these digital broadcasts passes through you will hear directly through the audio.

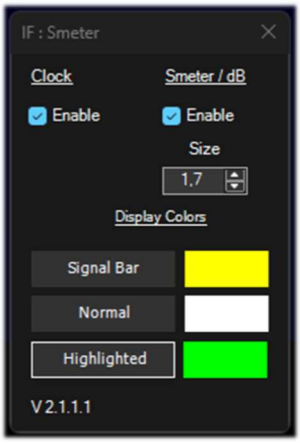


S-Meter v2.1.1.1

The friend Gabriele M. recently informed me about the update of his personal plugin, which was born out of the blue to make up for the lack of an "old-fashioned" S-Meter that is not too obtrusive. The latest release is an updated version especially in the background color that now follows the SDR# set.

If enabled, it will be automatically displayed on the right side of the VFO.

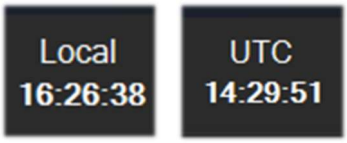




The plugin has this panel for customization of the various control settings of the S-Meter, the built-in Clock, the Size for the size of its width (from 1.0 to 3.0) and customization of the signal bar colors and its scale.

Then click on the S-Meter to switch between S-unit or dBm.

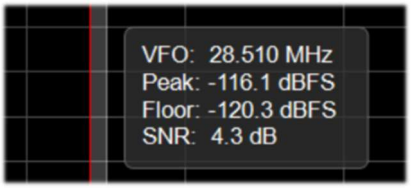
Clicking on the clock instead switches between local time and UTC.



The lines that will appear in the SDRSharp.config file will be as follows:

```
<add key="SmeterUnits" value="True" />
<add key="SmeterOffset" value="0" />
<add key="SmeterVisible" value="True" />
<add key="SmeterScale" value="2.3" />
<add key="SmeterUTC" value="True" />
<add key="SmeterClockVisible" value="True" />
```

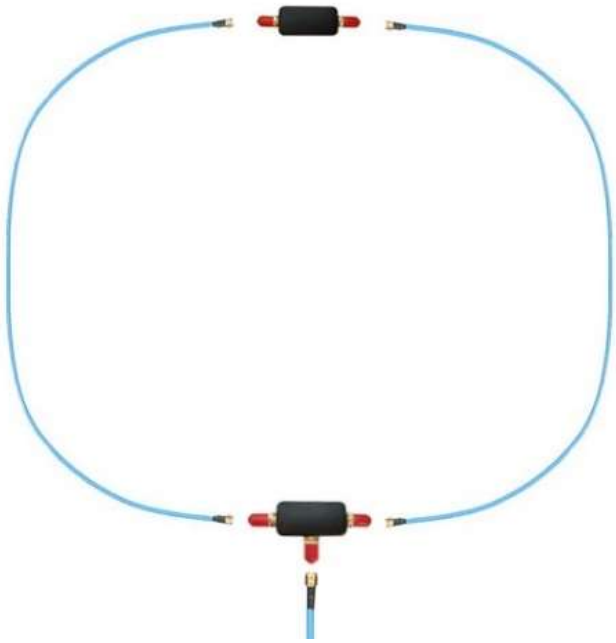
The S-Meter uses the value acquired from the SDR# API and found in the variable: SDRSharp.Common.ISharpControl.VisualPeak. It is the same value that is read by hovering the mouse within the bandwidth.



Antenna YouLoop

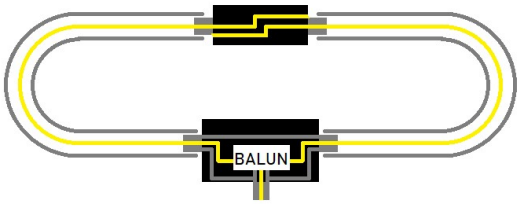
A new magnetic loop concept

The success of the Airspy HF+ series brought a lot of feedback from users noting that many problems were related to receiver antennas that were ineffective, too sensitive to surrounding noise, had excessive gain, and lacked the necessary linearity. This led to the idea of designing a new 'Noise-Cancelling Passive Loop' (NCPL) to solve the noise problem and take advantage of the low noise performance of Airspy receivers. The new loop antenna was named 'YouLoop' from its designer Youssef.



Architecture

YouLoop is a generalisation of the Möbius loop in which a two-turn balanced coaxial cable is used as the centre of a multi-turn loop. This construction is electrically balanced for large wavelengths, i.e. when Lambda is very large compared to the size of the antenna. This helps to cancel the electrical noise at the lower bands, exactly where most of the electrical noise energy is concentrated. To preserve this electrical balance from being disturbed by the transmission line, a miniature low-loss BALUN is used underneath the loop.



Performance in VHF too

Another interesting aspect of this two-wire coaxial construction is its response in the VHF range. The same HF magnetic loop can therefore be used in the FMW, aviation and 2m amateur radio bands with a slight change in its basic principle: it is now a folded dipole. In fact, the feed point of the folded dipole is at the top and the arms of the dipole form the ground of the coaxial. At the feed point, the signal is routed from the two sections of coaxial cable to the wideband, low-loss BALUN.

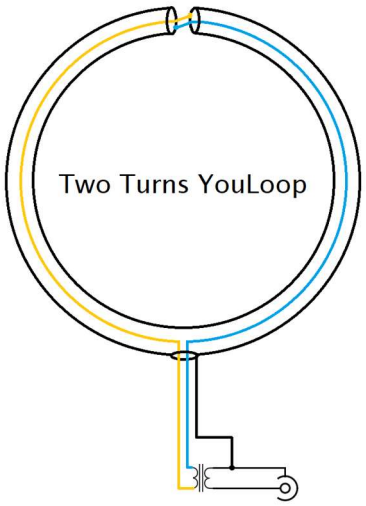
Equipment list

Once the best quality components have been chosen (beware of clones and imitations!!) and the PCB has been pre-assembled with the BALUN, the only thing to do on receipt of the loop is to connect the blue cables marked SMA in just a few seconds *and maybe attach the antenna itself to a rigid Hula-Hoop, like the ones used in children's games, to make it more manageable and steerable on some temporary structure (e.g. a photo tripod).*

For semi-permanent outdoor installations, it is recommended that the "Balun-T" and "Phase Inverter" elements be sealed very well with sealing tape.

For optimum performance it is recommended to use coaxial cables suitable for the purpose.
Any phase or amplitude mismatch will result in sub-optimal performance. Those proposed meet the criteria for optimum performance:

- 2 arms (1m) in RG402 18 GHz coax cable, with male SMA connectors
- 1 transmission line (2m) in RG402 18 GHz cable, with male SMA connectors
- Phase inverter (upper part of the loop)
- Wideband low-loss T-shaped BALUN (bottom of loop)

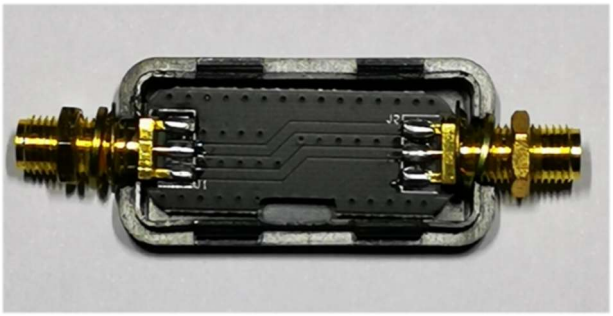


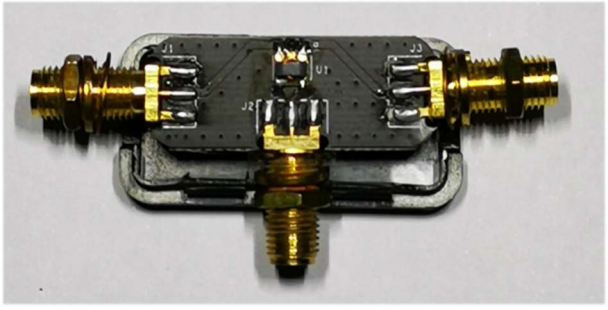
Technical specifications:
HF: from 10 kHz to 30 MHz
VHF: up to 300 MHz
Maximum Power: 250 mW
Passive design and no tuning/synchronisation required
Low-loss, wide-band BALUN (0.28 dB loss)

Compatibility:
Airspy HF+ Discovery (Recommended)
Airspy HF+ Dual Port ((with R3 short-circuited)
Other SDR with MDS <= -140 dBm

It is very likely that your own third party receiver is not sensitive enough to work properly with the YouLoop...
Not using an Airspy HF+ Discovery, some people have even tried, without much success, to make preamplifiers to compensate for the lack of sensitivity and/or dynamic range required by substandard receivers.

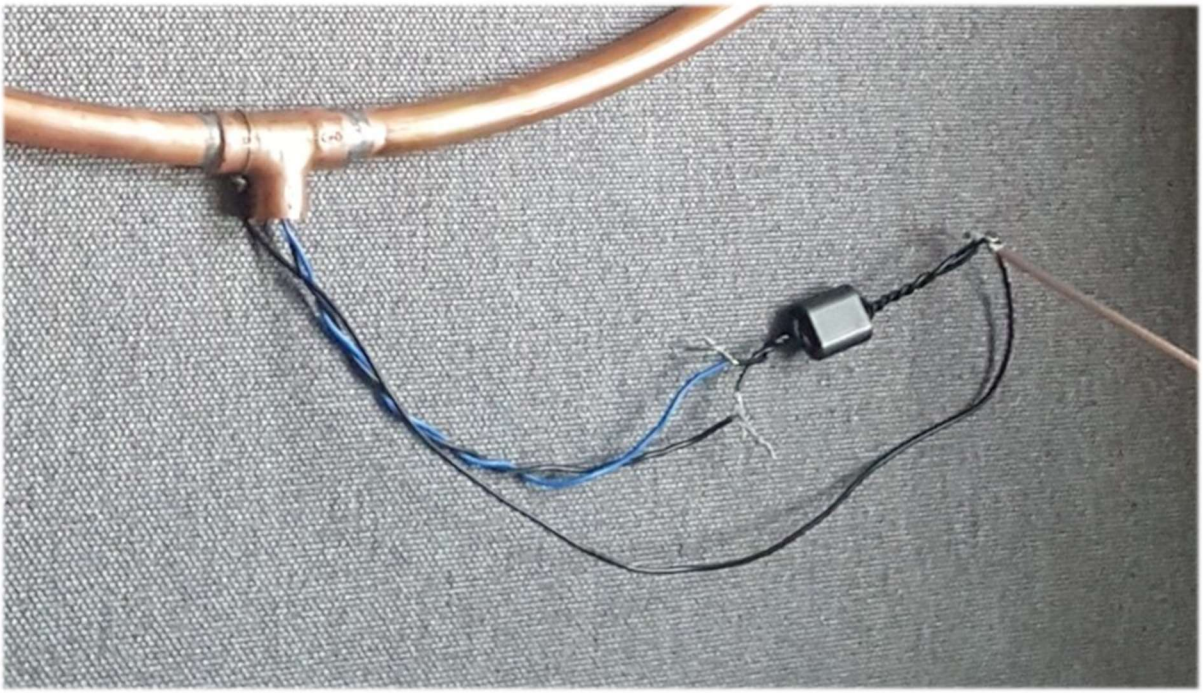
But what's inside?
A friend got me these images....





A possible implementation proposed by Youssef, could be the more rigid one made with copper pipe open at the top and soldered at the bottom in a T fitting.

Here are some construction details:



SpyVerter R2

Before the advent of Airspy HF+ Discovery/Dual Port you could extend your coverage below 30 MHz with the **SpyVerter Upconverter** in combination with your devices...

It is a low-loss, high-dynamic-range up-converter based on a switched-mode design, the same type that is used in high-end HF rx which, due to its frequency stability and sensitivity, can compete with analogue designs at a very affordable cost.
SpyVerter R2 is based on the successful SpyVerter architecture and enhances the key points of high performance HF reception.
The architecture is based on a dual balanced switched-mode mixer that transposes the entire HF spectrum in the VHF band between 120 MHz and 180 MHz.
An embedded microcontroller provides both PLL programming (Si5351C) and VCTCXO voltage control via its built-in DAC.

The substantial difference between SpyVerter R0 and R2 is the high-speed PLL instead of the TCXO.

Technical specifications:
RF Input 1kHz to 60 MHz
IF Frequency 120 MHz – Positive Image
Technology: Switched Double Balanced Mixer
Total Conversion Loss + Filtering: 8 dB typ.
35 dBm IIP3
LO leakage: -42dBm typ. (12 dB lower than the original SpyVerter)
Phase noise at 10kHz separation: -122 dBc/Hz
RF Filtering: Low Pass Filter with corner at 65 MHz – 75dB ultimate rejection
IF Filtering: Band Pass Filter with corners at 120 MHz and 180 MHz – 75dB ultimate rejection
Max RF power: +10 dBm
Return Loss: -10 dB
Bias-tee voltage: 4.2v to 5.5v
Internal 10 MHz Reference Clock input
Current consumption: < 100 mA

Compatibility:
Airspy R2
Airspy Mini
HackRF One
RTL-SDR

The SpyVerter offers HF coverage starting near DC and up to 35 MHz where it overlaps with Airspy's VHF-L.

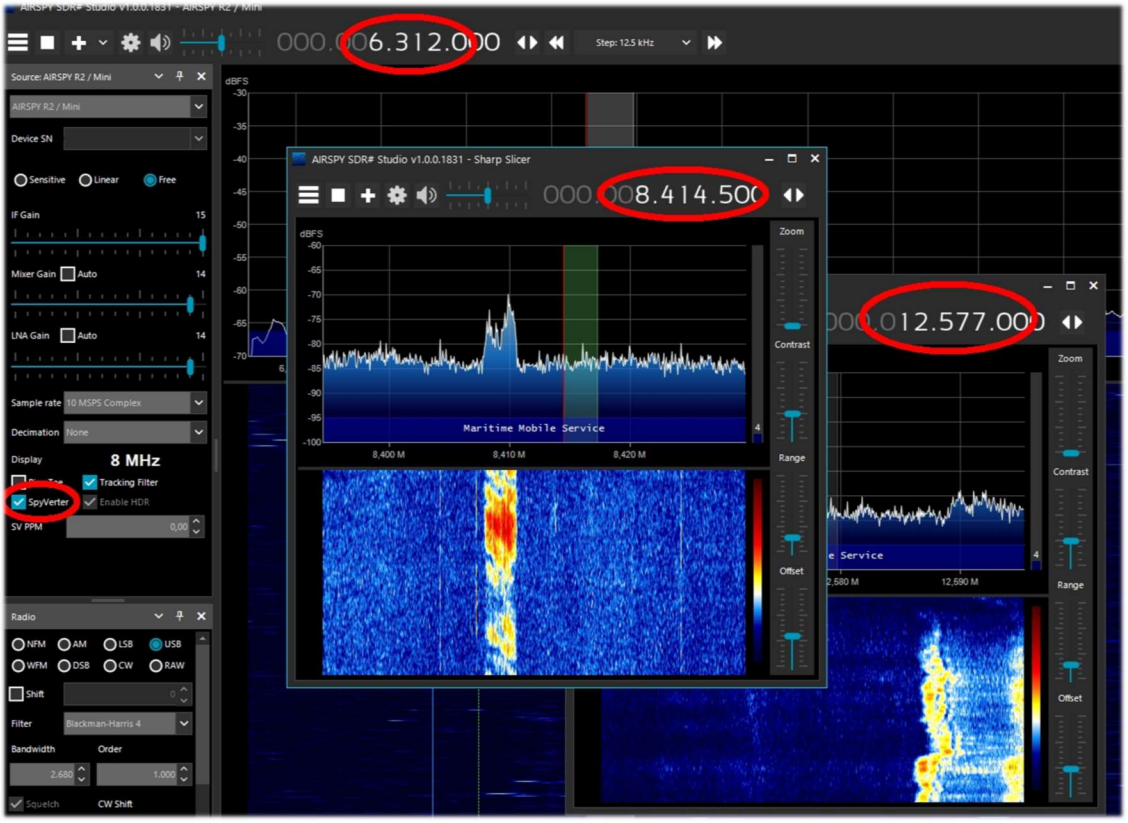
The default software settings allow the Airspy to power the SpyVerter unit via the "bias-tee" feature, so no extra power is needed. Simply connect the SpyVerter's IF output to the Airspy's RF input via the supplied barrel adapter.

It is recommended to use the "Linear gain" mode in HF.

An idea I had recently was to use the SpyVerter in combination with an Airspy R2 for simultaneous HF decoding of ALE and GMDSS signals thanks to the brand new multi-channel decoders by Chris Smolinki's Black Cat (W3HFU)...
Let's see some steps together.

I'll give more information on Black Cat's ALE and GMDSS decoders in the "Listening Recipes" chapter later, but this is the general principle.
Exploiting the SpyVerter and the combined use of the Slice *(see "New Slice" in the chapter "Main settings and controls")* is like having several independent receivers (but always within the default bandwidth) to which you can feed several signals to monitor and decode!
For example, for HF world nets in ALE or in the GMDSS system there are many frequencies to keep an eye on and not all of them are active at the same time or can only be received at certain times of the day...
If you could have a decoder active on each frequency you could optimise simultaneous reception and automatically catalogue a large number of logs.
All this is possible by using some new multi-channel decoders designed specifically for this purpose in combination with multiple virtual audio channels, with the only limitation of having a sufficiently powerful computer / CPU ...

Obviously you can start with two/three frequencies and its VAC properly configured on Line 1/2/3.
In this screenshot on an Airspy R2 tuned to 6.312 kHz of the GMDSS world system I opened two new Slice at 8.414,5 kHz and 12.577 kHz



Keep in mind, however, that the SpyVerter is designed to be a broadband HF receiver.
This may be useful for some scenarios, but it may lack the dynamic range for high performance use when receiving weak signals or with strong blocks in the vicinity. The limitation does not come from the SpyVerter as the weight is delegated to the VHF receiver output. Slice is the way to have stand-alone receivers with full functionality from the same front-end.

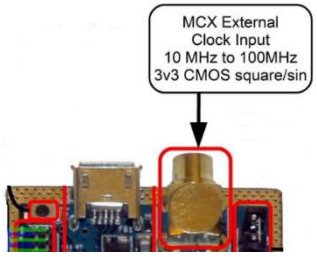
If the SpyVerter is used remotely, the following parameters must be used in the **spyserver.config** file

```
# Initial Center Frequency
#
initial_frequency = 7100000

# Minimum Tunable Frequency
# Comment if using the device default
#
minimum_frequency = 0

# Maximum Tunable Frequency
# Comment if using the device default
#
maximum_frequency = 35000000

# Converter Offset
# Set to -120000000 to enable the SpyVerter offset
converter_offset = -120000000

# Bias-Tee
# For AirspyOne only – Useful for LNA's and SpyVerter
enable_bias_tee = 1
```

External clock

The internal frequency reference, used for the tuner, is a temperature-compensated crystal (TCXO), with an accuracy of a couple of parts per million and a drift of a few tenths of PPM, making it perfect in daily use.

However, for very high precision work there is the possibility of matching an external clock. In fact, the devices can receive an externally generated high-stability signal, usually 10 MHz, for example from a rubidium oscillator, a GPS source, or even from an ordinary good oscillator.



For the Airspy R2, next to the USB socket, there is a small MCX connector for the reference input. It is necessary to turn off (disconnect the USB) and then reconnect the reference input so that the internal hardware can detect it. In fact, if an external clock signal is applied to the Airspy R2 before it is powered on, the Airspy will default to the external clock.

GPSDOs (Precision Frequency Reference) from "Leo Bodnar Electronics" are a good choice for this purpose.

OCXO Oven-controlled XO (crystal oscillators) can be found on the internet that offer greater stability than standard clock oscillators or the possible over-temperature variations of TCXOs.

Notch filter 88-108

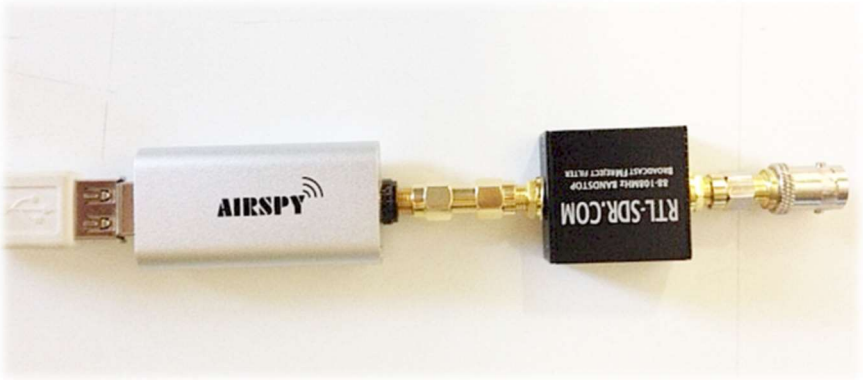
Those who live in cities or in the vicinity of strong/very strong signals from WFM broadcast stations will need to make or buy a good notch filter to attenuate the presence of these signals, which can also desensitise other portions of the spectrum not included in the operating range in question (e.g. the adjacent aeronautical band).

There are different forms and performances (with attenuation, expressed in dB, also very high). In the best ones, the insertion loss outside the operating band and up to 500 MHz is practically absent, while it is very low for higher frequencies.

In our case, it is preferable to choose the newer ones with an SMA connection so as not to put too much mechanical strain on the older, heavier ones still with BNC or PL connectors.

This is a typical usage configuration...

If you start using them, afterwards you won't be able to do without them!!



Variable notch filter

Another curious and unique accessory that I happened to use again recently was this variable notch from SSE UK (initials NF.96XI-1) purchased many years ago..

Compared with the previous 88-108 notch family, this one has the main feature of being able to be seamlessly tuned in the 80-190 MHz range, giving the possibility to attenuate all those analog/digital signals of civil services operating in VHF as well.

These the specifications:
- Insertion Loss: <1dB
- Notch Attenuation: -40dB
- Attenuation Below 1MHz: -60dB
- Receive Range Approx: 2000 MHz
- Impedance: 50 Ohms

137 MHz filter

Those with specific needs then there is the **"Nooelec SAWbird+ NOAA Barebones".**

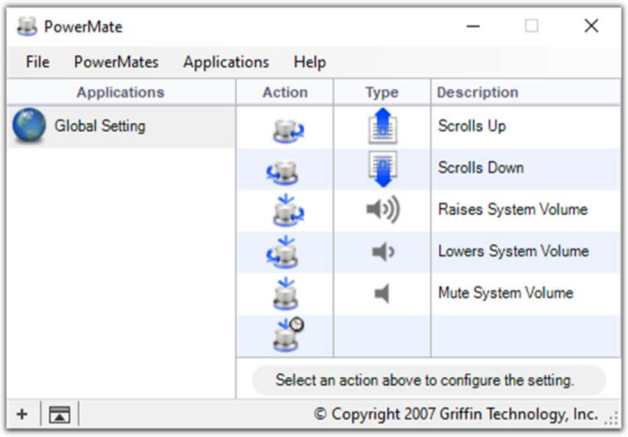
This stand-alone module with integrated SAW filter is designed to capture the beautiful weather images available from NOAA satellites on 137 MHz.
It has a very high attenuation outside the 5 MHz bandpass and a minimum gain of 30 dB. The nominal current draw is 180mA.
The module is fully EMI shielded and allows three different power supply options. The recommended one is through the SMA port with bias-tee capability if available from the SDR. Alternatively, external power options can be used through the microUSB port or the power supply input (3.3V to 5.5V DC).

https://www.nooelec.com/store/sawbird-plus-noaa.html

External controls

Those who need to have an external control for fine tuning the VFO faster than can be done through the mouse can think of matching an "external tuning wheel" (or VFO tuning knob or SDR tuning wheel)...

I had just one, the **"Griffin PowerMate"** lying unused in an old drawer: it works excellently with Airspy even in Windows 10 and tuning is definitely smoother and easier. Its customizable programming allows for volume change and "mute," for example. I had used it before with other SDR receivers. This is its interface (needs software and related drivers). As soon as it is connected to the USB, the silicone base lights up with a beautiful blue light that can be customized.



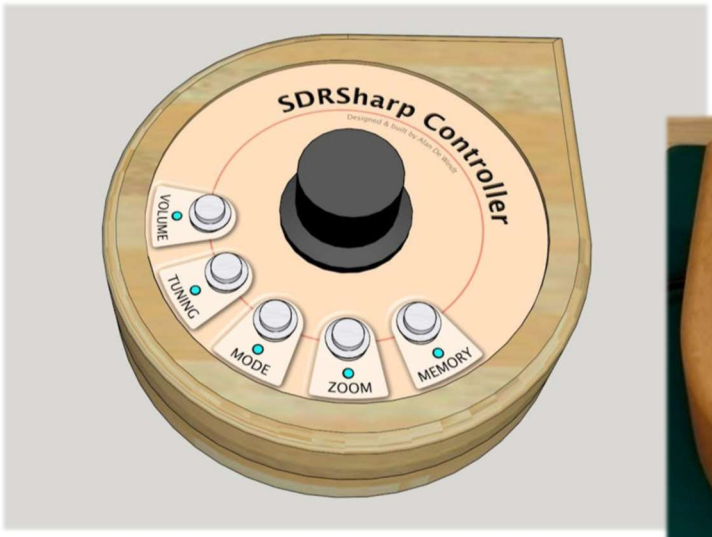
Unfortunately, I understand that this accessory has not been in production for some time, so the lucky ones may perhaps only be able to find it on the second-hand market.

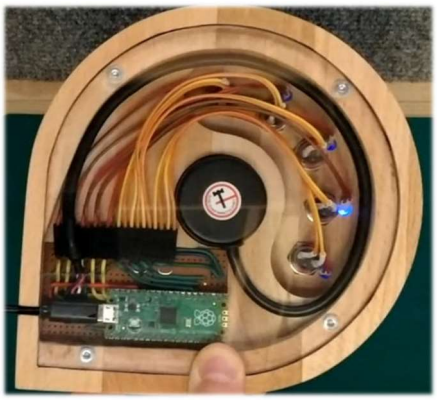
On the market, there are other external controls, some very expensive that I have not had a chance to try, nor do I know friends who use them. However the friend “Pierluigi” points me to an inexpensive "Tune kit" bought on the net, which requires no software/driver (it is a mouse simulation) and in SDR# allows:

- tuning management, if the "focus" is on the RF Spectrum, turning the wheel will change the frequency according to the left/right rotation (and of course the chosen step),
- if the "focus" is on one of the VFO numbers, turning the wheel increases or decreases these by one unit,
- if the "focus" is on one of the other sliders (e.g. Volume, Zoom, Contrast, Range, Offset or other) turning the wheel will increase or decrease these.

For those who enjoy easy DIY, the friend Ladislav OK1UNL points me to these interesting and informative links:
https://www.qsl.net/z33t/sdr_frequency_controller_eng.html
Here is a solution with programmable buttons, including "button mapping" with Pluralinput software (also for Win10).
https://19max63.wordpress.com/2016/05/15/tuning-knob-for-sdr/
https://pluralinput.com/index-old.html
Multi-pointer X for Linux:
https://wiki.archlinux.org/title/Multi-pointer_X

Another interesting project, recently published, is the very professional one by Alan De Windt that allows SDR# to be controlled with real physical controls for a much more natural experience...





There are so many features...

Each button is marked by its main function. When you want to adjust the volume, for example, you press the Volume button and then turn the center rotary control clockwise and counterclockwise to adjust. By pressing most of the buttons a second time, you can make a second setting.

Here are the primary and secondary functions of each button:

Volume - Primary: adjusts the audio volume/gain. Secondary: adjusts the squelch threshold

Tuning - Primary: adjusts the tuning / frequency. Secondary: adjusts the tuning pitch

Mode - Primary: changes the type of filter/demodulation mode (FM, AM, LSB, etc.). Secondary: adjusts the bandwidth of the filter.

Zoom - Primary: Adjusts the zoom of the FFT. No secondary function

Memory - Short pressure and release within one second: memory recall. Long press: memory setting

For more details: **https://www.hackster.io/AlanDeWindt/sdrsharp-controller-83baa8**

Dipole antenna kit RTL-SDR

Some dear friends suggested to me this portable antenna, indeed the "Multipurpose Dipole Antenna Kit" so defined on the official website: www.rtl-sdr.com/store

The kit can be very good for many outdoor occasions, during a trip given its minimal bulk and weight or sporadic testing. In fact, it is designed for portable and temporary outdoor use (however, not to be placed outside with inclement weather!). NOTE: this antenna is designed for reception only, it is NOT a TX antenna.

It includes:

1 dipole antenna base with 60 cm of RG174
2 telescopic antennas from 23 cm up to 1 m
2 telescopic antennas from 5 cm up to 13 cm
1 three meters RG174 extension cable
1 flexible tripod mount
1 suction cup mount

The connectors are all SMA.

Some suggestions for quick installation thanks to the provision of multiple accessories that allow the antenna to be taken and outside and in a higher position conducive to receiving VHF-UHF frequencies (up to the L-band at 1.5 GHz):

- suction cup mount for windows or car windows
- V-dipole orientation for satellite reception
- attachment to a desk, outdoor pole, tree branch or the door/window of the house with its flexible, articulated and rubberized tripod.

Window Suction Cup Mount

V-Dipole Satellite Orientation

Flex Tripod Mount on Table

Flex-Tripod Mount to Pole

Flex Tripod Mount to Tree

Flex Tripod Mount to Door

The RG174 cable of the antenna base is decoupled from the elements with a ferrite inductance to prevent the feed line from interfering with the radiation pattern of the dipole. The dipole base mounts on a standard 1/4-inch camera screw, so it can be mounted on a variety of mounts already available...

Aside from the length to be used for telescopic antennas, I often see people using them in strange and incorrect orientations for the polarization of transmitted signals. ***In almost all cases they should be used in a vertical position (as well shown in the previous pictures, except when using it for satellite reception in a kind of horizontal V).***

But what is the right dipole length? It depends on the frequency we want to tune in...
A simple formula comes to our aid:

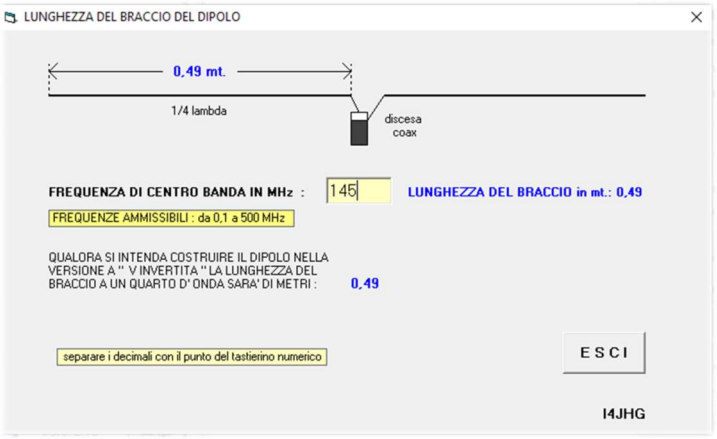
***V** speed of light / **F** frequency (kHz) = λ wavelength/2 (for single arm length):*

For convenience, I made use of Rainiero I4JHG's RADIOUTILITARY software, downloadable freeware from:
www.radioamatorimonopoli.it/files/radioutilitario.exe

Select: Antenne / Dipoli / Dipoli ½ lambda

Enter the frequency in MHz to get the precise length to extend the single telescopic element.

In this screenshot for the 145 MHz frequency we get the dipole arm length of 49 cm (inch 19.29).



I recommend visiting the following insights link to discover other valuable information such as VSWR diagrams: **https://www.rtl-sdr.com/using-our-new-dipole-antenna-kit/comment-page-1/**

In this table we can see at what frequency (in MHz) the two types of antenna (short and long) resonate with the same number of extracted elements:

Antenna	sections	cm (*)	inch (*)	MHz
short	1	7,1	2.79	1055
short	2	10,1	3.97	742
short	3	12,8	5.03	585,5
short	4	15,2	5.98	493
long	1	24,9	9.80	301
long	2	44,2	17.40	169,6
long	3	63,2	24.88	118,6
long	4	82,3	32.40	91
long	5	101,1	39.80	74,1

(*) including about 2 cm. (inch 0.78) of inner base

Instead, in this other one we can find the band center of some services and relative antenna length:

MHz	band	cm	inch
85	FM 76-95 Japan	84	33.1
98	FM 88-108	72	28.3
145.7	OM 2 meter band	49	19.3
157	Nautical band	45	17,7
225.6	DAB ch.12B	31	12.2
431	OM 70 cm band	16	6.3
560	DVBT ch.32	13	5.1

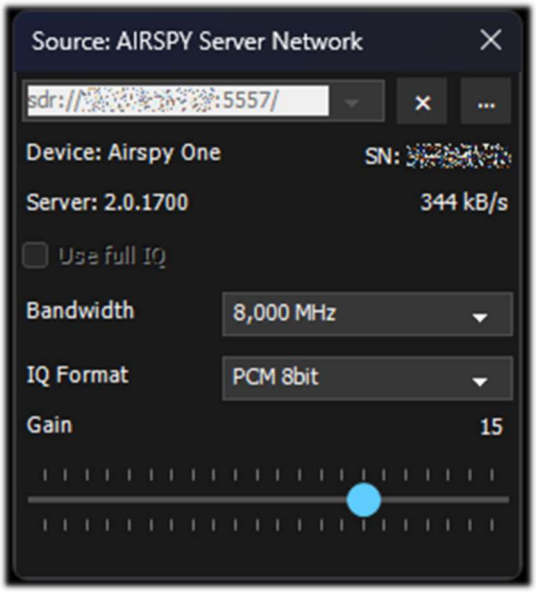
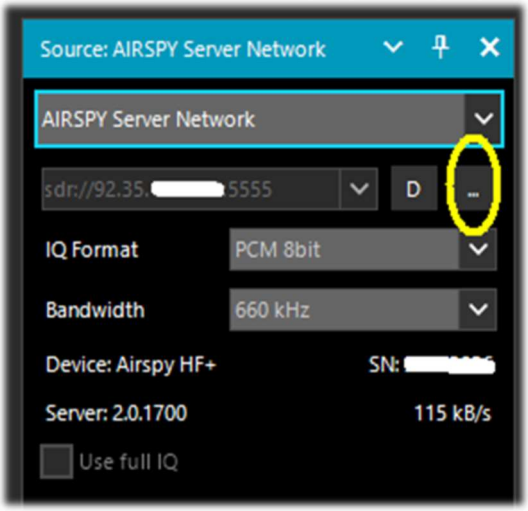
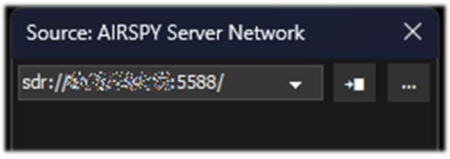
Airspy Server Network

From v1553 you can use or create your own remote SDR# server via the "SPYSERVER.EXE" tool. This allows you to connect via the Internet to many Airspy or RTL-SDR "clients" scattered around the world or create your own personal local network with your own remote SDR perhaps in the attic and connected wirelessly to your computer at home.
When only one user is connected full control (frequency, RF gain) is allowed while when there are multiple clients connected the frequency and RF gain are blocked.

These are the possible configurations in some environments and operating systems:

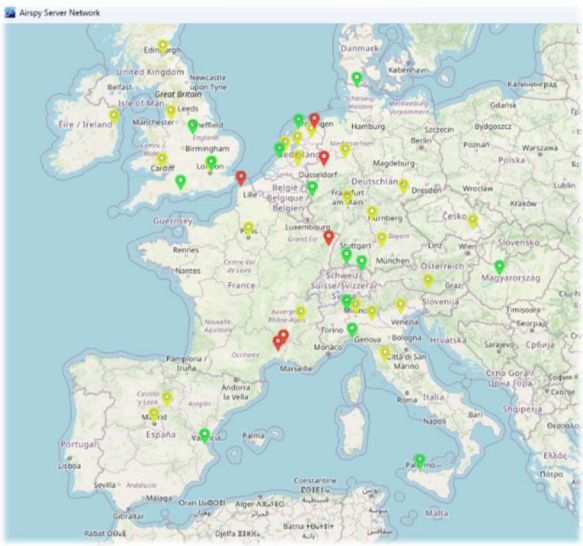
---------- Using the client on its own computer ----------

Selezionare la voce “AIRSPY Server Network” presente nel pannello Source.

<table>
<tr><th>from v1920</th><th>fino alla v1919</th></tr>
<tr><td> 
</td><td> 
</td></tr>
<tr><td> 

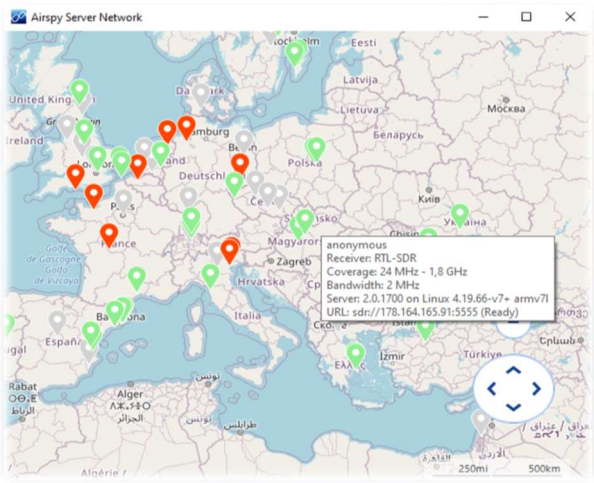
Click on  icon to open the webmap with the various servers. By hovering the mouse over the various icons a box appears with the technical characteristics: user name, receiver type, coverage (in HF, V/UHF or full), bandwidth, server type and URL. To connect click on the green-colored icons.</td><td>Clicking the yellow highlighted button of the "Browse Spy Server Network" will open a webmap where you can see the various servers. Since v1809 the webmap had been revamped with the Telerik RadMap.
Mousing over the various icons will open a box with all the technical characteristics of the remote server highlighted.
To connect you click on the green icon.</td></tr>
</table>

To access the remote URL already stored press the icon 



Active Spyservers are highlighted with a green icon while those with a yellow icon are currently unreachable.

To end the remote session, press the "D" (Disconnect) button.

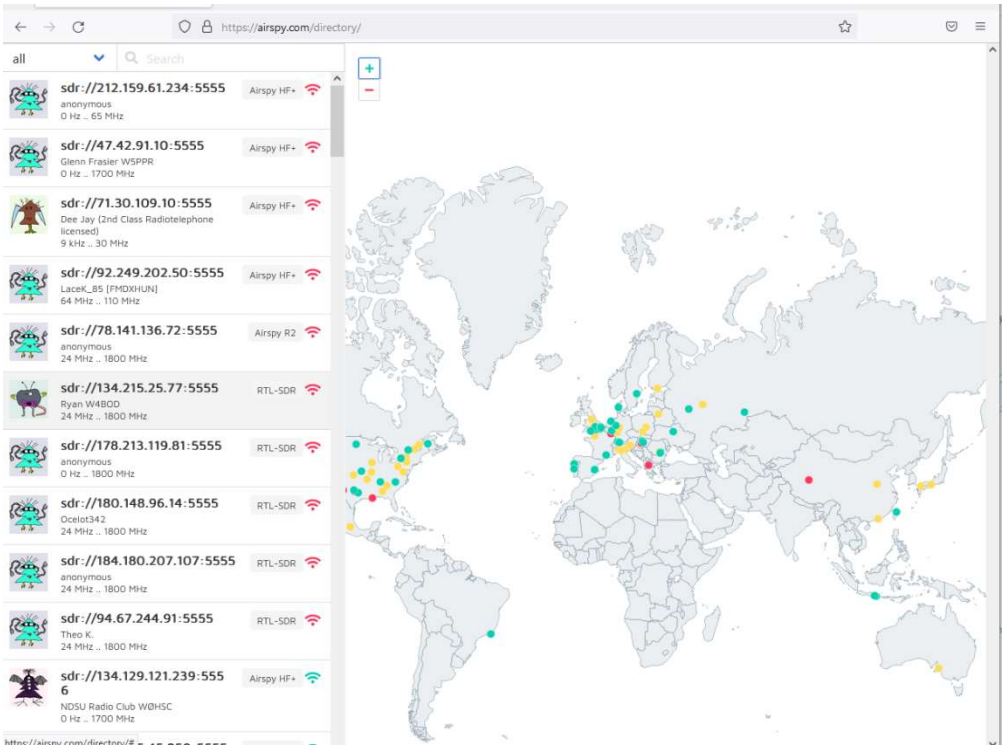


Active Spyservers are highlighted with a green icon.

Depending on the source device, adjustments can then be made to the IQ format, bandwidth and gain.

The "Use full IQ" option allows streaming of the full spectrum *as long as you have sufficient network bandwidth and a high-speed connection.*

A similar map can also be reached from the site under "Online SDR" or directly from this URL: *https://Airspy.com/directory/*



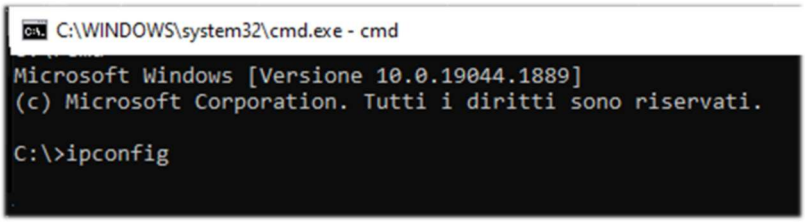
For proper viewing I suggest using only the FIREFOX browser.

---------- Server Network in Windows ----------

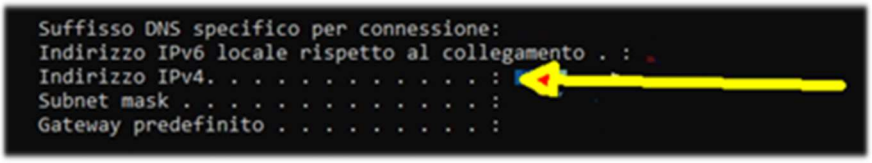
On the Airspy site, under "SPY Server - SDR Server for Windows" you should download the following zipped file: *https://Airspy.com/?ddownload=5857*
On my computer I extracted the files to the root directory of SDR# just being careful not to overwrite the newer ones!

Key point is to know your static IP *(not dynamic!!! If in doubt you need to contact your internet provider or use another network)* and verify that the ports are open and not blocked by any router/firewall/antivirus/etc. These are the steps I performed:

1. From the Windows menu, type RUN.
2. Type CMD, followed by Enter to open the command window.
3. Type IPCONFIG, followed by Enter. This will display the list of network cards on the computer with their IP addresses (Ethernet and/or wireless).



4. On my computer the useful number (private IP) is the one that appears next to the ***IPv4 address*** in the format 192.xxx.xxx.xxx (yellow arrow). *All other addresses are not useful to us and should be left out*.



5. You still need to know your public IP address, which you can track by logging on to one of the many online services such as WWW.MYIP.COM On my side it is 128.xxx.xxx.xxx which I will go and note down…
6. Summarizing: 192.xxx.xxx.xxx (private IP)
   128.xxx.xxx.xxx (public IP)
7. Connecting to your router create a rule like the following in PORT MAPPING & FORWARDING to open port 5555 and assign it to your private IP 192.xxx.xxx.xxx

8. At this point of you will have to edit the text file **"spyserver.config"** previously extracted being careful not to change those entries of which you are not sure and remembering that deleting the # character makes the following instruction active.

Here is a short excerpt of the file (in red color the one I modified) for remote use of my AIRSPY HF+ DISCOVERY:

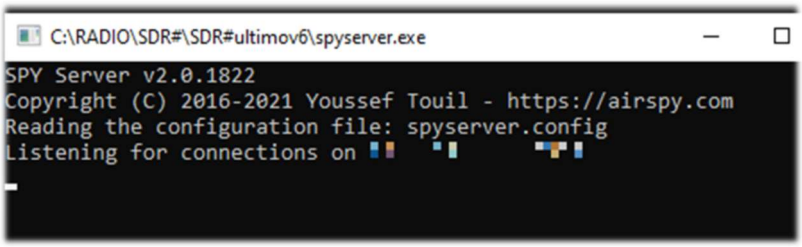
```
# SPY Server Configuration File

# TCP Listener
#
bind_host = 192.xxx.xxx.xxx  (IP privato)
bind_port = 5555

# List Server in Airspy Directory
#
list_in_directory = 1

# Owner Name
# For example: John Doe L8ZEE
owner_name = Paolo

# Antenna Location
# For example: 48.858332, 2.294560
antenna_location = 45.xxx, 7.xxx

# Device Type
# Possible Values:
#   AirspyOne (R0, R2, Mini)
#   AirspyHF+
#   RTL-SDR
#   Auto (Scans for the first available device)
#
device_type = AirspyHF+

# Device Serial Number as 64bit Hex
# For example: 0xDD52D95C904534AD
# A value of 0 will acquire the first available device.
#
device_serial = 0

# Device Sample Rate
# Possible Values:
#   Airspy R0, R2 : 10000000 or 2500000
#   Airspy Mini   :  6000000 or 3000000
#   Airspy HF+    :   768000
#   RTL-SDR       :   500000 to 3200000
#
device_sample_rate = 768000

# Initial Gain
#
initial_gain = 10
```

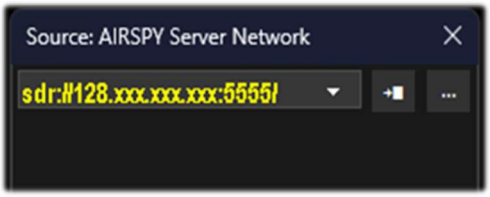
the value 1 makes our server visible on the map

will position your receiver in the Spyserver map

Gain level (0 / 21) (Linearity)

Once the file is saved we are ready to run **spyserver.exe**. On the screen this indication will appear: *"Listening for connections on 192.xxx.xxx.xxx:5555"*



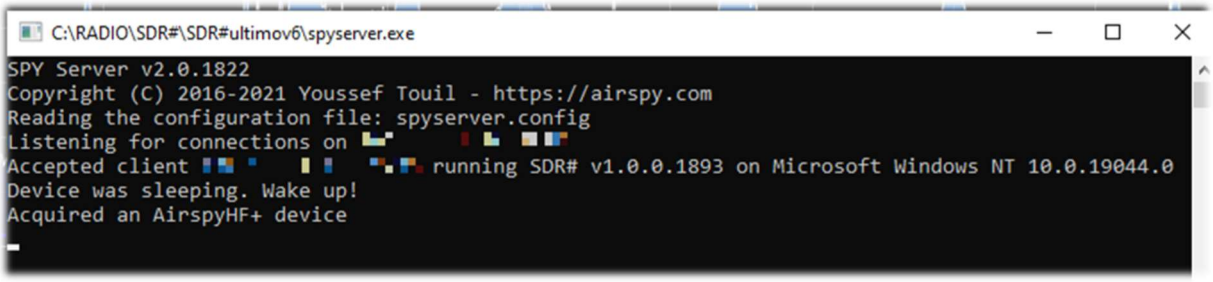


From the Source panel in SDR# we are ready to enter our public IP address in the following format:
sdr://128.xxx.xxx.xxx:5555/

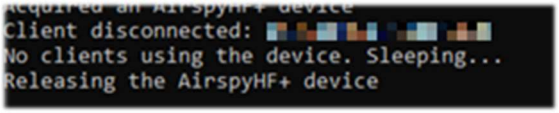
and then press the button 

You can test right away if you can connect to your client...

The previous panel will update with the following information about the client:
"Accepted client 128.xxx.xxx.xxx:xxxxx running SDR#...
Device was sleeping. Wake up!
Acquired an AirspyHF+ device"



To close the session you will have to press the  button and the screen will indicate:
"Client disconnected: 128.xxx.xxx.xxx:xxxxx
No clients using the device. Sleeping…
Releasing the AirspyHF+ device"



And you can now also turn off the **spyserver.exe**

Going back one step to the "**spyserver.config**" file, in case we want to make our server visible to third parties on the world map (by entering the value 1 in the above script) we can provide in the script additional information such as our name, QTH, device type, antenna location which allows to correctly place the marker on the map (which otherwise highlights the one of our own provider), tunable frequencies, etc. etc.

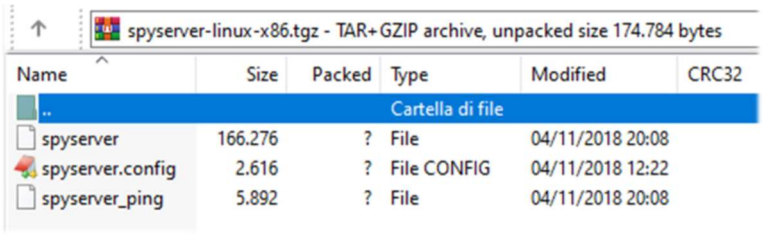
In the image, the larger green dot highlights the location of the remote Network Server while the smaller blue dot highlights the location of the client that is connecting.

On the Spy Server Client side, instead of using Zoom on a reduced portion of the spectrum, less bandwidth can be used at the server thus taking advantage of the better resolution of the FFT. The zoom bar remains for convenience.

---------- Server Network in Linux ----------

These are the main steps to be performed:

1. In the Linux device used as the server, open a terminal window
2. Install the RTL-SDR and librtlsdr drivers:
   ```
   sudo apt install rtl-sdr librtlsdr-dev
   ```
3. Create a folder named spyserver and place inside:
   ```
   mkdir spyserver
   cd spyserver
   ```
4. From the Airspy site, under **"SPY Server - SDR Server for Linux x86"** you can download and extract the Spyserver version for 32bit Intel/AMD CPUs:
   ```
   wget -O spyserver.tgz http://Airspy.com/?ddownload=4308
   tar xvzf spyserver.tgz
   ```
   Or under **"SPY Server - SDR Server for Linux x86_64"** you can download and extract the version of Spyserver for 64bit Intel/AMD CPUs:
   ```
   wget -O spyserver.tgz http://Airspy.com/?ddownload=4262
   tar xvzf spyserver.tgz
   ```
5. Find the IP address of the device using the ifconfig command and note down the number.
6. Use a text editor such as Nano to edit the **"spyserver.config"** file in the downloaded package with what was already mentioned in the previous section for Windows:
   ```
   nano spyserver.config
   ```
   Save the edited file.
7. Running the SpyServer:
   ```
   ./spyserver
   ```



---------- Server Network with Raspberry Pi ----------

On the Airspy site, under **"SPY Server – SDR Server for Linux ARMHF"** you can download the zipped file useful for Raspberry PIs:

https://Airspy.com/?ddownload=4247

While under **"SPY Server - SDR Server for Linux ARM64"** the one for the Raspberry Pi4:

https://Airspy.com/?ddownload=5795

For specific instructions, please refer to the appropriate chapter "Raspberry Pi 3&4" below.

Let us pause a moment longer to better understand what the SpyServer technically does.

*It is basically a TCP server with the ability to create narrowband IQ files after appropriate slicing (or slicing). This means that you get a given bandwidth X from the hardware to the SpyServer which slices 0.1 * X and sends only that part after a good amount of computation! What you get in the end is not the entire spectrum, but rather a narrowband IQ representation of the signal you are listening to.*

For convenience, a low-resolution FFT is also sent for display. Slicing always takes place in the SpyServer. All plugins that require the IF signal still work with this model, giving the false impression that the operation is local or the other "even more false" impression that the server is transmitting all the IQ data but this is not true.

Instead, you are transmitting the minimum amount of data required to make things work properly, unless you ask the server to transmit the data in "Full IQ" mode. There is a setting in the server to set the maximum data to send and a timer to prevent any users from "sucking up" your Internet bandwidth.

Now even when you use the server in your local LAN at "Full IQ," you cannot create additional sessions (slices).

This was not implemented for the simple reason that nothing prevents you from using multiple SDR# instances for streaming from the same server, either in "Full IQ" or "Reduced IQ." There are a couple of settings in the configuration file to set "reduced" bandwidth limits when using the SpyServer.

NOTE: The Airspy Server Network can deliver multiple streams at the same time. It is only necessary to change the "maximum_clients" entry in the "spyserver.config" file by setting it to a value greater than 1. In this case, for example, we configure it for 3 clients...

```
# User sessions
#
maximum_clients = 3
```

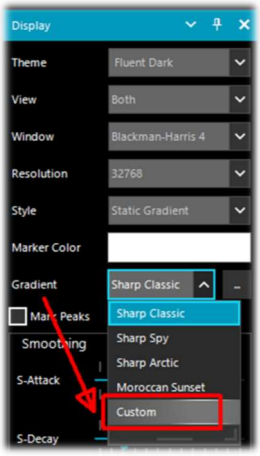
Keep in mind that the center frequency, gain setting, etc. becomes locked as long as two or more clients are connected. In other words, the first connected client must tune the radio so that all bands of interest are visible in the full width of the spectrum. Then more clients can connect and tune wherever they want within that part of the spectrum.

SDRSharp.config file

For the more curious it may be interesting to know the "behind the scenes" of a very important support file, in which all the configurations and settings of SDR# are saved, that is SDRsharp.config, but always pay attention to what you modify taking care to save the file previously...

Let's take a look at some strings already covered in previous specific chapters:

<add key="stepSizes" value="1 Hz,10 Hz,100 Hz,500 Hz,1 kHz,2.5 kHz,3 kHz,5 kHz,6.25 kHz,7.5 kHz,8.3333 kHz,9 kHz,10 kHz,12.5 kHz,15 kHz,20 kHz,25 kHz,30 kHz,50 kHz,100 kHz,150 kHz,200 kHz,250 kHz,300 kHz,350 kHz,400 kHz,450 kHz,500 kHz,1 MHz" />
Listed here are the possible choices of VFO steps between 1 Hz and 1 MHz. If you need to use an unexpected step, simply edit it and enter the new value, e.g. "3.125 kHz".

<add key="waterfall.gradient" value="FF0000,FF0000,FBB346,FFFF00,FFFFFF,7AFEA8, 00A6FF,000091,000050,000000,000000" />
Implementation initially suggested by Youssef for applications of the gradient in High Dynamic Range. *Note the custom key specification:* *"waterfall.customGradient" which is different from the official standard one:* *<add key="waterfall.gradient" value="0" />* *So once we have added the key to our configuration file we need to activate it in Display / Gradient / Custom as highlighted opposite.* 

<add key="core.pluginsDirectory" value="Plugins" />
Directory where all plugins are saved.

<add key="deemphasisTime" value="50" />
To establish the time of deemphasis in FM. In Italy and European countries the time constant is 50 µs while in the USA it is 75 µs. SDR# defaults to 50 and does not have a selector for immediate change, which must be done manually by changing the value of this line.

<add key="DCS.OnlyUseDcsCodesInTable" value="1" />
DCS: an option has been introduced to use only the DCSs that exist in the table, thus reducing their list (see previous plugin "CTCSS & DCS").

<add key="DCS.SwapNormalInvertedDcsCodes" value="True" />
DCS: Introduced additional option to eventually switch the display of DCS codes between "Normal" and "Inverted" (see previous plugin "CTCSS & DCS").

<add key="waterfall.useUtcTimeStamp" value="False" />

*The "Time Markers" option (in Display) enables the display of the **current local** date and **time** on the left side of the waterfall.*
*For who need to display the **time in UTC** format, it will be sufficient to change the value from "False" to "True" in the row highlighted above, which will then display the new format with the final letter Z (zulu)...*

…
<add key="plugin.AudioEqualizer.ParametricGainValues" value="0,-2,-3,-4,-5,-4,-3,-2,0:0,0,0,0,0,0,0,0,0,0:0,0,0,0,0,0,0,0,0,0:0,0,0,0,0,0,0,0,0,0:0,0,0,0,0,0,0,0,0,0" />
<add key="plugin.AudioEqualizer.ParametricPresetNames" value="Music:Flat Preset 1:Flat Preset 2:Flat Preset 3:Flat Preset 4" />
…

Also all configuration data of "Audio Equalizer" plugin by BlackApple62 are saved here automatically in "plugin.AudioEqualizer.ParametricGainValues..." section, here are some lines highlighted.

…
<add key="FilePlayerLastFileName" value="C:\SDR#\092,100 MHz (2021_12_15 1030).wav" />
<add key="FilePlayerLoopEnabled" value="False" />
<add key="FilePlayerShowRealTime" value="False" />
…

Vasili Beliakov's previous FilePlayer added in the <add key="FilePlayer..." section several lines of configuration, here are some highlighted..

…
<add key="core.frequencyDialZoom" value="0.50" />
…

Since v1904 the VFO font size is scalable at will.

In this example, the value is set to 0.50

While in this one it is 0.90

…
<add key="plugin.bandPlan.updateDatabase" value="false" />
…

The bandplan tracks whatever changes you do in the bands and stores them (manually or from Frequency Manager). If you want to disable the Database updates, you can configure it..

```
…
<add key="Airspy.debug" value="0" />
<add key="Airspyhf.debug" value="0" />
…
```

If you set the value to "1," you get the respective extended menu enabled for debug mode and to adjust the calibration in case the displayed frequency does not match the real one (also read the FAQ).

Improving listenings in AM

It is always a bit of a difficult thing when you approach the SDR world from your good old analog receiver that has been used for decades. There are plenty of SDRs on the market, but to achieve performance similar to a high-class analog receiver, people spent a lot of money because the underlying technology was (and still is!) very expensive when aiming for a high level of performance. Those who initially venture into SDR often confuse performance with displayed bandwidth, when in fact it is quite the opposite. The more one is "open" to other signals that one does not need, the worse it is. In addition, some hobbyists unfortunately often confuse their strong local interference with their radio's ability to cope with dynamics in the various bands

The analog radios of yesteryear no longer stand much of a chance in today's noisy environment.

Back then, there were no switching power supplies, LED lighting, DSL Internet, and ordinary household appliances could be brought to market even if they did not pass the very stringent EMC tests. The bar of the past was very low and we never think too much about it, relying perhaps only on memories and feelings of the past.

Today it is necessary to have very high-end receivers with state-of-the-art DSP to get decent listening.

Progress is inexorable as Jim Al-Khalili quotes (see his motto in the "conclusions and quotes"): not all SDRs are the same and not all DSPs are the same. *In SDR# the DSP is implemented in the library "shark.dll" (Sharp Kernels) in a mix of C, C++ and SIMD intrinsics.*

There is a great disparity in the market in terms of performance and also it takes a new "radio and computer literacy" on the part of all of us to discriminate the good from the mediocre/poor.

So what can be done with an Airspy and SDR#, for example, to improve AM listening and achieve very high quality? Certainly a lot, much more than what other SDRs offer...

In order we can use some (or a mix) of the features and basic functions of our SDR# always rendered obviously in freeware mode for all:

* Synchronous demodulation
* Asymmetric IF filtering and Notch.
* Anti-fading
* Broadband noise filtering.
* Narrowband noise cancellation
* Audio noise suppression
* Co-channel suppression
* IF noise reduction
* Audio noise reduction
* Audio filtering
* Audio equalization

Here's the novelty introduced with v1892: ***Super PLL***
Now the "Lock Carrier" is achieved using a special "Super PLL" that has a great resilience to the loss of lock. Basically, when the PLL loses the lock, it starts another process at exactly the same phase where the lock was lost and keeps spinning. This keeps the signal of interest at the same frequency before the loss of lock. When the carrier is available again, the PLL locks and resumes the signal without any phase discontinuities or other glitches. This tool is especially useful for High End DXing with intermittent signals.

To appreciate the features of DSP plugins (NINR, notch, etc.) this video is to be seen:
https://www.youtube.com/watch?v=kLnvwZ2qzrM

Signal decoding and analysis

As mentioned elsewhere in this guide, a very interesting possibility is the study of digital signals and their decoding, using special software and a "virtual audio cable".
This is necessary in order to redirect the audio of SDRSharp (or other SDR programs) towards external decoders for many signals that we can find in HF (examples: MultiPSK, Fldigi, WSJT-X, Morse, Wefax, DReaM [1] etc.,) or in V-UHF (examples: DSD+ [2], APRS, satellites and weather satellites, etc.).

- *DReaM for the Digital Radio Mondiale (DRM), which is the only worldwide digital broadcasting system planned for long, medium and shortwaves that can use the same frequencies currently allocated to the amplitude modulation (AM) broadcasting service in the spectrum up to 30 MHz. The system is currently active but with only a few stations.*
- *DSD+ (Digital Speech Decoder) è un programma open source per la decodifica di segnali di parlato digitale multistandard come il DMR, Dstar, Fusion, P25, ecc...*

In general for decoding there are some aspects to consider in order to improve the chances of success, these are the main indications:

- Except in special cases, check whether your "Virtual Audio" program is configured for 48 ksps sampling rate on both input and output ports.
- Check that the SDR software is set to an appropriate volume level (not too low and not too high). All decoding programs have a level indicator that allows you to see the incoming signal and fine-tune it. You can start with a volume of 60/70% if the decoders don't report any errors... Remember that when the audio is redirected, for example to a Line1 or similar, it is no longer heard through the loudspeaker, but often the software comes with a suitable "audiorepeater" in case you still want to listen to the digital signal being processed.
- Disable Squelch and all those plugins (e.g. Audio Processor or Filters) that act on the audio level, which must be absolutely deactivated when receiving digital signals, otherwise they will result in incorrect or incomplete decoding or dirty signals.
- Check that the SDR software is set to the correct reception mode for the decoder. For example, on HF the USB (upper sideband) prevails, while on VHF-UHF the FMN is used. For narrower digital modes such as CW, DGPS, RTTY, you can go gradually with a narrow filter of 400 or 600 Hz and increase to 1500/3000 Hz for FT8 or wefax. You can also do it the other way around: start with a wide filter and then narrow it down to reduce noise and get proper decoding.

We should now be able to start searching the radio waves for signals other than speech, and make use of the many websites (with frequencies and lists of utility stations), to better understand what we will find in our listening sessions...
I would recommend the UDXF (Utility DXers Forum) for the exchange of news and information related to utility stations and signals below 30 MHz: ***http://www.udxf.nl***

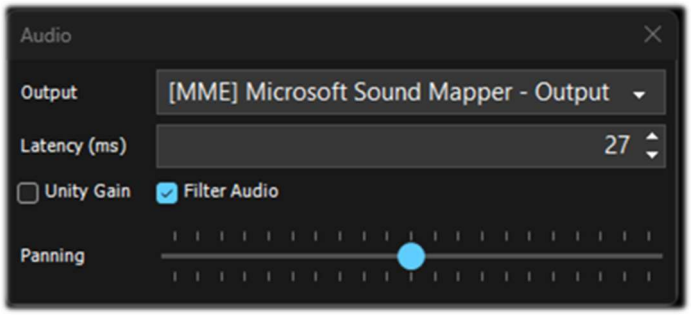
Much more complex and fascinating topics are the analysis of signals and transmission modes and related protocols.

It would take a book just to minimally introduce the subject (there are a few on the net) so I will just give a flash, citing the most professional I know and the only one of its kind, the blog by Antonio Anselmi: ***http://i56578-swl.blogspot.com*** and also his Twitter : ***https://twitter.com/i56578_swl***

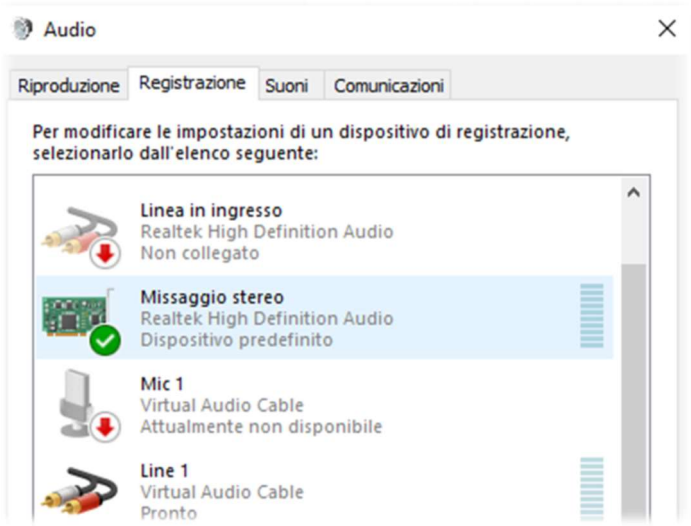
WARNING! SOME OF THESE SYSTEMS IT COULD BE ILLEGAL IN YOUR COUNTRIES! Check carefully and thoroughly the regulations in force in your country. Some of this radio system was specifically designed for use by government, emergency services, for public safety networks, etc etc. who all share spectrum allocated.

Another possibility is to use the sound card to share the signal without any particular need for decoding, but to read in your own language what a broadcasting station is transmitting at that moment...
In fact, you can direct the audio to the Google translator to have it translated in real time into your native language (try it to believe it!). This is really very nice and funny, let's see what you need to do...

The prerequisite is to use the Google Chrome browser, which allows you to convert speech audio directly live via your sound card on your computer.



"Audio panel" SDRsharp with Input/Output for your sound card. You can also use "Stereo Mix" by enabling it in the "Recording" tab in the Windows Audio setting.



If the entry does not appear, you need to click on the other input devices and temporarily choose "Disable".

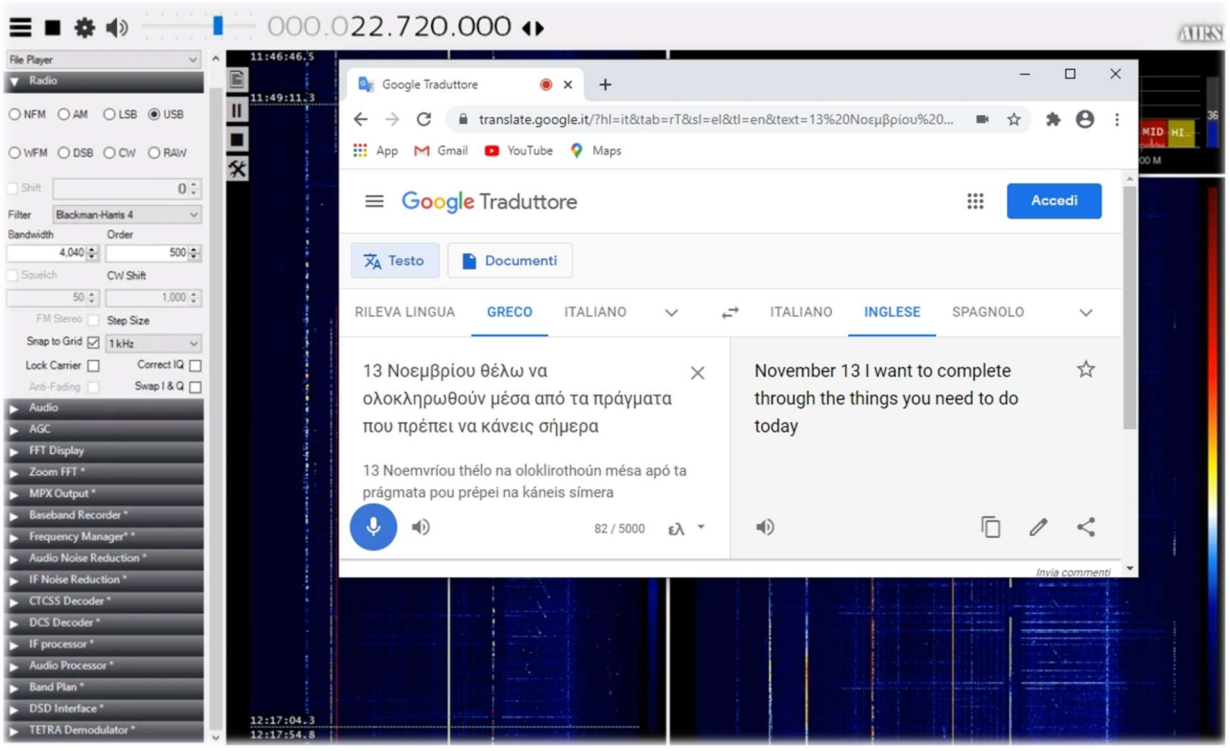
It should now be enabled and set as "Default device" with a green tip icon.

Access Chrome's settings by clicking on "Settings", then "Privacy and security", then "Site settings" by scrolling until you find "Authorisation - Microphone". From the drop-down menu, select "Stereo Mix".

You start up Google Chrome, select the source language (automatic detection is not yet working...) and the target language:

and finally click on the blue icon  of the microphone symbol and this is the result, when I captured China Radio International on 7435 kHz frequency during a language lesson in Chinese, promptly translated into my own language.

here is another example…

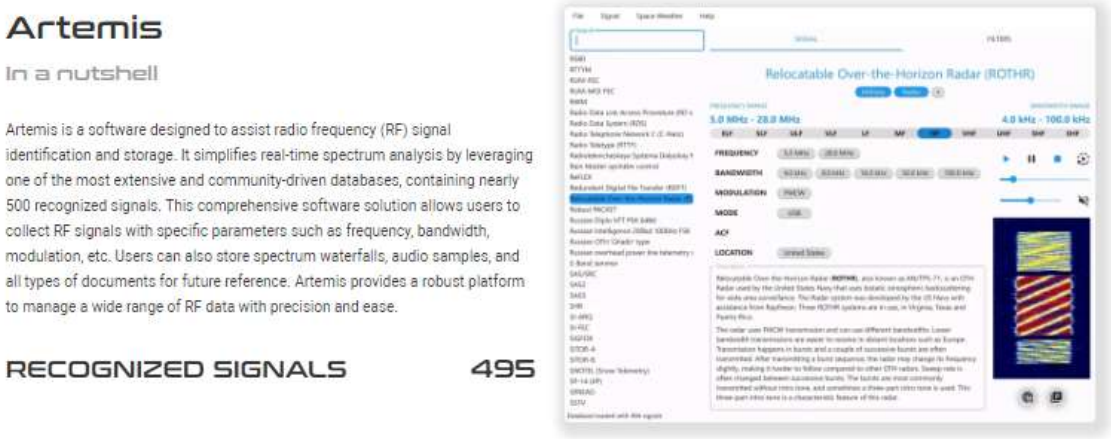


ARTEMIS v4.0.5
Signal identification and storage

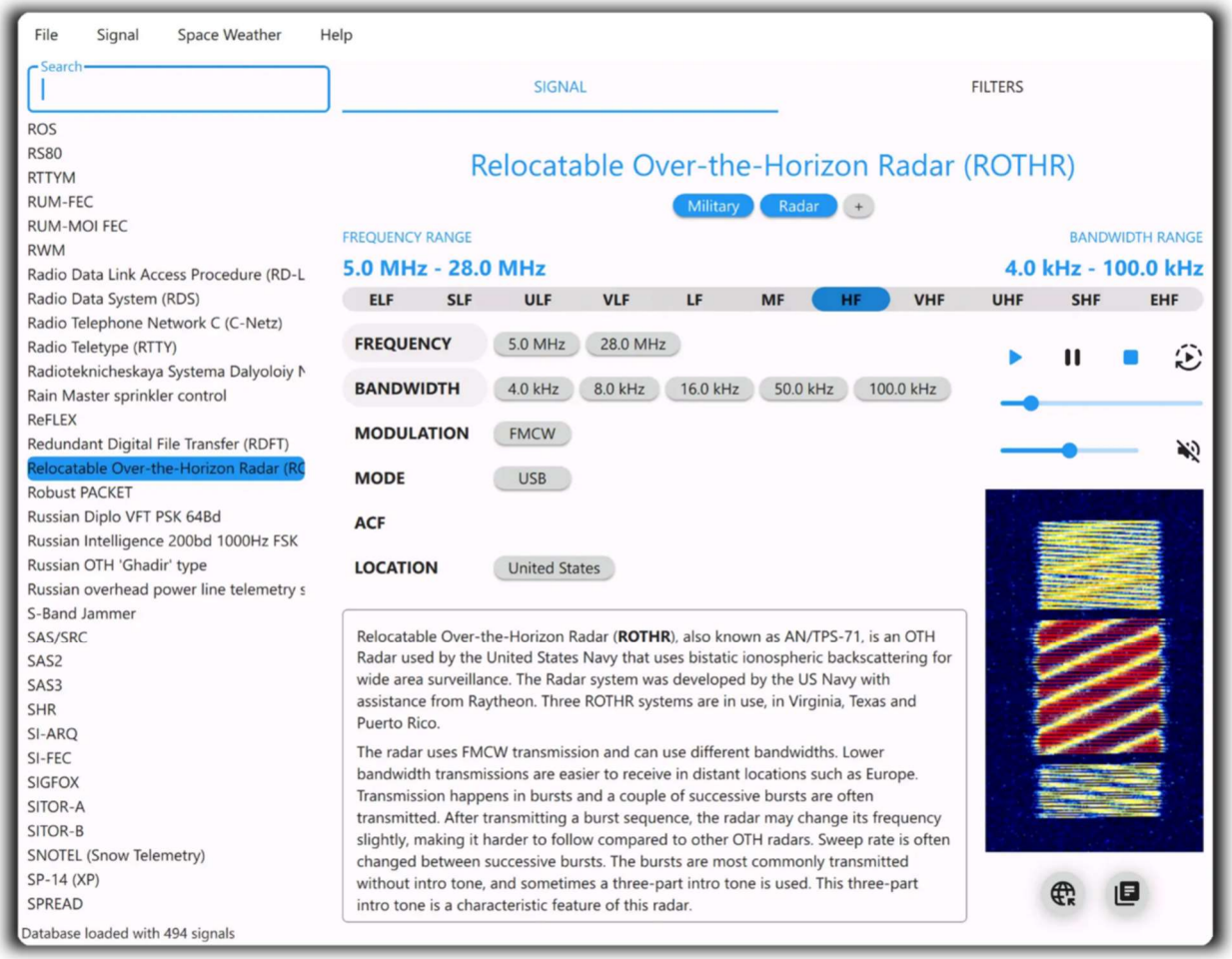
One of the first approaches to attempting to recognize the myriad types of signals and modulations is to use this free software, an indispensable tool aimed at all radio listeners thanks to the work of developers Marco and Alessandro.
Artemis is software designed to assist in the identification and storage of radio frequency (RF) signals. It simplifies real-time spectrum analysis by leveraging one of the largest and most community-oriented databases containing as many as 515 recognized signals. This comprehensive software solution allows users to collect RF signals with specific parameters.

The visual aid provided by our SDR# (thanks to the waterfall) allows you to take advantage of a real-time comparison with the excellent Artemis database and the properties of different signals (frequency, bandwidth, modulation, ACF, etc.) by verifying what you are looking for through reproducible audio samples and related image. A set of filters allows you to narrow down your search, making it easier to identify. However, I would like to point out that at the moment this is a user-led offline identification aimed at identifying radio signals through sound and waterfall samples and not an automatic signals recognition system...

Early June 2024 saw the release of the latest Artemis written in Python 3.11 which harnesses the power of the PySide6 graphics framework. This approach greatly reduces the need for third-party libraries since all operations are handled natively by the program itself or by PySide6 for all supported operating systems. To ensure a simple, clean and non-invasive installation via downloadable executables from the site, Artemis always comes with a portable version of Python, completely independent of any pre-existing installation, containing all the necessary libraries. It is available for the following O.S.: **Windows, Linux, Mac OS, and Raspberry.**



It all stems from Carl Colena's online guide to signal identification: Sigidwiki.com with the same purpose of addressing the identification of radio signals through examples of sounds and waterfalls:
https://www.sigidwiki.com/wiki/Signal_Identification_Guide

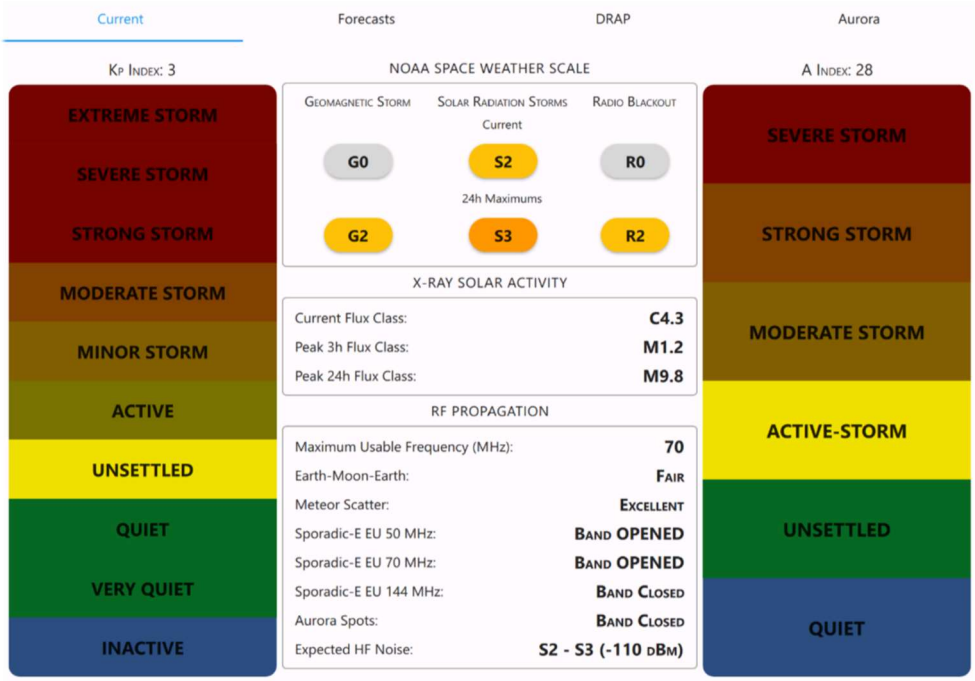


This is the main screen. From the "File" menu you load and manage the signal database which opens to **"SIGNAL"**

- On the left side the list of signals in alphabetical order (with the possibility of searching via the "Search" field on the contents of the whole database)
- in the center a large set of technical information and code description, customizable on the user side via the icon 
  In fact, Artemis allows one to create one's own database containing a collection of signals. These can be assigned an arbitrary number of parameters and/or documents for future reference. Personal databases can be exported and imported so that you can share your collections with other dx'er enthusiasts.
- on the right side a player to play the sound of the audio sample and display the respective waterfall. The icon  links directly to https://www.sigidwiki.com

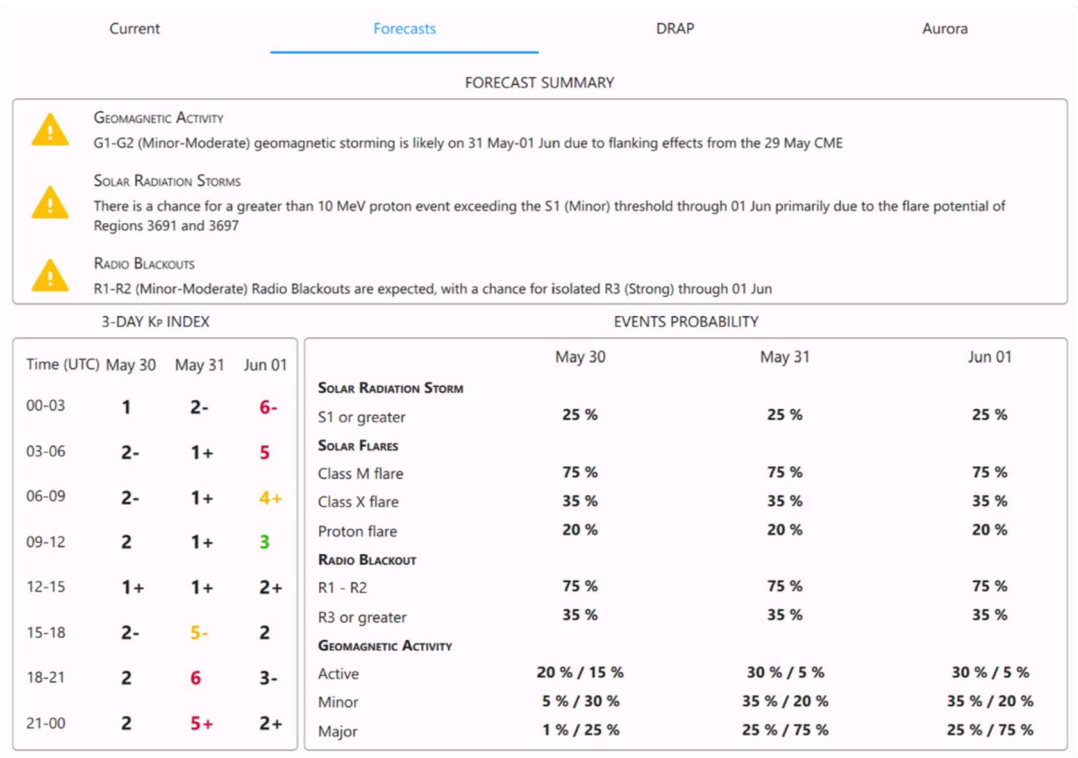
There is also the **"FILTERS"** tab to create specific search filters on the fields: **Frequency, Bandwidth, ACF, Modulation, Location, Category**

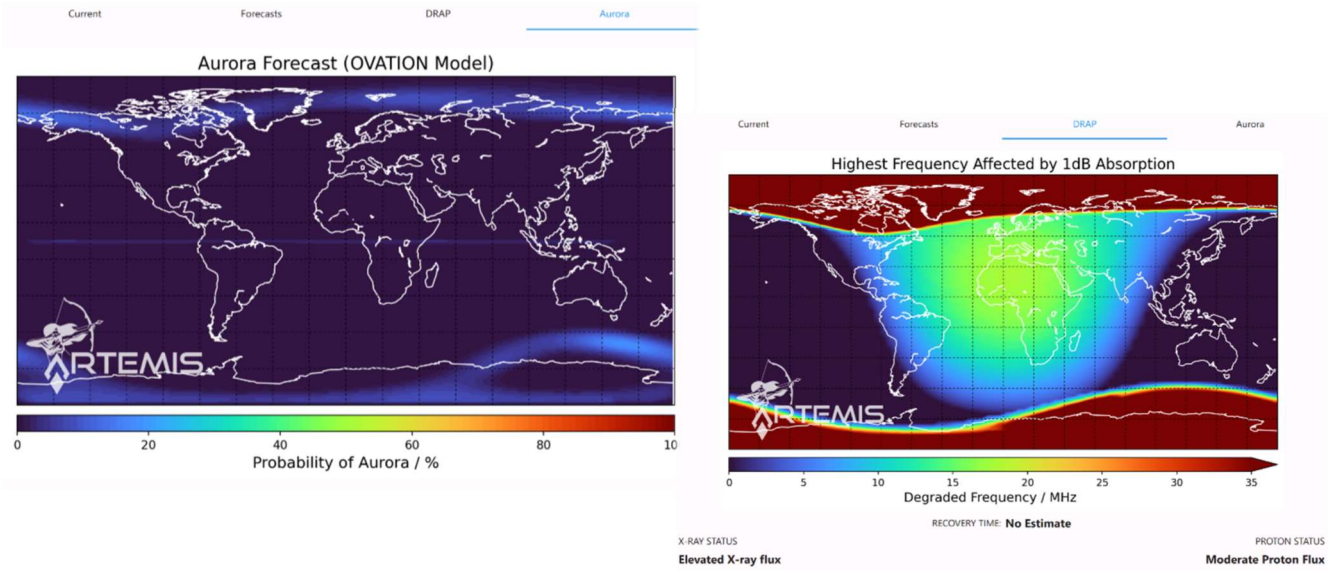
The **"SPACE WEATHER"** menu provides access to propagation data and related submenus: **Current, Forecast DRAP, and Aurora**. In fact, Artemis aids in propagation analysis with comprehensive monitoring of a variety of data and forecasts in real time (every 5 minutes) to help users predict the effects of solar, geomagnetic conditions on radio communications and any bands open for E-skip...



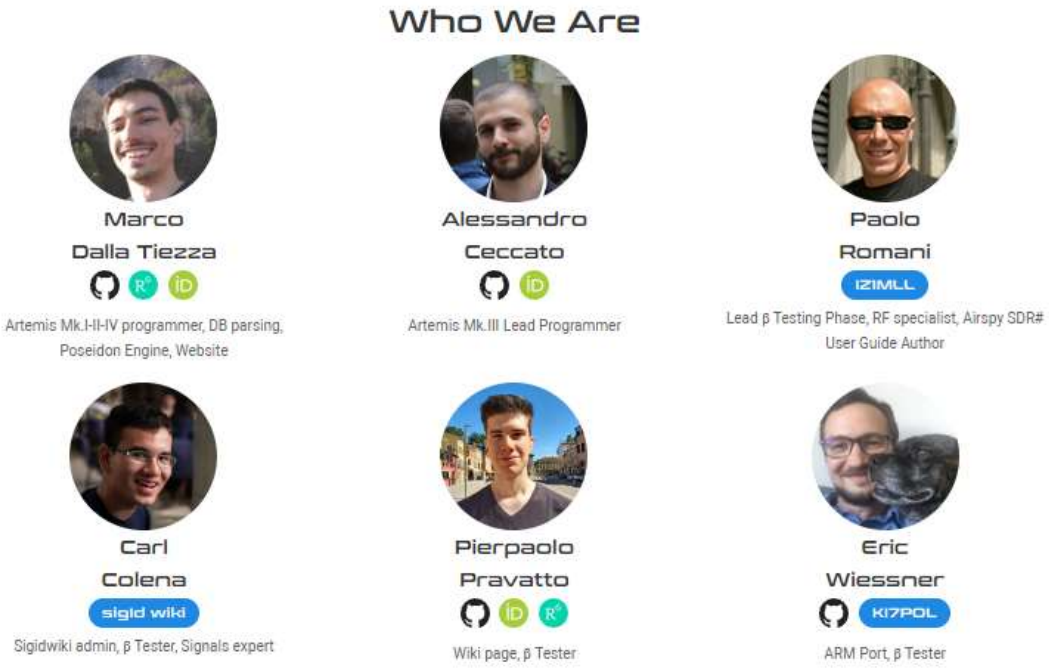
Some of the features include:

- Kp/Ap indexes & NOAA GSR Scale Alerts
- 3-day Space Weather Forecast
- Maximum Usable Frequency
- Earth-Moon-Earth, Meteor Scatter, Sporadic-E, Aurora Spots
- Expected HF Noise
- D-Region Absorption Prediction (DRAP)
- Aurora Forecast (OVATION Model)





Documentation on "space weather" can be found here:
https://aresvalley.github.io/Artemis/space_weather/current



I like to mention that from the very beginning of the journey together with Marco I had the honor of participating as a beta tester!

This is the link to the beautiful, graphically very appealing site:
https://www.aresvalley.com/

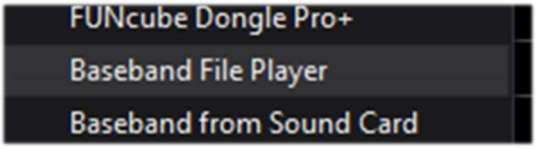
To view a nice video of TECH MINDS:
https://youtu.be/W_8Y_4FvoHI?si=nEGG5D0UdQbS3SwU

IQ Audio Samples

If someone needs to familiarize himself with the advanced functions of SDR# to test the goodness of filters and denoisers at various levels, it may be useful, at least in the beginning, to have a reliable archive of baseband files.
On the net is available this archive with a lot of files in WAV format from Günter Lorenz and valid for the purpose:

http://pira.fmlist.org/perseus/

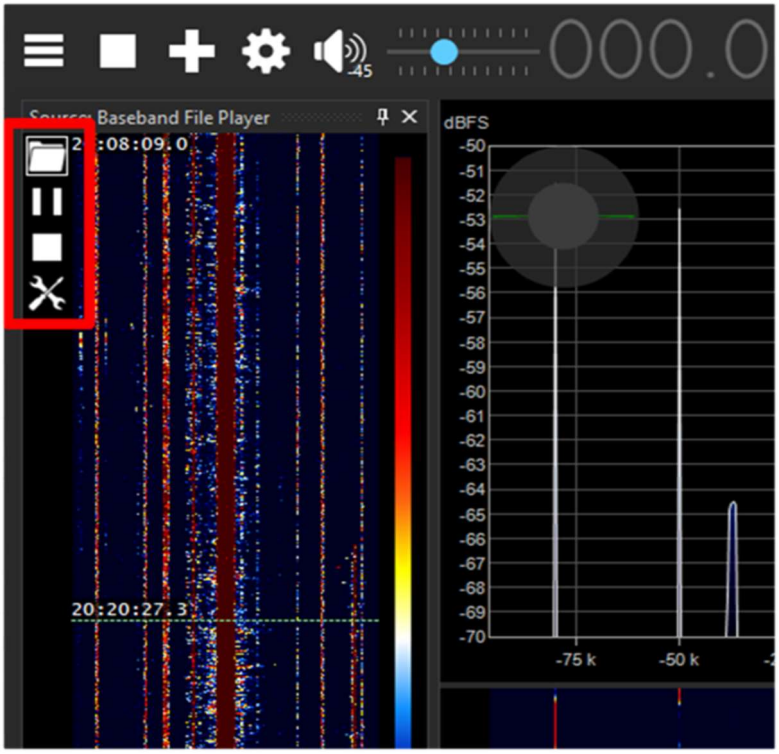
Let's download a few by hovering the mouse over the file name and right-clicking...



At this point you launch SDR#, from the Source panel you select "Baseband File Player"

To search the HD and upload an IQ file, simply click on the icon  (see step 4 - device configuration)

A side window will open that will allow you to move within the WAV file using the buttons highlighted in the rectangle in red color.



Then, at a more leisurely pace, everyone can make their own audio samples through the functions of the "Baseband Recorder" to which I refer reading in previous chapters...

Listening recipes …old and new

How to combine a great SDR and some excellent software to set up a useful receiving capability.

As in the best cookbooks are written the recipes, ingredients and operations necessary to deal with culinary dishes of all kinds, in this new chapter I will collect some screenshots, just with a title and a few brief comments, leaving the images the right weight and trying to arouse personal interest for subsequent insights that will be made by following the instructions of the developer of the individual software indicated.

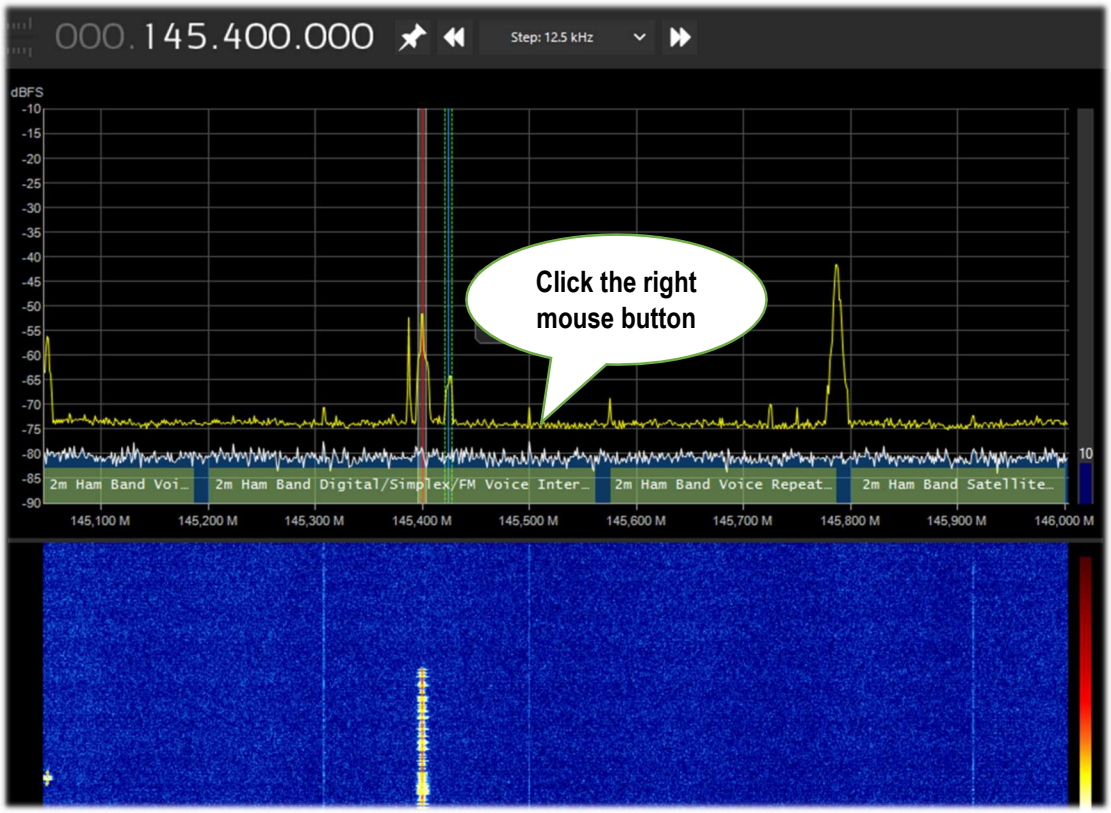
I would like to point out that ALL third party applications are made by different individuals/companies who have no connection to SDR# and Airspy. Third-party applications are stand-alone programmes that add or complement functionality.

WARNING! SOME OF THESE SYSTEMS IT COULD BE ILLEGAL IN YOUR COUNTRIES! Check carefully and thoroughly the regulations in force in your country. Some of this radio system was specifically designed for use by government, emergency services, for public safety networks, etc etc. who all share spectrum allocated to a city, county, or other entity.

The legendary yellow 'peak colour' line (see RF Spectrum feature) SDR#: RF Spectrum + right mouse button

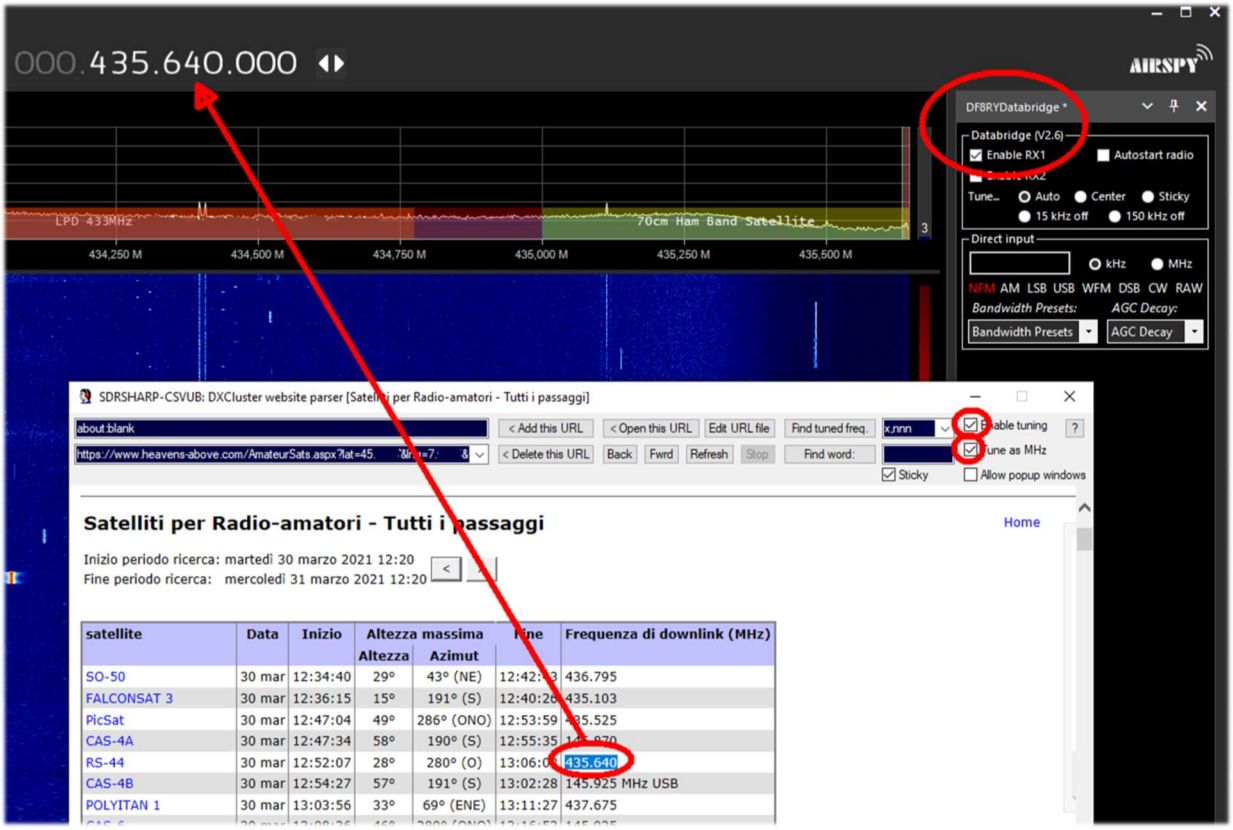
I find this option very interesting, a kind of chronological memory of the RF Spectrum. In the example, in the amateur radio band 2 meters, already after a few minutes you can see the peaks of the stations that have been activated and, positioning the mouse over them, you can read the frequency and intensity of the signal received.

An idea could be to use it in some portions of the spectrum not well known and after a few hours to see what has occurred ... a bit like going fishing with our SDR 😉

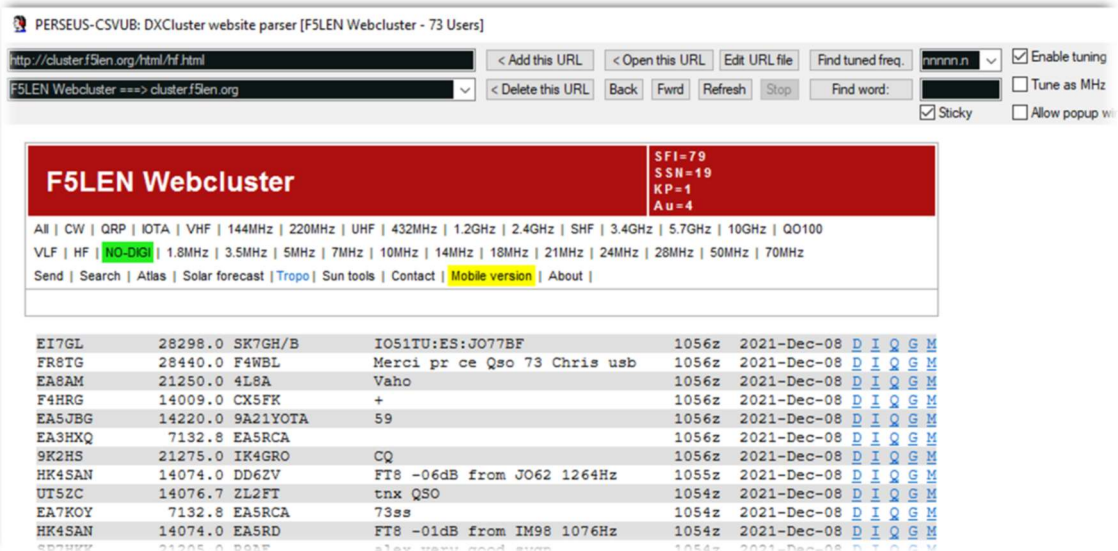


Tune a frequency with a simple move
SDR# + CSVUB plugin in “frequency parser” mode

With CSVUB plugin, previously mentioned, it is possible to tune the VFO of SDR# only highlighting the frequency taken from a site like DXcluster or, as in the following example, from a site of calculation of radioamateur satellite transits.
Enable the DF8RYDatabridge plugin (top right) with the flag on "Enable RX1" will access from the menu WEB / DX CLUSTER WEBSITE PARSER (or with Ctrl+Shift+D keys) where you will choose the URL to which we want to connect…
Really convenient and very fast!!

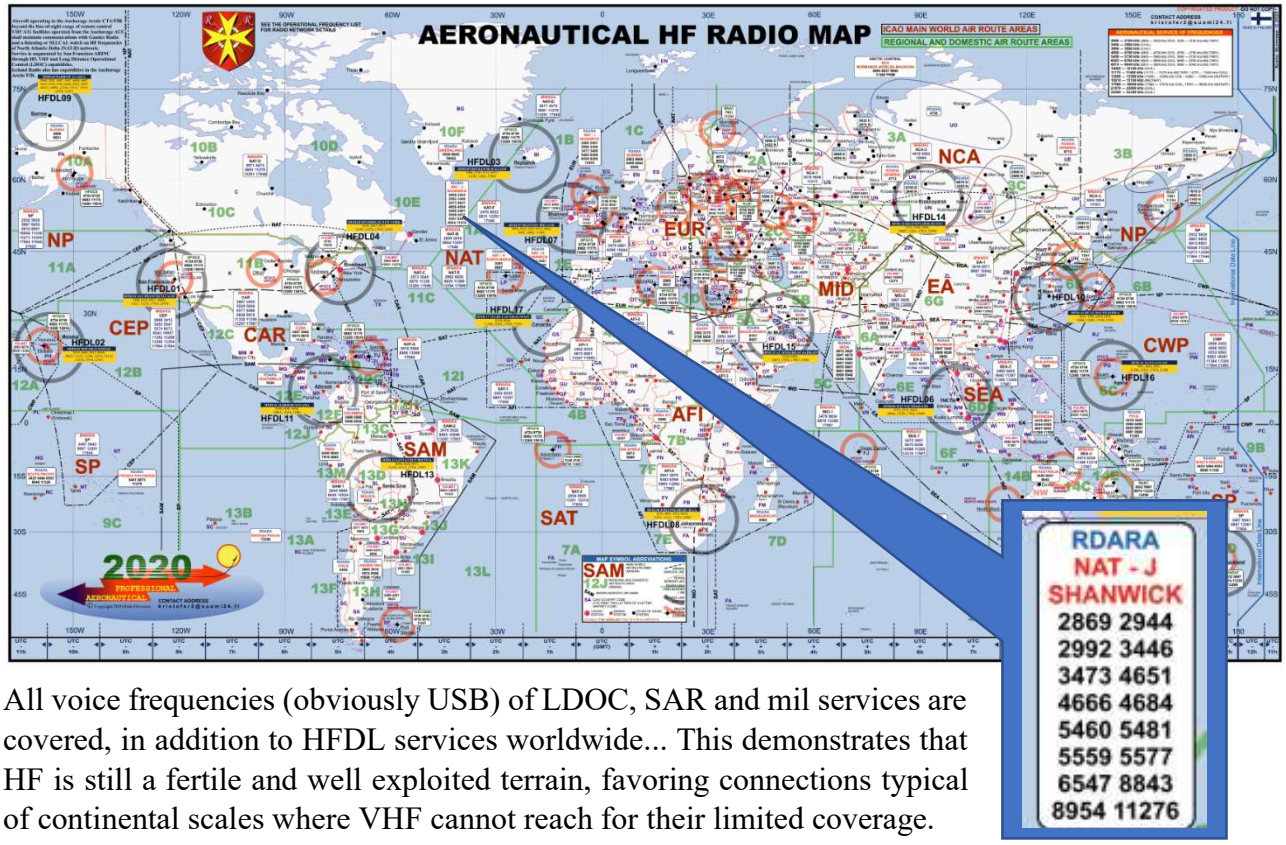


The same thing is possible in HF with one of the many radioamateur Webclusters even more interesting...



Aerolist ...the entire aeronautical world!
Airspy HF+ Discovery

An excellent list is Risto's Aerolist (OH2BVB), known by most of us, which reports all HF frequencies used by aircrafts in flight, towers and company operators. The package he distributes for a modest fee includes an Excel file with three thousand records, MWARA, Volmet, RDARA tables and a large list of high resolution PDF maps and charts as well as many audio samples...



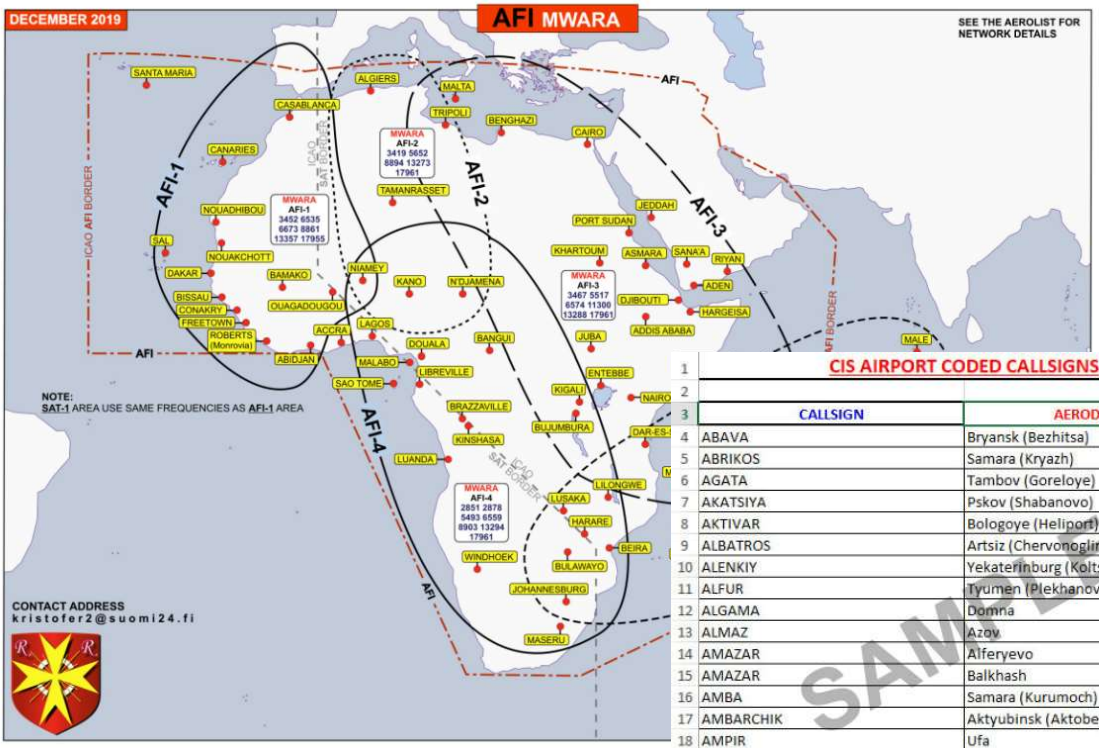
All voice frequencies (obviously USB) of LDOC, SAR and mil services are covered, in addition to HFDL services worldwide... This demonstrates that HF is still a fertile and well exploited terrain, favoring connections typical of continental scales where VHF cannot reach for their limited coverage.

All aircraft crossing continents and oceans must still be able to rely on shortwave to contact air traffic controllers, assisted of course by satellite communications and new technology (though not all of it always available in certain transpolar routes or where satellite coverage is poor or critical).

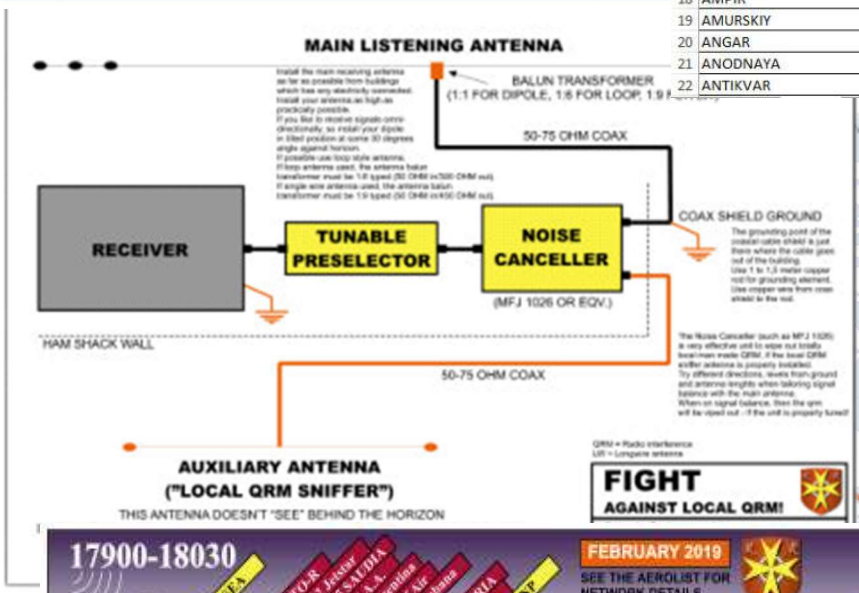
5643		DEC19	MWARA SP	Auckland OAC (SP-6), San Francisco OAC (SP-7), Nadi ACC (Fiji) (SP-6/7), Tahiti (Papeete) ACC (SP-7), Brisbane OAC (SP-6), Nauru Is. ACC, Pascua ACC (Easter Is.)	
5646			ITU ALLOCATION	MWARA NCA	
5646			ITU ALLOCATION	RDARA 12G	
5646		JUL13	LDOC	SAUDIA, Jeddah (Domestic flights)	
5646	1		MWARA NCA-1	Khanty Mansiysk, Syktyvkar, Yekaterinburg, Vologda	
5649			ITU ALLOCATION	MWARA NAT SEA	
5649		SEP20	MWARA NAT-C	Gander OAC, Shanwick OAC, Iceland (Reykjavik) OAC; (Central and Northern routes with aircrafts registered east of 30W)	
5649	1		MWARA SEA-2	Sanya ACC, Singapore ACC, Manila ACC, Bangkok ACC, Phnom Penh ACC, Hong Kong ACC, Vientiane ACC, Hanoi ACC, Ho Chi Minh ACC, Kota Kinabalu ACC	
5650		JUL20	VOLMET/R	Khanty-Mansiysk meteo. The WX information of areas Noyabrsk, Khanty-Mansiysk, Salekhard, Tomsk. Transmission 15 minutes by russian language and 15 minutes by english. Then again by russian language 15 minutes, then by english, endlessly. Taped information. Automated female voice. Federal Air Transport Agency/Aeronautical Information Service.	
5652			ITU ALLOCATION	MWARA AFI CWP	
5652		FEB10	MWARA AFI-2	Algiers ACC (Maghreb Control), Niamey ACC (East sector), Tripoli ACC, Maiduguri ACC, N'Djamena ACC, Tamanrasset ACC, Ghardaia (Noumerate ACC)	
5652		FEB17	MWARA CWP	Tokyo OAC, San Francisco OAC	
5652		JAN17	HFDL	Riverhead (New York USA) [4]	
5653	A	OCT08	UNID	Greek/YLOM/11OCT02/0416UTC // 01OCT08/0632UTC/Calling [...TIRO TREA...]	
5654	A	NOV13	UNID	RR/2OM/13NOV2013/1558UTC/Station c/s LODA-40 and KARLOTA-57/Suspected russian MIL AERO	
5655			ITU ALLOCATION	MWARA EA SEA	
5655		APR20	MWARA EA-2/SEA-2	Singapore ACC, Manila OAC, Hong Kong ACC, Kuala Lumpur ACC [LUMPUR], Ho Chi Minh ACC, Vientiane ACC, Sanya ACC, Hanoi ACC, Bangkok, Phnom Penh, Guangzhou, Irkutsk, Pyongyang, Ulaanbaatar	
5655		SEP20	HFDL	Hat Yai (THAILAND) [6]	

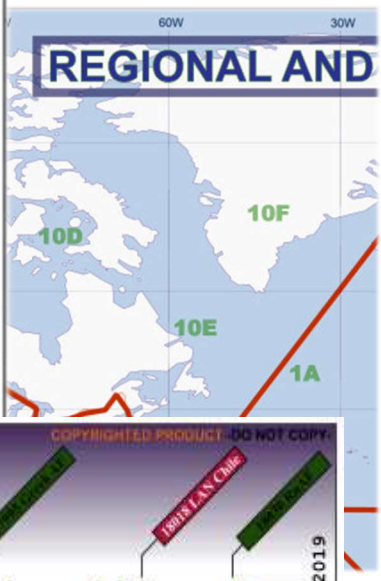
AERADIOS 2020 | MWARA FREQ | RDARA | HFDL | VOLMET | DELETED INFO | ABBREVIATIONS & COLOR STATUS

A recent email exchange with the author confirms to me that he has made a major update for the operating frequency list which is now updated for the year 2022. A minor update has also been made to the HF world radio map.

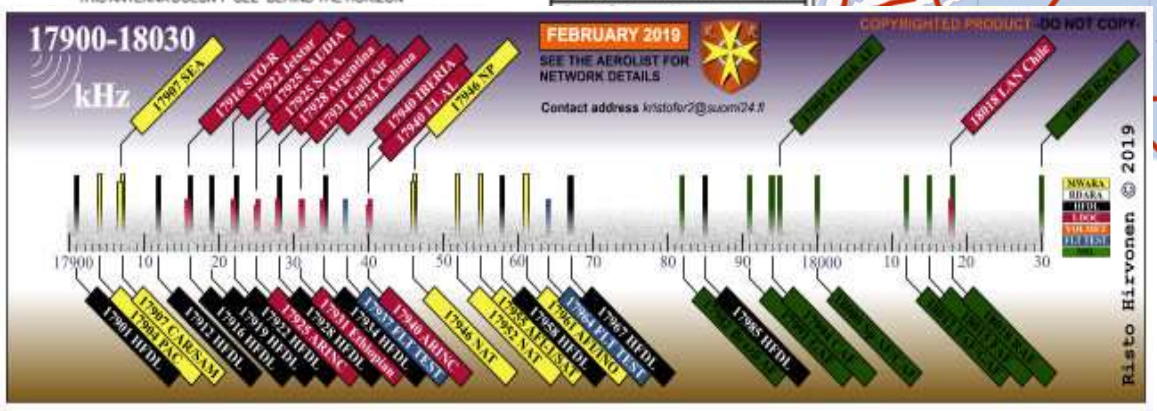


	CIS AIRPORT CODED CALLSIGNS	
1		
2		
3	CALLSIGN	AERODROME
4	ABAVA	Bryansk (Bezhitsa)
5	ABRIKOS	Samara (Kryazh)
6	AGATA	Tambov (Goreloye)
7	AKATSIYA	Pskov (Shabanovo)
8	AKTIVAR	Bologoye (Heliport)
9	ALBATROS	Artsiz (Chervonoglinskaya)
10	ALENKIY	Yekaterinburg (Koltsovo)
11	ALFUR	Tyumen (Plekhanovo)
12	ALGAMA	Donna
13	ALMAZ	Azov
14	AMAZAR	Alferyevo
15	AMAZAR	Balkhash
16	AMBA	Samara (Kurumoch)
17	AMBARCHIK	Aktyubinsk (Aktobe)
18	AMPIR	Ufa
19	AMURSKIY	Fregat
20	ANGAR	Kasimovo
21	ANODNAYA	Kazan (Borisoglebskoye)
22	ANTIKVAR	Yekaterinburg (Aramil)





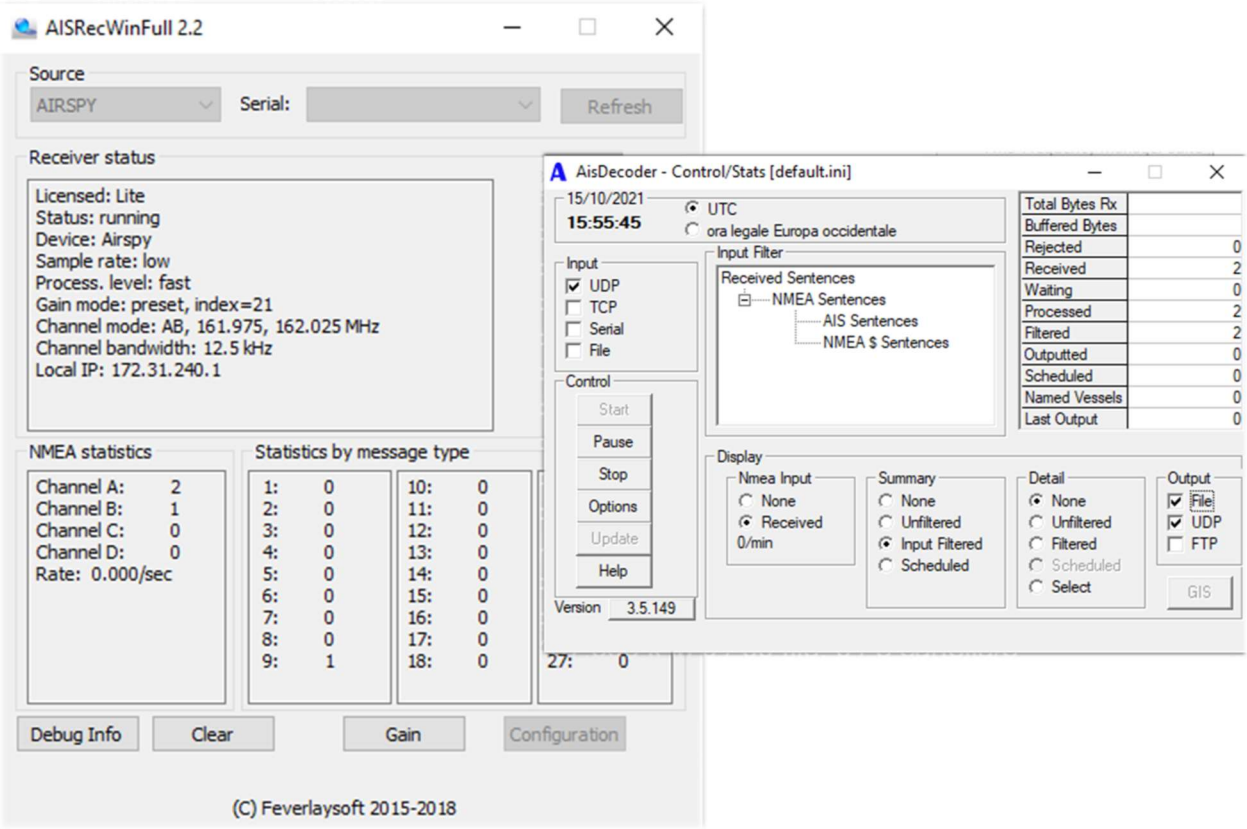
For



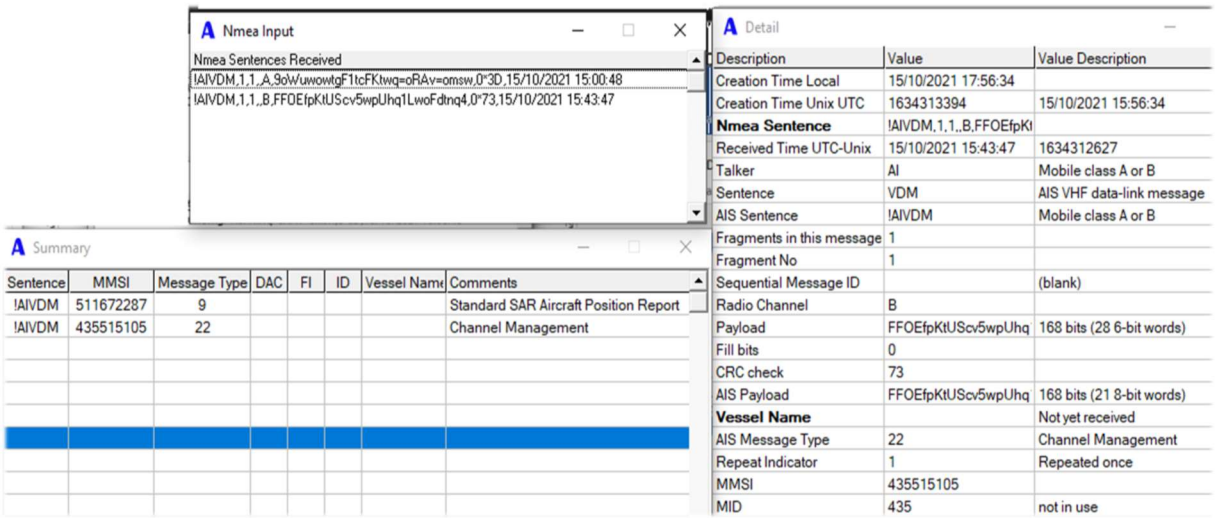
many more examples and informations this the reference site: ***http://www.elisanet.fi/bvb1438/***

AIS …to navigate a bit virtually!
Airspy R2 with software AISRec + AIS Decoder

In the vicinity of the coast it is easy to come across two VHF frequencies in the nautical band that H24 transmit AIS spots: 161.975 and 162.025 MHz.
The AISRec software for Windows allows you to simultaneously receive the two signals in IQ format and extract the NMEA sequences to send them via UDP to another software (AIS Decoder) for decoding all 27 types of AIS messages provided...



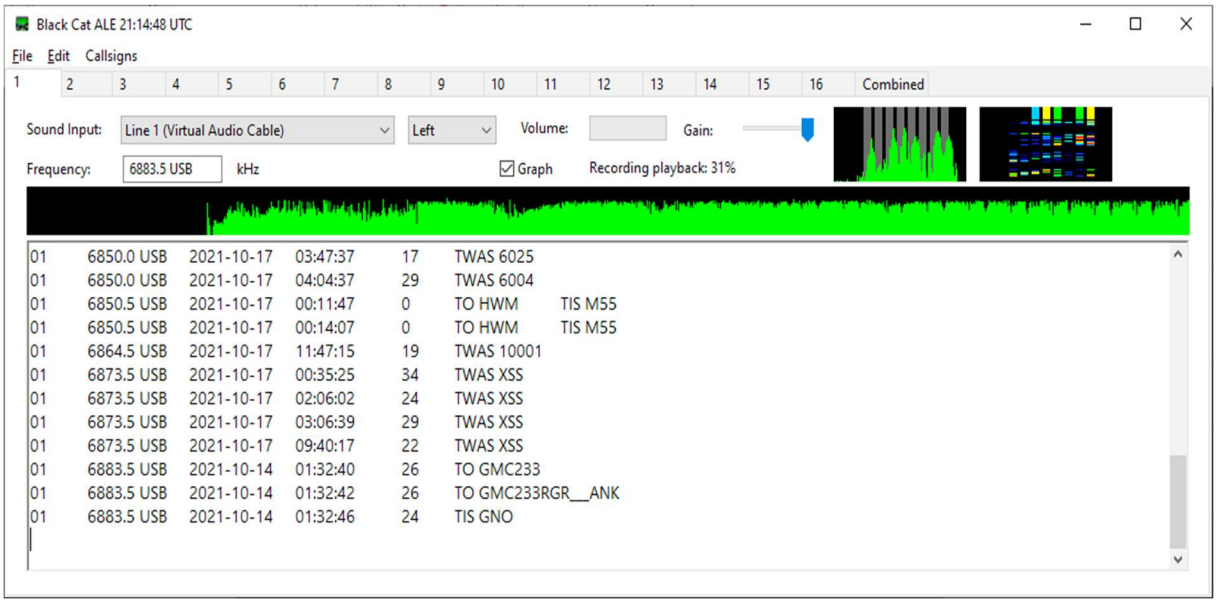
To complete the thing
, it is also possible to combine a map (e.g. with the OpenCPN freeware) which, thanks to the geographical coordinates received, makes it possible to display the position of vessels and various fixed stations received by our receiving system...



ALE …a new decoder, indeed a multi-channel decoder!
Airspy HF+ Discovery and software Black Cat ALE

HF is always fertile ground for developers with new ideas and strong technical knowledge...

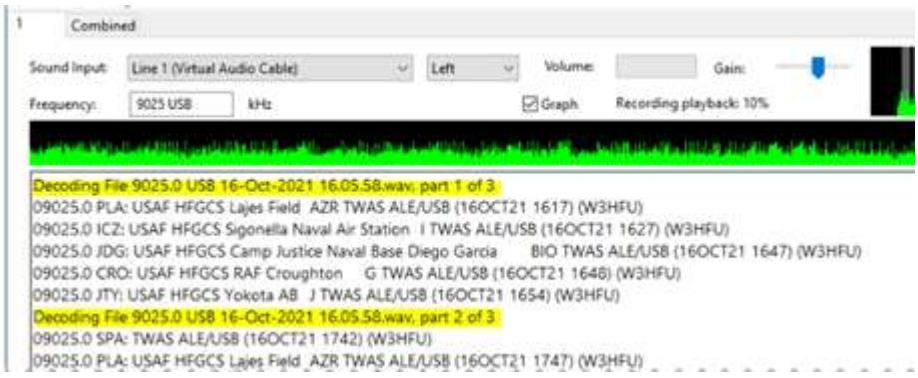
It is still in beta but you can download the demo that allows you to try for 30 days a software with extreme sensitivity compared to other software used by fans for a long time and with the ability to use up to 24 decoders simultaneously (SDR and computer permitting!).



With the "regular license" you can use up to 3 decoders simultaneously, while you can get up to 24 with the "High performance" mode.
It is possible to monitor in a totally automatic and independent way different frequencies or single net of interest, each one combined to a specific audio channel (example VAC on Line1/2/3/x).
Each decoder/tab will display the text (in different formats provided in the software) in a special screen 1/2/3/x, while in the "Combined" one there will be the result of all individual channels.

Other innovative features are:

- Use previously recorded audio files or WAV samples present on the network (even more than one at the same time) with an impressive decoding speed (even 10 times the real one)!



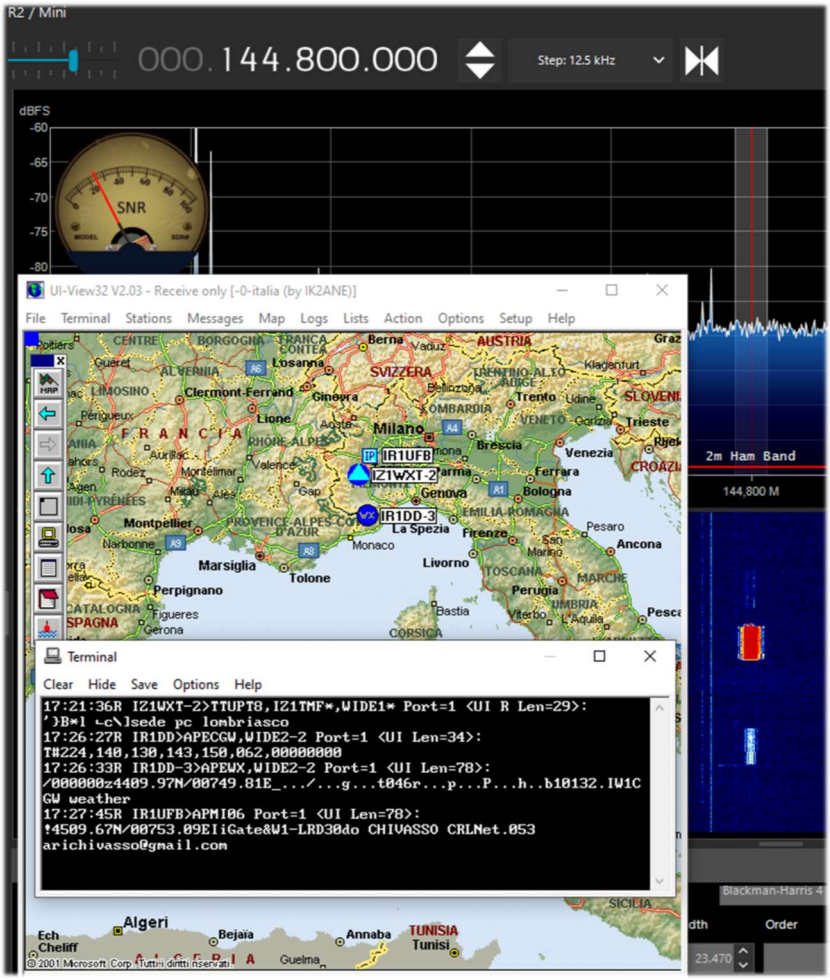
- Creation of Logs with different custom formats, also for the UDXF Bulletin Board
- Other features still under development to work on specific Callsigns / Net / ...

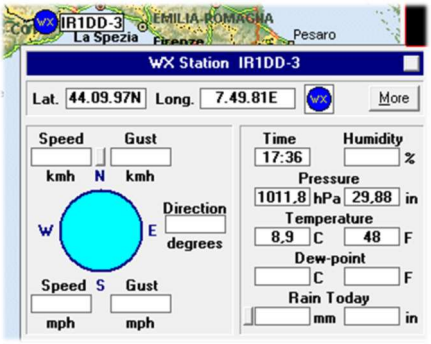
I have write a PDF guide that can be downloaded here:
https://blackcatsystems.com/download/BlackCatALEGuide.pdf

APRS
Airspy R2 with software AGWpacket/UI-view32

APRS in amateur radio was so fashionable in the 1990s, but it is still possible to do something about it today: here are some directions for reception. The frequency is 144.800 MHz (FMN).

In this brief mention of the radiolocation/information relay system, I have exploited two freeware software together with the result of having on a map the location, in the form of various icons, of fixed and/or mobile stations (digipeaters, vehicles, laptops, etc.).
A mobile station can get its location continuously updated by pairing a GPS with its VHF radio tuned to the indicated frequency.
It is possible to view on Google Maps (http://aprs.fi) all worldwide APRS traffic in real time, generated via radio and internet.





Some of these fixed stations also transmit weather informations of their QTH (wind, direction, pressure, temperature, etc. etc.). An example is this on the side.

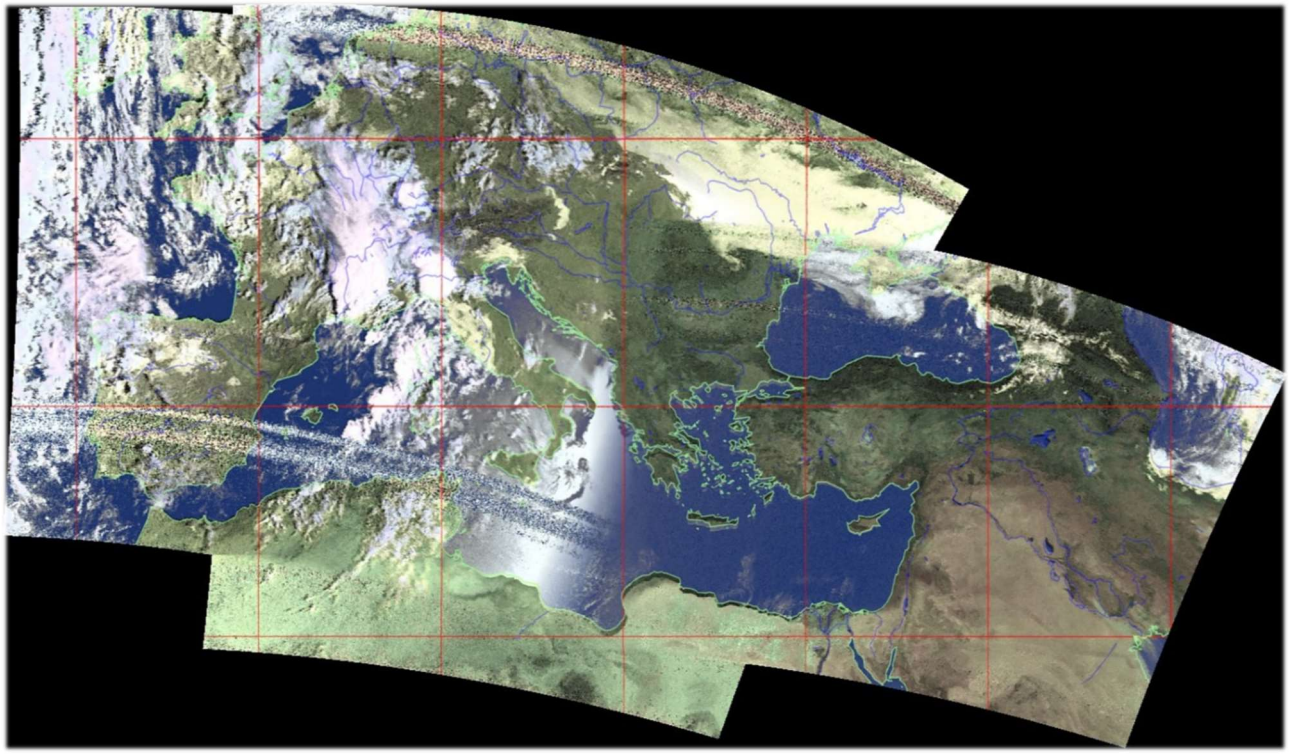
The APRS system has found great utility in emergency situations or natural disasters (earthquakes, floods) because it allows the movement of vehicles and people to be kept under control while also allowing the exchange of small packet messages between stations. In fact, traditional Packet Radio and APRS are very similar in operation in that they use the same AX.25 protocol with the substantial difference that in APRS it does not require that a connection be established between station and station, but the signals are "broadcast to all" in their data packets.

Another software that has always given me much satisfaction is **MULTIPSK by Patrick Lindecker.**

APT NOAA, images mosaic…
Airspy R2 with software WXtoImg

WXtoImg is one of the best software for fully automated decoding of APT and WEFAX (WXsat) weather satellite signals.
The software allows recording, decoding, editing and viewing in Windows, Linux and Mac OS X.

It supports real-time decoding, map overlays, advanced color enhancements, 3-D imagery, animations, multi-pass imagery, projection transforms (e.g. Mercator), text overlays, computer control for many satellite weather receivers, and much more...

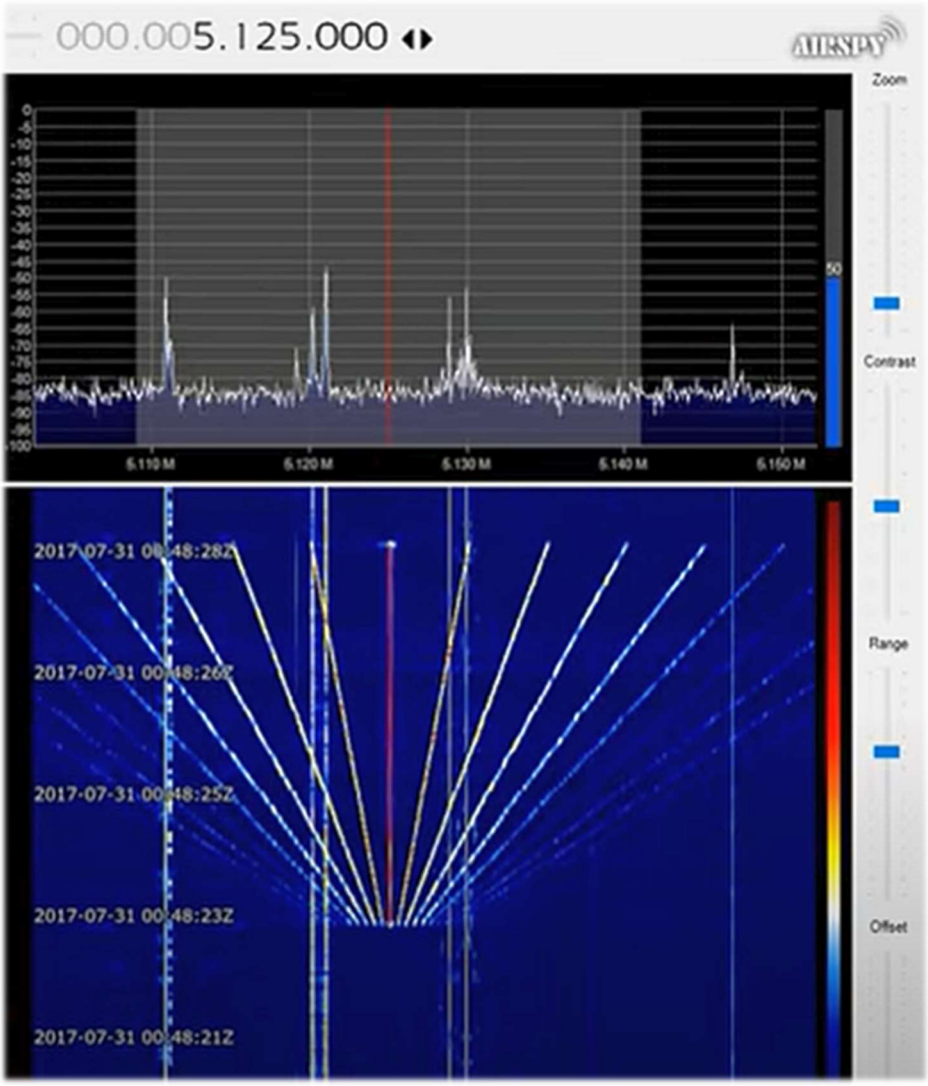


This a mosaic of images captured in September 2021 by friend Rob (IZ0CDM) assembling the output of signals received from the following satellites at later times:
NOAA15 06:52 UTC,
NOAA19 07:18 UTC,
NOAA18 07:59 UTC...

Arecibo sweeping … and its "inverted umbrella"
Airspy HF+ Discovery

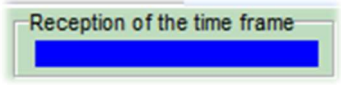
From my friend BlackApple62 I report below a very curious sweeping received in 2017 while tuned in shortwave at 5125 kHz.

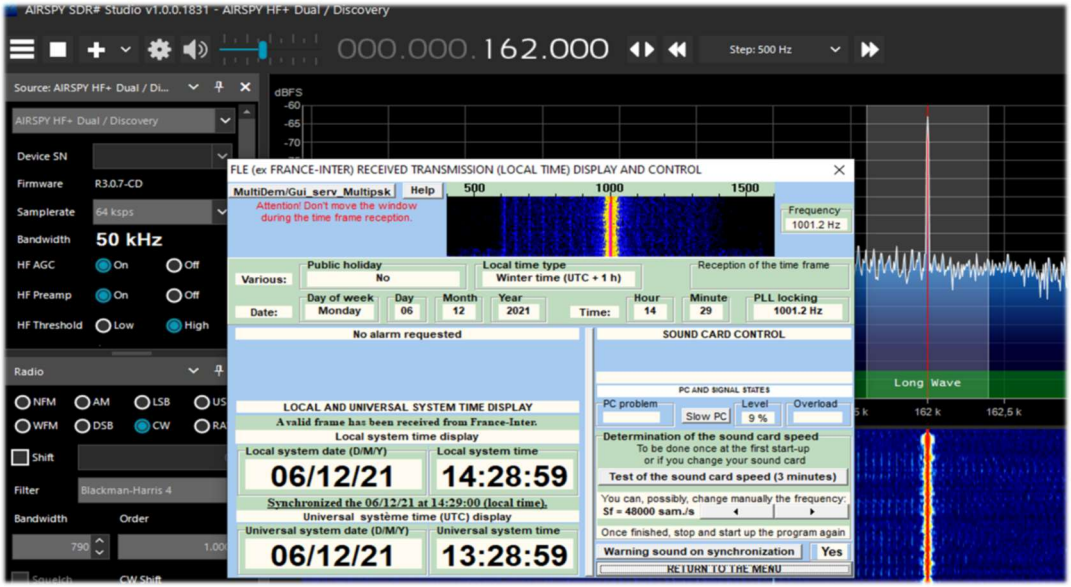
Truly a peculiar shape that was drawn into the waterfall while receiving radio signals from the HF radar at the Arecibo Observatory (Puerto Rico) when it was active at the time to study the state of the Earth's ionosphere.



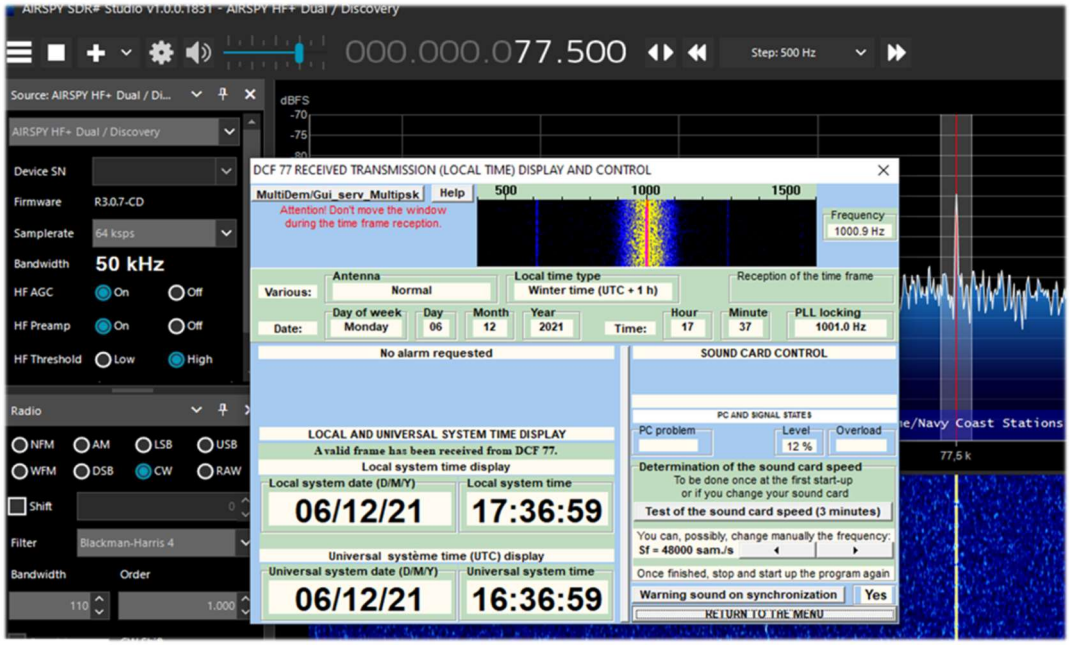
CLOCK, synchronize via radio the time of your computer
Airspy HF+ Discovery

"Clock", included in "MultiPSK" Windows software by Patrick Lindecker (F6CTE), provides date and time by decoding time frames received via radio from FLE (ex France-Inter), DCF77, HBG, MSF, BBC, WWVB, WWV, WWVH, CHU, RAI, JJY, or via GPS or the Internet.
In addition, you can synchronize the local (current) and universal time (UTC) of your computer with the time received by radio! After synchronization (confirmed with a loud beep) the computer clock will be accurate to within 1 second with the real time.
In the following screen, received on 162 kHz frequency of FLE (ex Radio France-Inter), once the signal is locked and confirmed by PLL Locking (in my case demodulated in CW), the field "Reception of the time frame" will be initially colored blue and after a while the decoding of the data of "Public holiday", "Local time type", "Minute", "Hour" and at the end of each minute of the remaining information: "Day of week, Day, Month, Year".



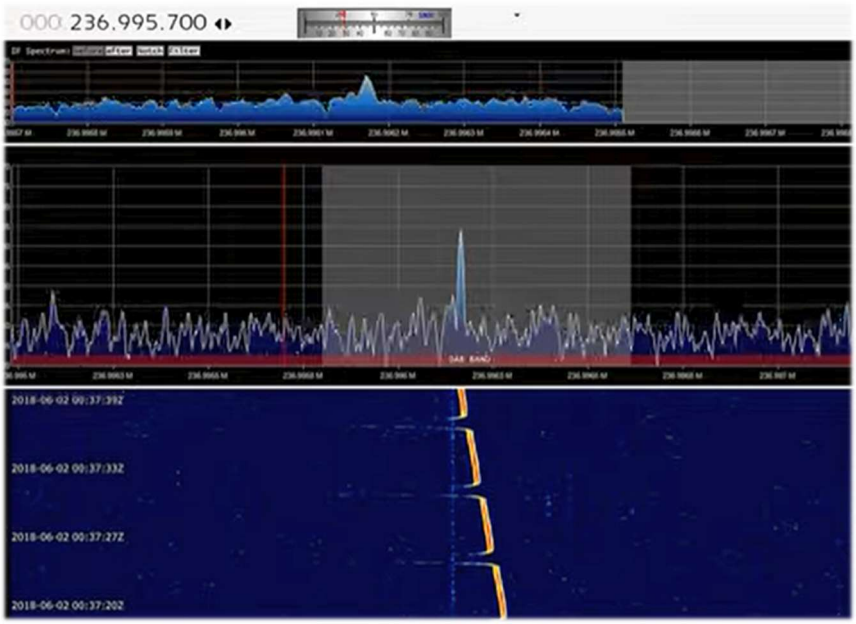


While the following, similarly, is a time frame of DCF77 (Mainflingen, Hesse, Germany) received by tuning at 77.5 kHz frequency.



“DeadSat” (Dead Satellite): still a sign of life from the 1960s! Airspy R2

One of the many things that has always fascinated and intrigued me is the presence of satellites in orbit, which at the end of their useful life, tirelessly continue to transmit their beating hearts to us even after many, many decades...



In this screenshot, kindly granted to me by my friend BlackApple62, you can see the UHF variable carrier, at about 237 MHz, of the U.S. satellite LES-1 (Lincoln Experimental Satellite) abandoned back in 1967 but still receivable!

If I have aroused some curiosity I refer you then to this unmissable and up-to-date link:

https://github.com/happysat/Deadsat/blob/main/README.md

where you can find useful and up-to-date information about our DeadSat

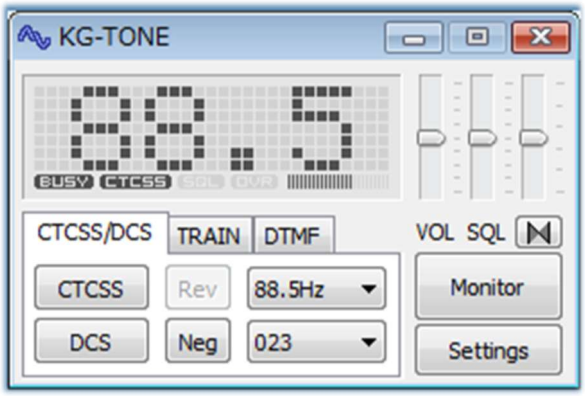
CBERS-2B	32062	2007-042-A	180.010	USB			Note 22.	17.03.2021 No Signal
LES-1	1002	1965-008-C	237.0000	USB		LES-1	Note 23.	19.03.2021
LES-5	2866	1967-066E	236.7487	BPSK			Note 24.	
OV 3-3	2389	1966-070-A	258.4750	USB		OV3-3	Note 25.	18.03.2021

You can then proceed to install the freeware Orbitron software that allows satellite tracking (or equivalent) and use the updated TLEs by picking them up by clicking directly on the icon highlighted above.

Decoding CTCSS / DCS / DTMF /…
Software KG-TONE

A very good external software to help identify unknown radio link signals using subaudio tones (or CTCSS) and/or DCS digital codes.

It's called KG-TONE and it's free. The latest release for Windows XP/Vista/7 is 1.0.1 (Dec'2011) at:
http://www2.plala.or.jp/hikokibiyori/soft/kgtone/kgtone.zip



In KG-TONE, the following sources were provided as input signals in menu "Settings / Wave input device" *(useful to know as SDR receivers were not contemplated at the time):*

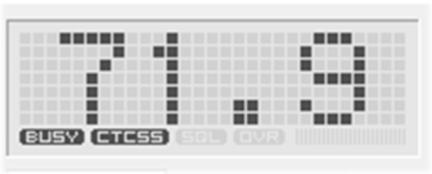
FM voice - obtained from the headphone socket or the loudspeaker socket, is not always good, as the audio path may be filtered in later stages (e.g. the elimination of audio subtones!).
FM detect – i.e. the signal taken before filtering by subsequent stages of the receiver: for decoding purposes it is better than the previous one.
12 kHz I/Q – the I and Q components are samples of the same signal detected orthogonally in phase and therefore contain different informations. With their separation it is possible to measure the relative phase of the signal components, which is useful not only for FM demodulation. This is the best mode, ideal for signal analysis and can be processed directly by the software without any loss. The manual at the time stated to check if your receiver was equipped with a 12 kHz I/Q output socket and referred only to the AOR-5001D and ALINCO DJ-X11 receivers.

Translated directly from Japanese (in the hope of interpreting it correctly from the brief instructions included with the software), I provide a comprehensive table of all the decoding possibilities in the various modes:

Source signal type	NQSL	CTCSS	DCS	TRAIN	MSK	DTMF
FM voice (audio)	**C**	**A**	**C**	*	*	*
FM detection	**A**	**A**	**B**	*	*	*
12 kHz I/Q	*	*	*	*	*	*

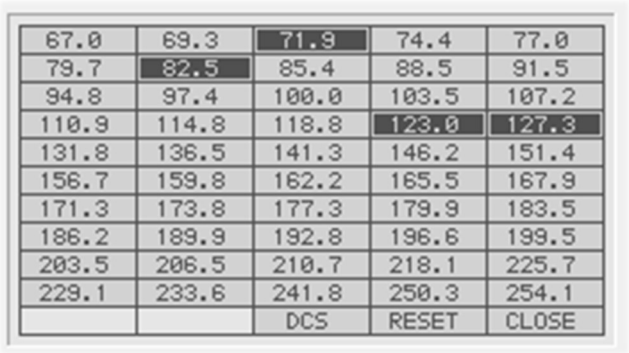
= Possible in many cases
(A) = Possible, but depends on the model
(B) = Impossible, but it depends on the model
(C) = Almost impossible
NSQL = Noise Squelch operation (noise silencing)

Operationally, using with our SDRs, I noticed no difference with the settings set to "Audio" or "Discriminator".



Turn on the audio from e.g. the VAC *(also running the audiorepeater.exe file to keep hearing the audio!)*, choose the audio input in KG-TONE and press the 'OK' button.
The software analyses the signals and displays the detected data in its small, graphically appealing panel. If the audio paths are correct, and the noise squelch is open, the "BUSY" icon and thus the detected tones will be highlighted on the left in bold.

It also has a "COMBO" mode with which you can have a larger underlying panel displaying all CTCSS or DCS and with a useful "memory effect" of all those activated over time appearing on a dark background. *Truly an excellent piece of professional software!*



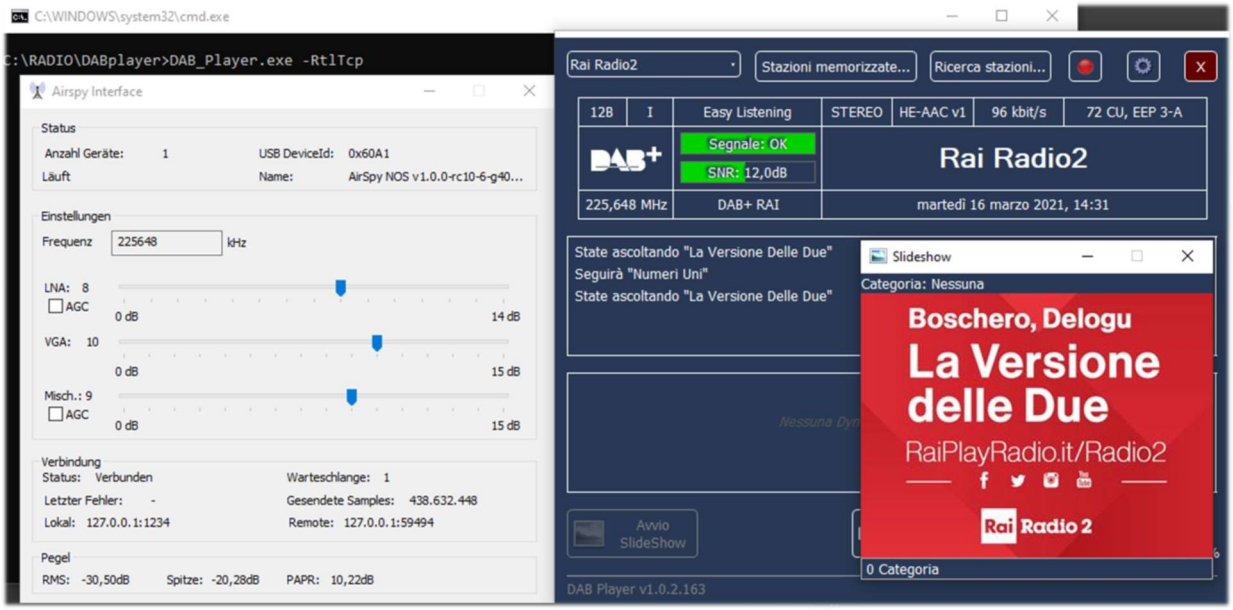
It can also detect DTMF but I was not able to test it for TRAIN / MSK modes, which are not active in my country.

Unless specifically required, the sliders can be held initially in the following positions:

Perhaps few people are aware that by pressing the button it is also possible to activate an "audio inversion band" decoder and adjust its tone (pitch) by slightly moving the vertical slider above...

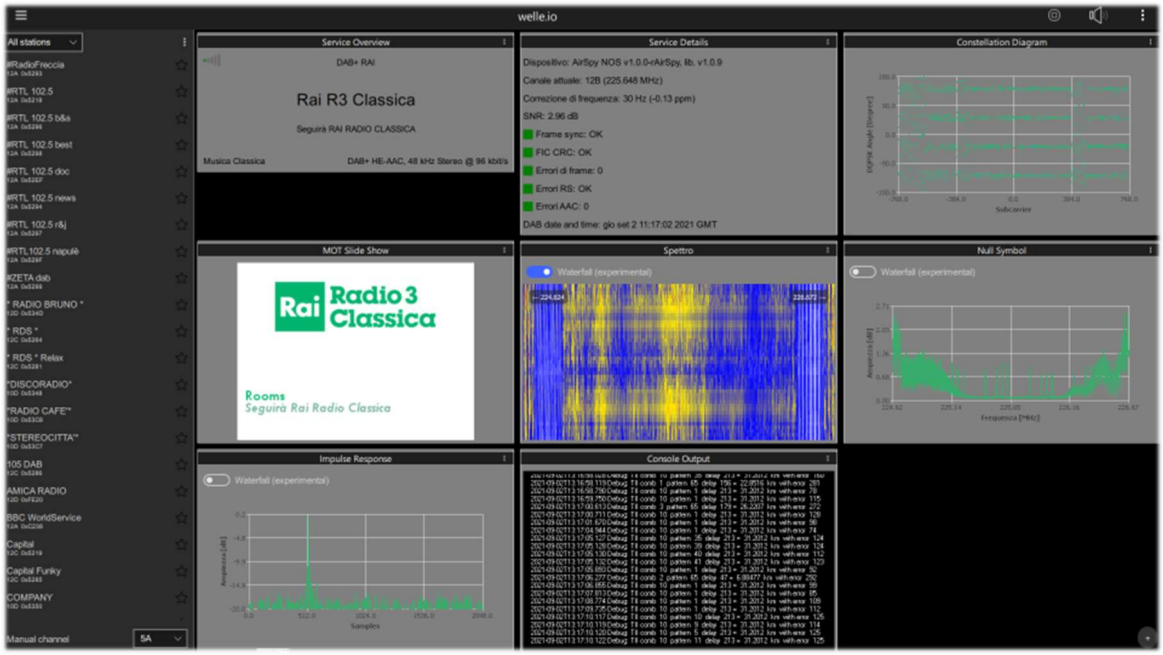
DAB / DAB+ (part 1)
Airspy Interface + DABPlayer

Simple but ingenious interface to connect via TCP your Airspy devices to Andreas Gsinn's DABplayer and enjoy the full DAB content with slideshows, quality recordings and lots of informations on Ensemble, FIC, MSC and audio...



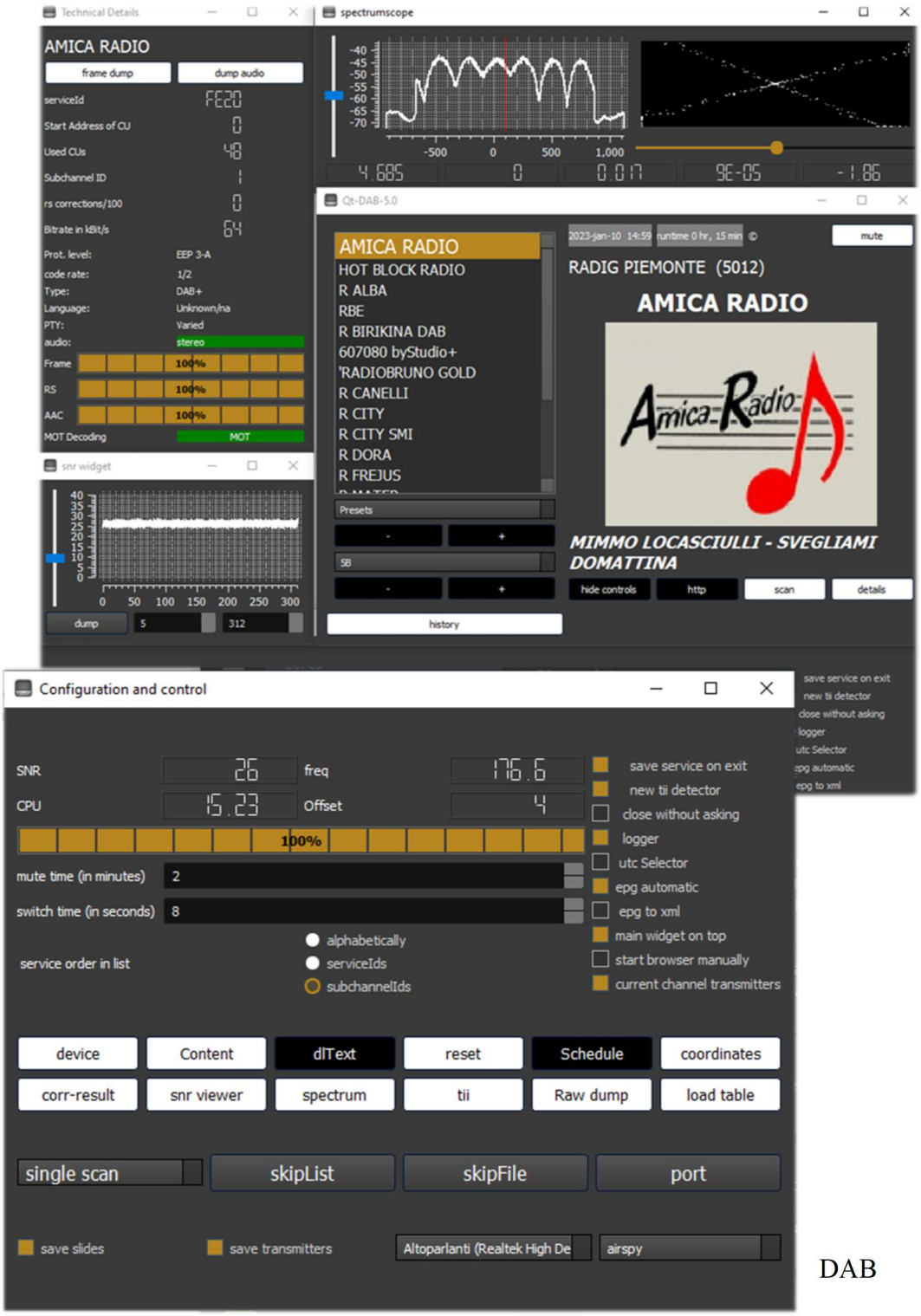
DAB / DAB+ (part 2)
software WELLE.IO

It is an open source SDR (for Windows10, Linux, macOS, Android), with support for Airspy (R2/Mini), RTL-SDR, SoapySDR. It supports high DPI resolutions including touch screen displays and also works on cheap mini computers like the Raspberry Pi 2/3 and various tablets / smartphones.



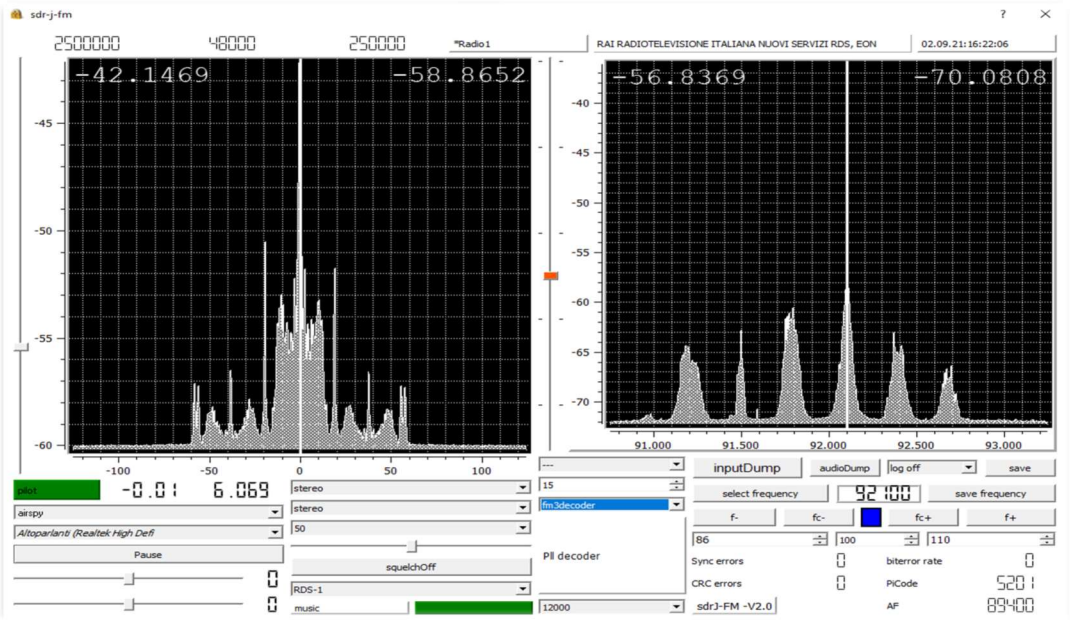
Suite DAB / FM (and spectrum displays) software Qt-DAB v5.0

This is a rich suite of different open source SDR programmes (for Windows and Linux OS) for receiving FM, DAB/DAB+, etc.
Support is for Airspy, HackRF, Lime, Pluto, RTL-SDR and SDRplay. + and the second is of FM reception.

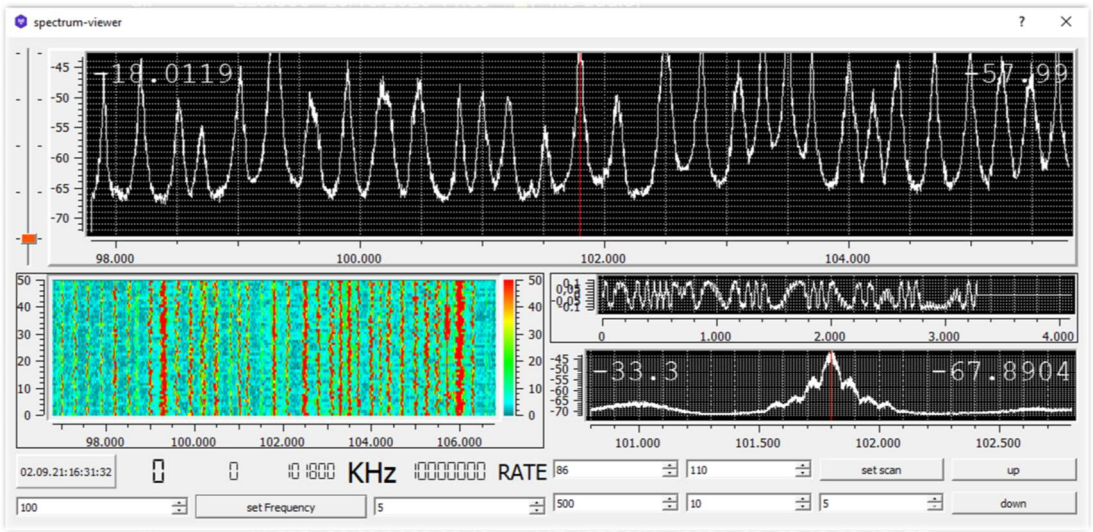


DAB

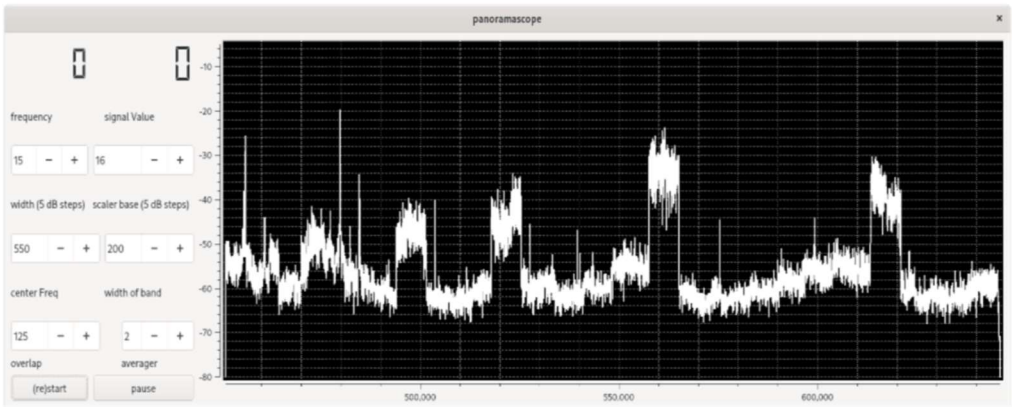
FM



A **Spectrum-viewer** also completes the equipment



and the **Panoramascope**



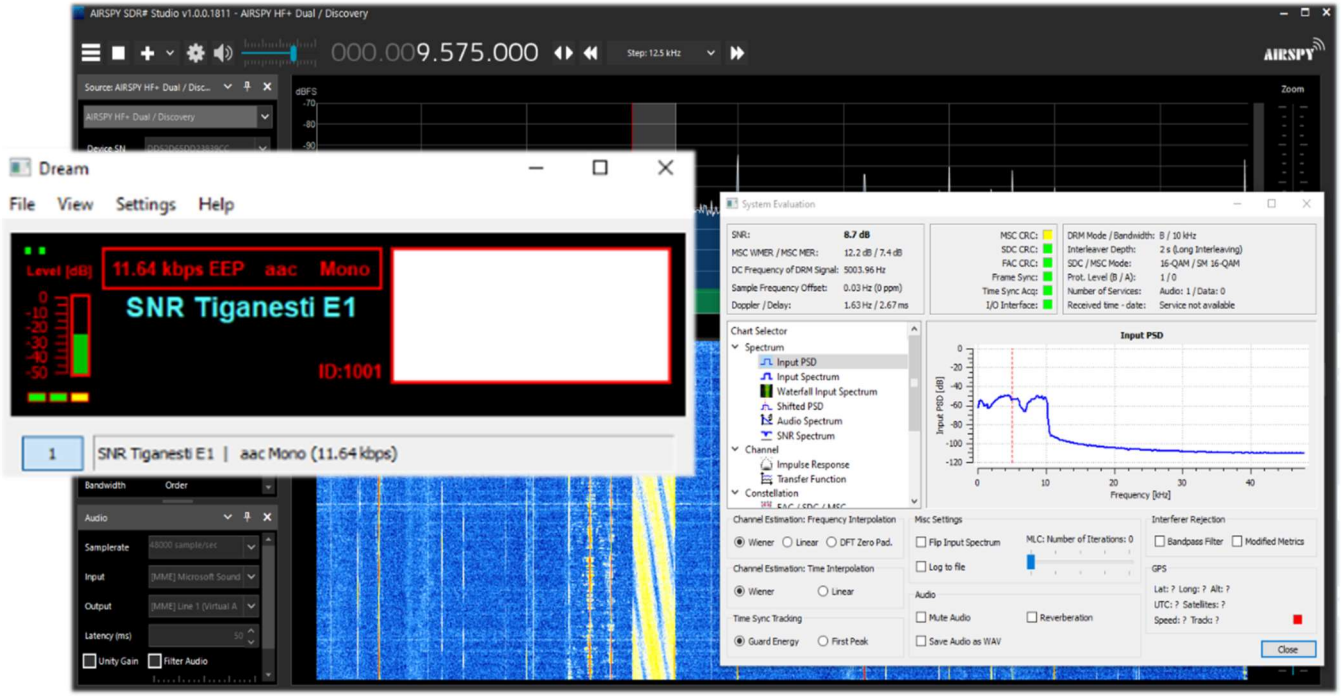
Link here: ***https://www.sdr-j.tk/index.html***

DRM in HF
Airspy HF+ Discovery and DReaM software

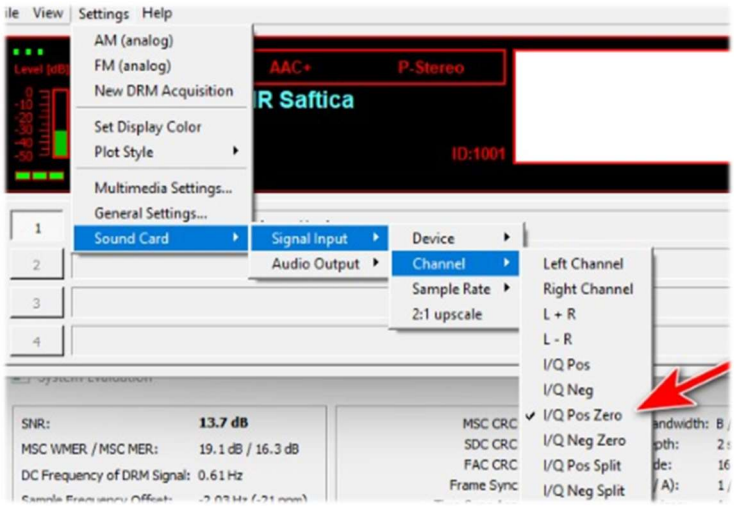
An Airspy HF+ Discovery makes it easy to listen the DRM (often multichannel) broadcasts in high quality with the free DReaM software that can be found free here:

https://sourceforge.net/projects/drm/

Digital Radio Mondiale or DRM is the only non-proprietary worldwide digital broadcasting system planned for HF capable of using the same frequencies currently allocated to the AM broadcasting service. In years past there were many more broadcasters in this now declining mode...



We recommend using the RAW mode in SDR#, select the sample rate at 768 ksps and perhaps increase the audio latency a bit.
Make sure AGC is enabled and set to a reasonable value: -50 dBFS is a good start.



In DReaM (a recent version of the software such as v2.2.1 and the appropriate VAC is recommended) you need to set "I/Q Pos Zero" as shown on the screen.

Read DTMF ...without a decoder!
Software Audacity

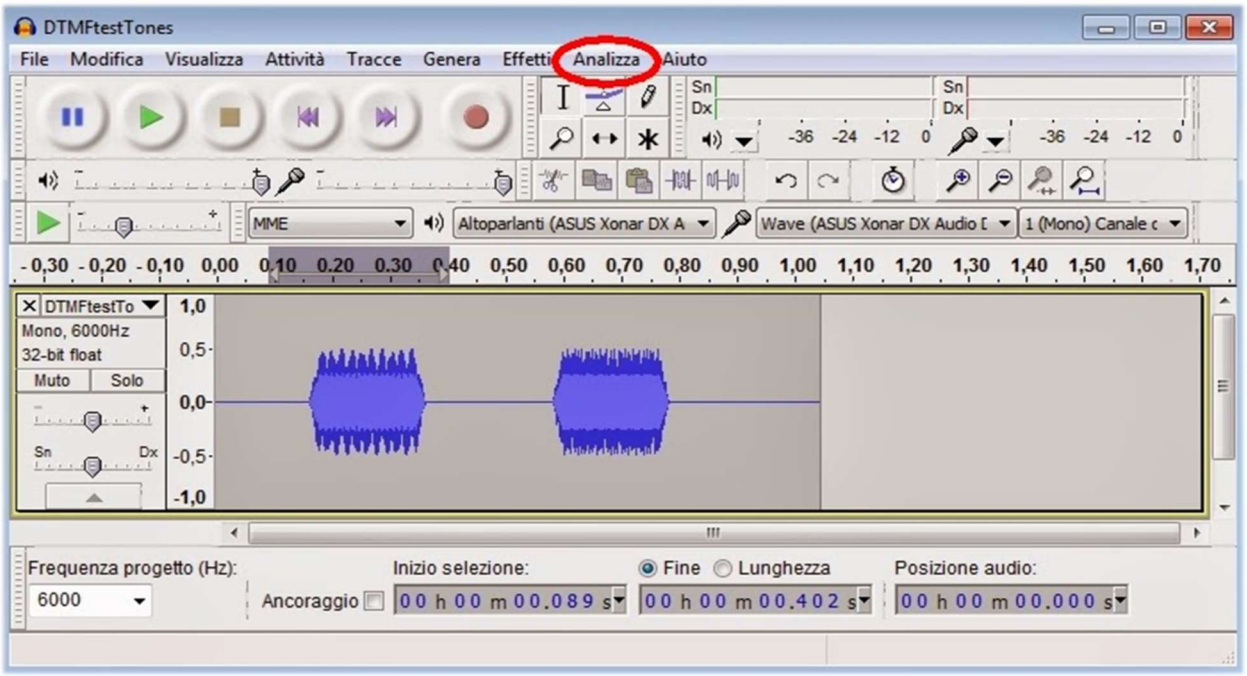
We do not always have a decoder to detect DTMF tones such as the one shown above. So let's see how to easily identify DTMF frequencies (in hertz), which is a coding system created for telephony, at Bell laboratories, to encode numerical codes in the form of sound signals in the audio band.

1	2	3	A	697 Hz
4	5	6	B	770 Hz
7	8	9	C	852 Hz
*	0	#	D	941 Hz
1209 Hz	1336 Hz	1477 Hz	1633 Hz	

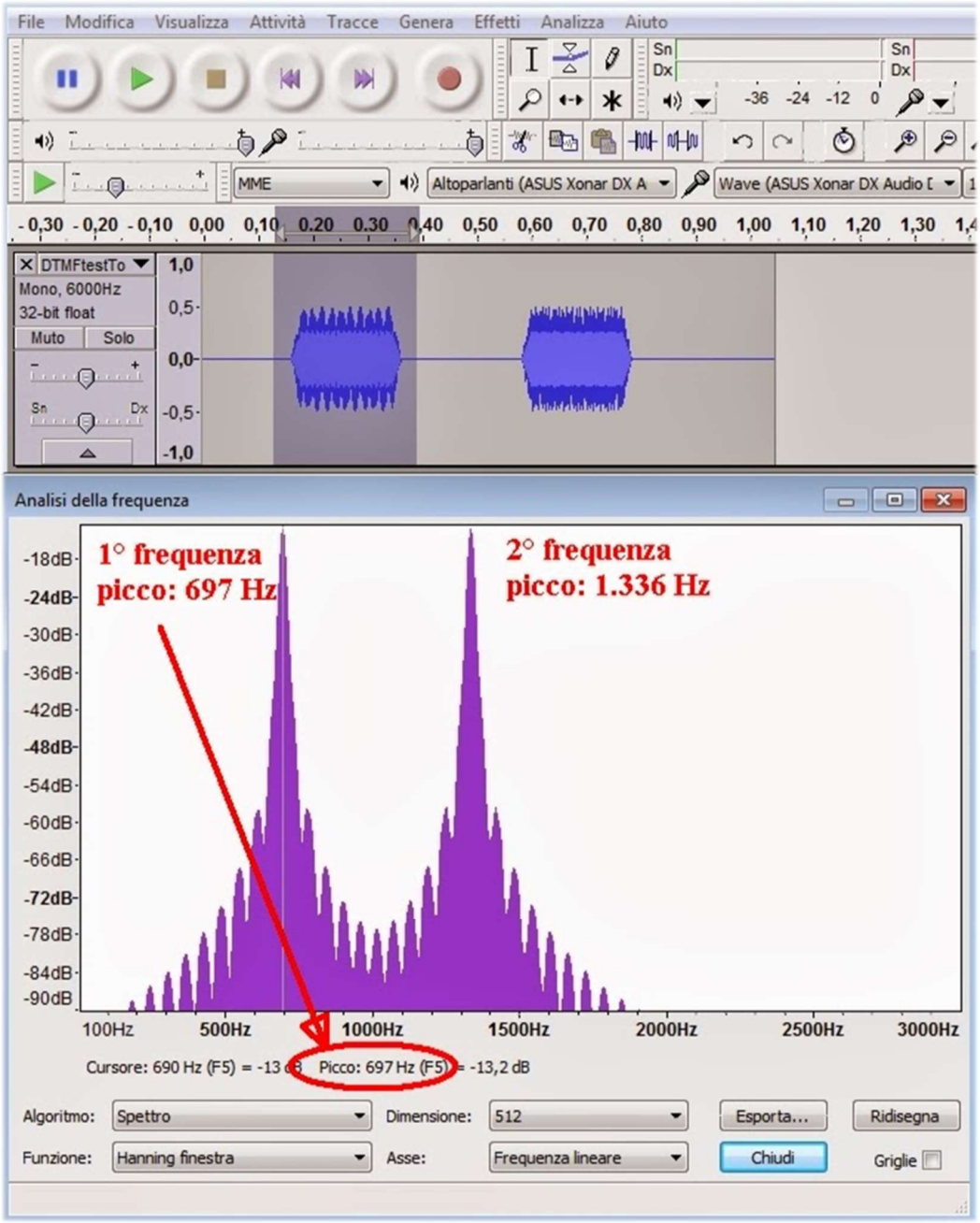
The DTMF keyboard consists of a 4×4 = 16-position matrix, where the row represents a low frequency and the column represents a high frequency. For example, pressing the 2 key generates two sine waves at frequencies of 697 Hz and 1336 Hz.

Rather than using 16 different frequencies for the 16 numbers/letters on the keyboard, 8 different frequencies have been used, with 2 being associated to each key. The term multi-frequency is therefore derived from the simultaneous use of two audio tones.

The frequencies have been allocated appropriately and with good intrinsic safety.
So to start it is necessary to save a WAV file from our SDR and analyse it for example with the freeware software Audacity.



Load the wave file, select the first portion of the DTMF signal, go to the menu "Analyze" and then "Show spectrum" where the program will perform the analysis of the frequencies.



In this window we will position ourselves on the two peak frequencies, reading down the two frequencies at 697 Hz and at 1336 Hz, which from the previous table correspond in fact to the number "2".

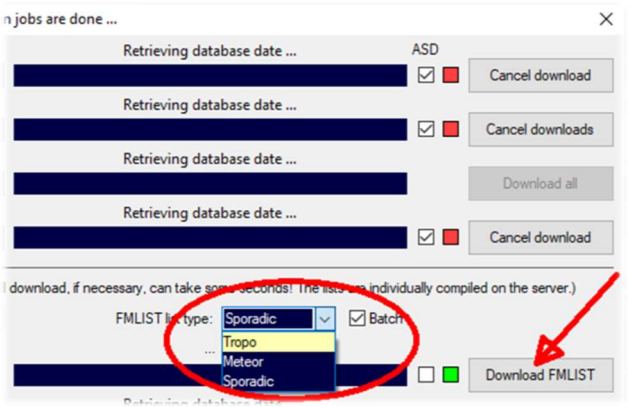
Then move to the second audio portion and repeat the analysis.

FM & FM-DX
Airspy R2/HF+ Discovery and CSVUB

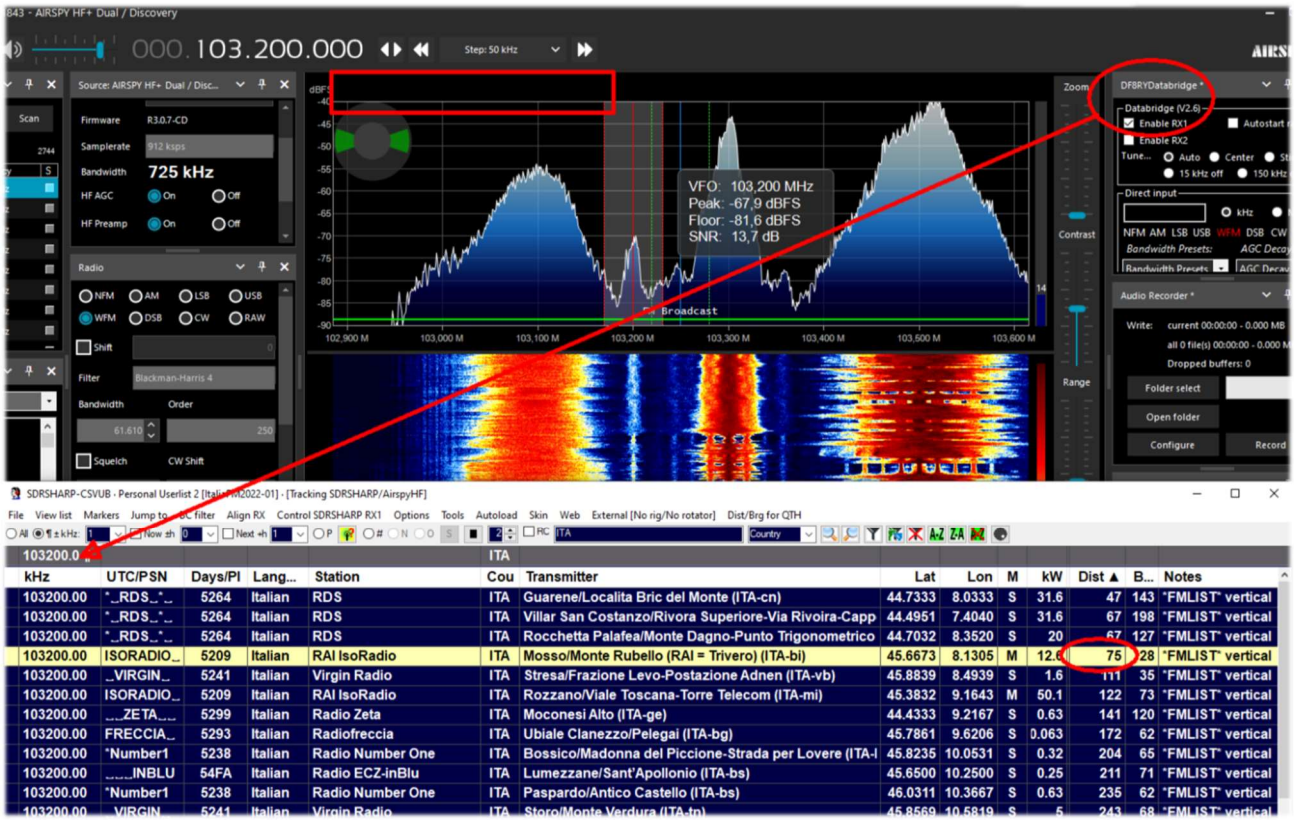
I have already written CSVUB extensively in the previous Plugins section, but this time I want to illustrate another of its excellent features in managing FMLIST databases:

https://fmscan.org/index.php

Certainly of interest to all FM and FM-DX enthusiasts, let's look at a few things...
Starting the program we go to TOOL / QTH MANAGER to enter our geographical coordinates, then to WEB / DOWNLOADER-CONVERTER to download one of the three lists or all together by flagging "Batch".
Now that we have the updated archives we can load them and use them to make searches and filters in conjunction with SDR# using the "DF8RYDataBridge" plugin mentioned above.

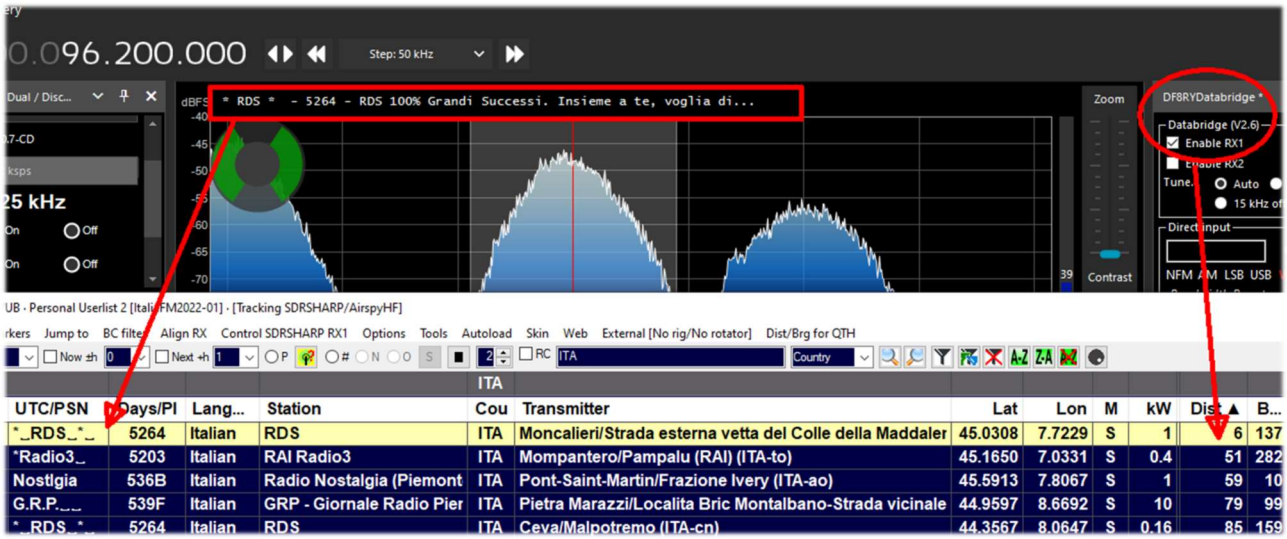


The tool can be useful for example to identify some distant and interfered signal that does not have the possibility to carry RDS (as in the example below at 103.200 MHz, where RDS is completely absent and the small signal appears between two powerful big-powers). In CSVUB, automatically hooked the frequency of the VFO, I sorted the database on the column "DIST" (distance in kilometers from my QTH) by clicking on it while holding down the CTRL key, will appear a little black triangle like this **Dist ▲**

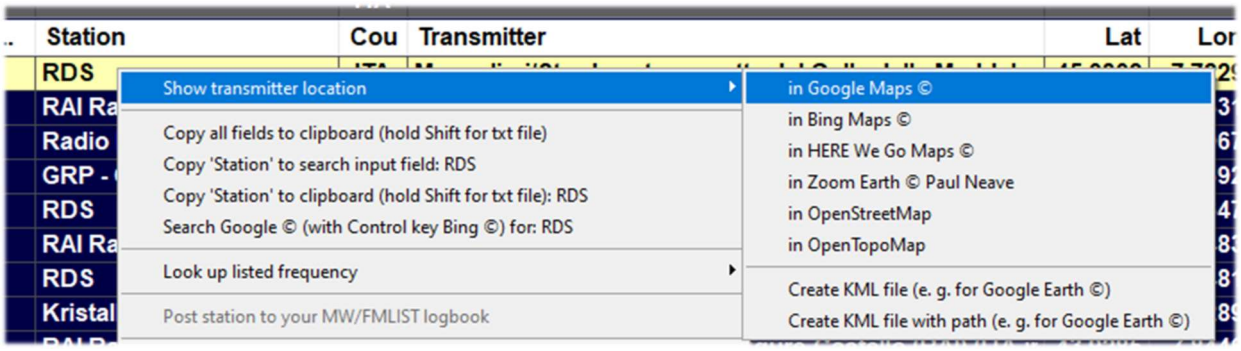


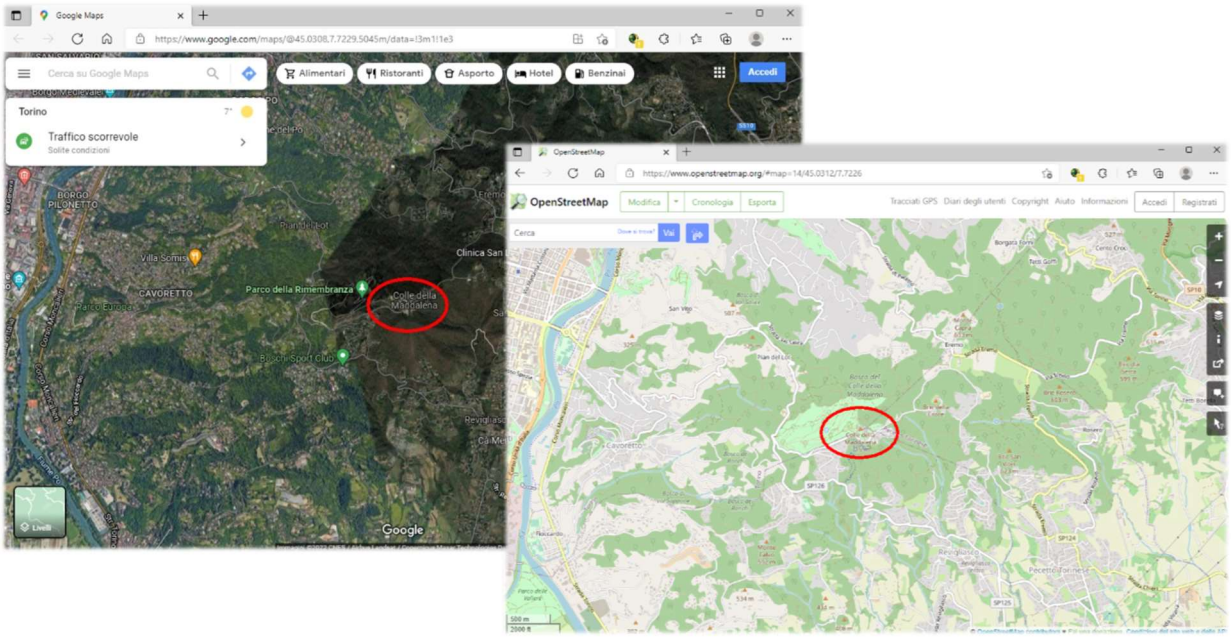
kHz	UTC/PSN	Days/PI	Lang...	Station	Cou	Transmitter	Lat	Lon	M	kW	Dist ▲	B...	Notes
103200.00	*_RDS_*_	5264	Italian	RDS	ITA	Guarene/Localita Bric del Monte (ITA-cn)	44.7333	8.0333	S	31.6	47	143	*FMLIST* vertical
103200.00	*_RDS_*_	5264	Italian	RDS	ITA	Villar San Costanzo/Rivora Superiore-Via Rivoira-Capp	44.4951	7.4040	S	31.6	67	198	*FMLIST* vertical
103200.00	*_RDS_*_	5264	Italian	RDS	ITA	Rocchetta Palafea/Monte Dagno-Punto Trigonometrico	44.7032	8.3520	S	20	67	127	*FMLIST* vertical
103200.00	ISORADIO_	5209	Italian	RAI IsoRadio	ITA	Mosso/Monte Rubello (RAI = Trivero) (ITA-bi)	45.6673	8.1305	M	12.6	75	28	*FMLIST* vertical
103200.00	_VIRGIN_	5241	Italian	Virgin Radio	ITA	Stresa/Frazione Levo-Postazione Adnen (ITA-vb)	45.8839	8.4939	S	1.6	111	35	*FMLIST* vertical
103200.00	ISORADIO_	5209	Italian	RAI IsoRadio	ITA	Rozzano/Viale Toscana-Torre Telecom (ITA-mi)	45.3832	9.1643	M	50.1	122	73	*FMLIST* vertical
103200.00	_ZETA_	5299	Italian	Radio Zeta	ITA	Moconesi Alto (ITA-ge)	44.4333	9.2167	S	0.63	141	120	*FMLIST* vertical
103200.00	FRECCIA_	5293	Italian	Radiofreccia	ITA	Ubiale Clanezzo/Pelegai (ITA-bg)	45.7861	9.6206	S	0.063	172	62	*FMLIST* vertical
103200.00	*Number1	5238	Italian	Radio Number One	ITA	Bossico/Madonna del Piccione-Strada per Lovere (ITA-l	45.8235	10.0531	S	0.32	204	65	*FMLIST* vertical
103200.00	_INBLU	54FA	Italian	Radio ECZ-inBlu	ITA	Lumezzane/Sant'Apollonio (ITA-bs)	45.6500	10.2500	S	0.25	211	71	*FMLIST* vertical
103200.00	*Number1	5238	Italian	Radio Number One	ITA	Paspardo/Antico Castello (ITA-bs)	46.0311	10.3667	S	0.63	235	62	*FMLIST* vertical
103200.00	VIRGIN	5241	Italian	Virgin Radio	ITA	Storo/Monte Verdura (ITA-tn)	45.8569	10.5819	S	5	243	68	*FMLIST* vertical

Hearing by ear that it was the transmission "RAI ISORADIO", the first line highlighted by the database CSVUB reports in fact that the station received could be that distant 75 km with 12 kW of power...
Much easier is the case that the RDS code is detected and with the immediate feedback of the PI code we have a certain and precise identification on the database of the stations present in CSVUB!



In the example above we can see for the station tuned to 96.200 MHz, in CSVUB: name of the station "RDS-Radio Dimensione Suono", its PI code "5264", transmitter site information, power in kW, geographical coordinates, distance from my QTH and the bearing in degrees if you have an antenna rotor connected to your system.
Clicking then on the highlighted record, with the right mouse button, opens a specific menu that allows you to view the site of the transmitter with various graphics and details.





Think about the possibility of easily identifying during FM-DX sessions, distant stations that reach us only thanks to propagation or summer phenomena of E-sporadic…

WARNING:

In the last quarter of 2023, the creators of the "fmscan.org" website decided to provide RDS PI codes only to regularly registered users anymore. Therefore, those who do not have an account, the Downloader in CSVUB will no longer download PI codes.
In order to continue to receive the lists with PI codes (for strictly personal use and not transferable to third parties and/or unauthorized persons) this is the procedure to follow.

All users are also invited to cooperate by sending their listening logs to keep the common database up to date...

- Register with Email/Password and/or go to the site: ***www.fmlist.org***
- Go to the site: ***www.fmscan.org***
- Set up your QTH if not already done.
- Go down and click under "Tools (userlists etc.)"
- Click on "Userlists for Perseus/ELAD/Winradio/Stationlist/SDR Console".
- Click on the desired list (FM+Tropo, FM+ Meteor Scatter, Sporadic E).
- Choose "CSV format" and to the right "CSV Separator: Tab."
- Left-click on "DOWNLOAD userlist1.csv" and wait.
- Save the file by renaming it, for example, as "userlist-FMtropo.txt."
- Now you can load it into CSVUB with "Open CSV userlist (1...30)."
- To prevent this list from being inadvertently overwritten by a new version without PI codes, the ASD (AutoStart Download) flag next to "Download FMLIST" in the Downloader must be removed.

This procedure must be performed manually each time a new list is available.

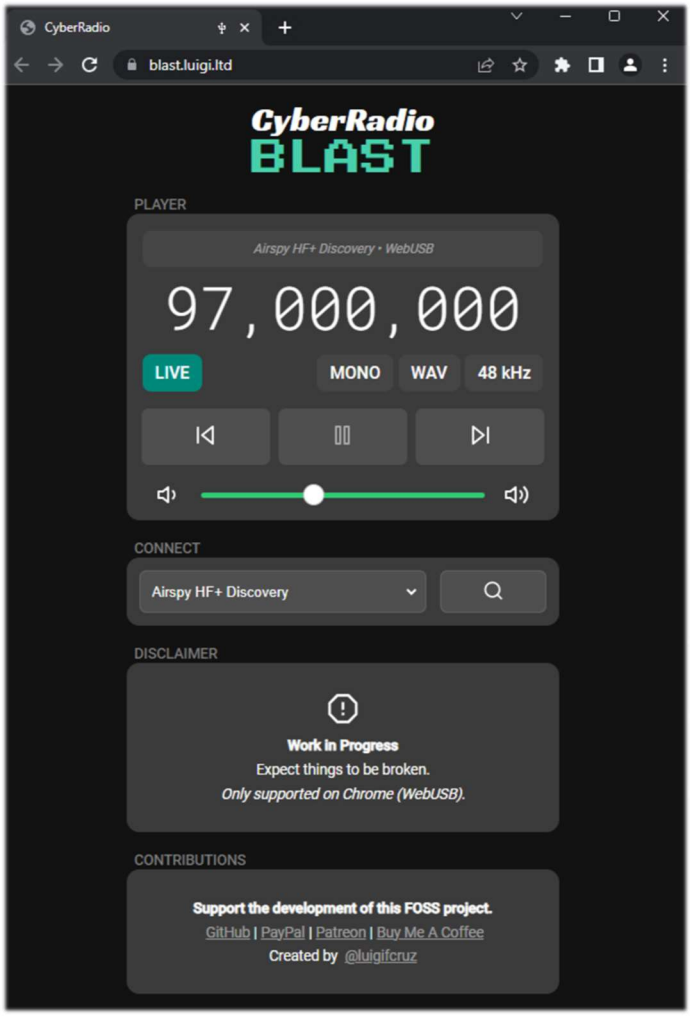
The automatic download of FMSCAN and AMSCAN lists was blocked by the owners of fmscan.org in **January 2024** referring to the fact of data theft by bots. CSVUserlistBrowser is considered to be a bot. AMSCAN und FMSCAN lists must be downloaded manually directly from the fmscan.org website and only for registered users and active contributors

FM … directly from the browser!
Airspy HF+ Discovery and Chrome

This is a really nice new feature, the brainchild of Luigi Cruz (PU2SPY). In this very first version you can listen to your FMW stations directly in the Chrome browser without any software installation.

Simply log on to the link of "CyberRadio BLAST": **blast.luigi.ltd**

In the "Connect" field we select our Airspy HF+ Discovery and this that what will appear. At the moment only in Mono, no RDS or BW filters, but this is just the beginning: think what can be done in a while!

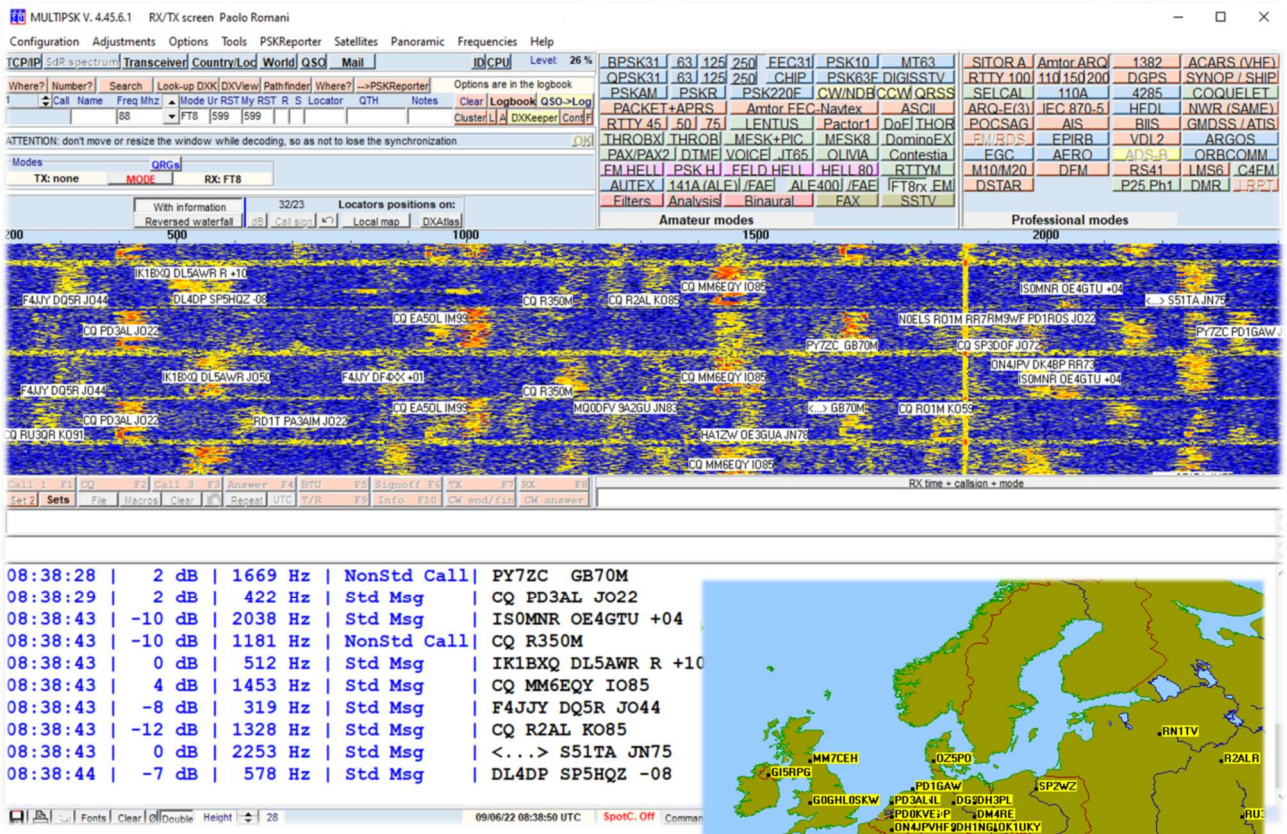


The development is based on its library "webusb-libusb" powered by WebAssembly and Emscripten. The project is the first translations layer from libusb to webusb with the aim of supporting most SDR libraries directly within a browser.

For upcoming updates and implementations let's keep an eye on GitHub, thanks Luigi!!!: **https://github.com/luigifcruz/CyberRadioBlast**

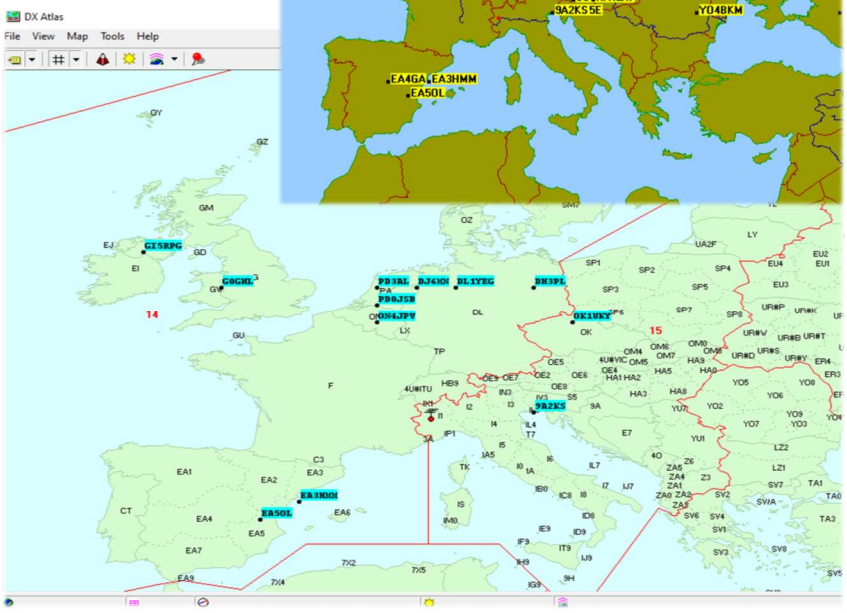
FT8
Airspy HF+ Discovery and MultiPSK

The indefatigable Patrick Lindecker (F6CTE) has recently released a beta of his MultiPSK v4.45.6.1, which now also covers decoding of the FT8 ham radio protocol born in 2017 by Joe Taylor (K1JT) and Steve Franke (K9AN). The name comes from "Franke-Taylor design, 8-FSK modulation."
Designed for "multi-hop Es where signals may be weak and fading, openings may be short, and you want to complete reliable and confirmable QSOs quickly," it has now supplanted previous systems and is present in all radioamateur bands.
In this screen SDR# HF+ Discovery was tuned to 14.074 kHz in USB, and MultiPSK was working in conjunction with the VAC.



Note on the MultiPSK waterfall the labels of callsigns just received and taken from the tabular part below full of other information!

It is also possible to view on an internal map or "DX Atlas" the ratio (dB) and locations of received stations, to check what the directivity of one's own antenna is (and the gain compared to a SWL or neighbor).



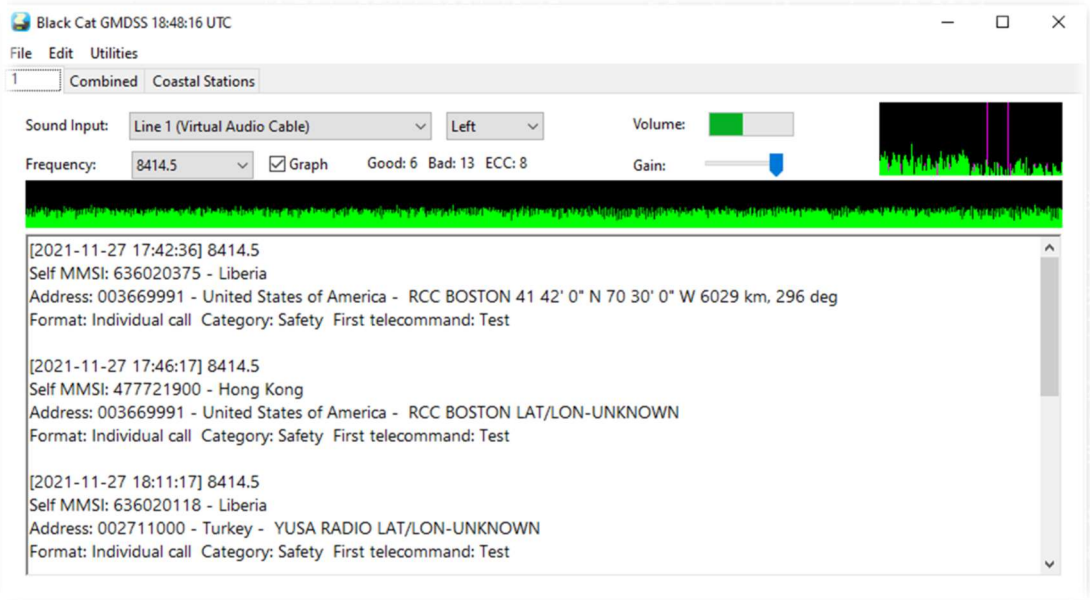
GMDSS, a multi-channel decoder
Airspy HF+ Discovery and Black Cat GMDSS decoder

Black Cat GMDSS is a first multi-channel GMDSS HF decoder with new concept compared to previous radioamateur decoders that will surely make people talk!

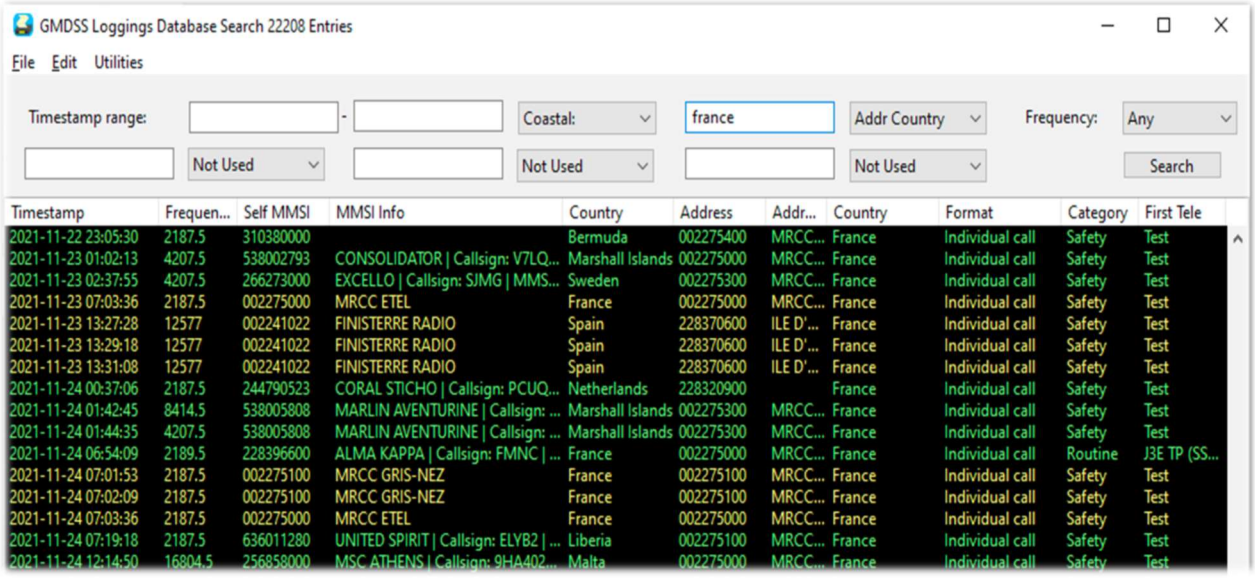
https://blackcatsystems.com/software/black_cat_gmdss_decoder.html

Up to 8 decoders can run at the same time for all GMDSS channels provided by the worldwide system at 2187.5, 4207.5, 6312, 8414.5, 12577, 16804.5 kHz.

Each decoder can be connected to its own audio input source (a virtual audio device or physical sound input device).

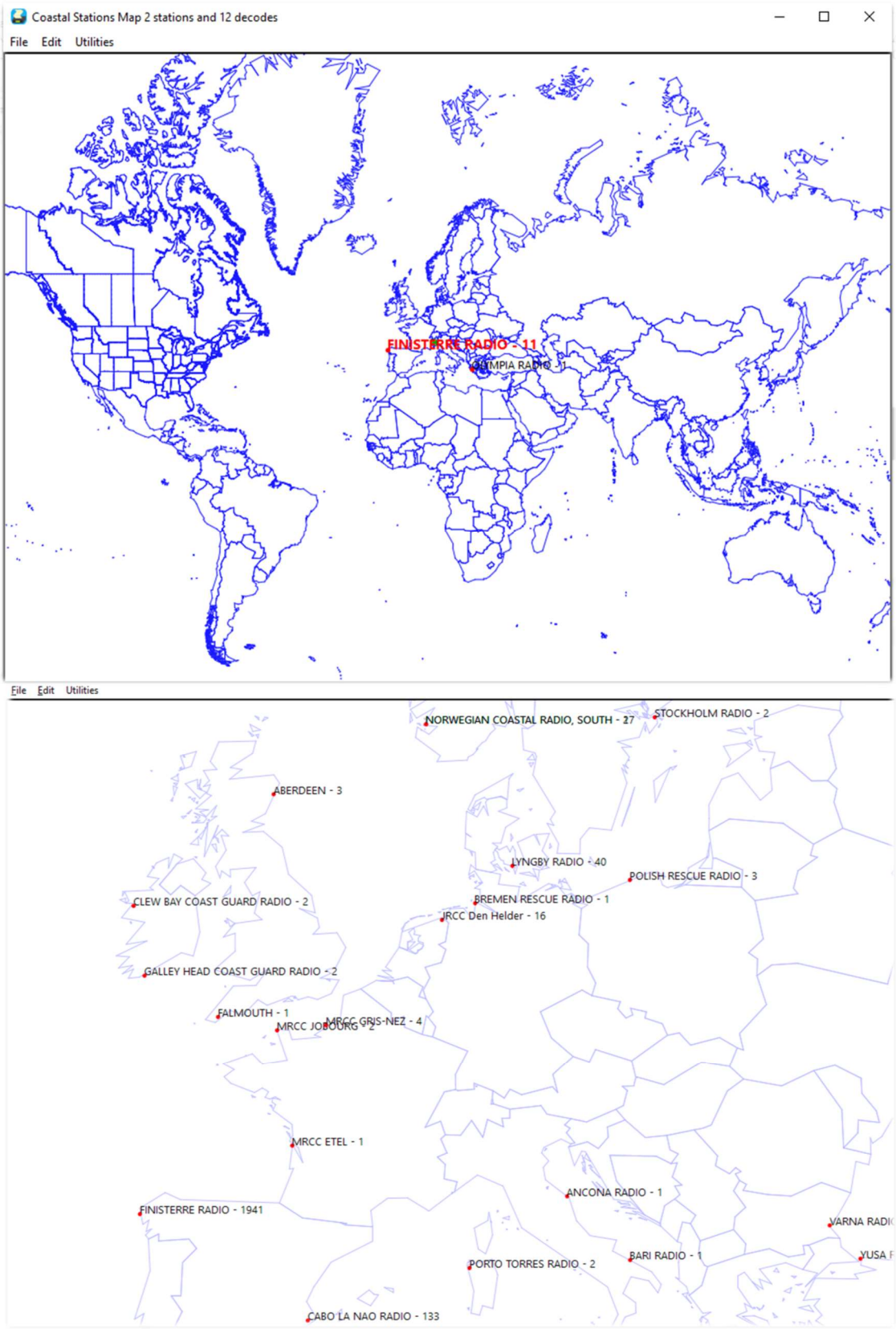


One of the distinctive features of the decoder is that it can decode directly from a WAV audio file. Multiple files can be selected, they will decode one after another.
Decoding of WAV files is much faster than real time decoding, limited by the speed of your computer, often 10x real time processing.
There are also a number of useful tools for map visualization and online search on the MMSI database.



Timestamp	Frequen...	Self MMSI	MMSI Info	Country	Address	Addr...	Country	Format	Category	First Tele
2021-11-22 23:05:30	2187.5	310380000		Bermuda	002275400	MRCC...	France	Individual call	Safety	Test
2021-11-23 01:02:13	4207.5	538002793	CONSOLIDATOR | Callsign: V7LQ...	Marshall Islands	002275000	MRCC...	France	Individual call	Safety	Test
2021-11-23 02:37:55	4207.5	266273000	EXCELLO | Callsign: SJMG | MMS...	Sweden	002275300	MRCC...	France	Individual call	Safety	Test
2021-11-23 07:03:36	2187.5	002275000	MRCC ETEL	France	002275000	MRCC...	France	Individual call	Safety	Test
2021-11-23 13:27:28	12577	002241022	FINISTERRE RADIO	Spain	228370600	ILE D'...	France	Individual call	Safety	Test
2021-11-23 13:29:18	12577	002241022	FINISTERRE RADIO	Spain	228370600	ILE D'...	France	Individual call	Safety	Test
2021-11-23 13:31:08	12577	002241022	FINISTERRE RADIO	Spain	228370600	ILE D'...	France	Individual call	Safety	Test
2021-11-24 00:37:06	2187.5	244790523	CORAL STICHO | Callsign: PCUQ...	Netherlands	228320900		France	Individual call	Safety	Test
2021-11-24 01:42:45	8414.5	538005808	MARLIN AVENTURINE | Callsign: ...	Marshall Islands	002275300	MRCC...	France	Individual call	Safety	Test
2021-11-24 01:44:35	4207.5	538005808	MARLIN AVENTURINE | Callsign: ...	Marshall Islands	002275300	MRCC...	France	Individual call	Safety	Test
2021-11-24 06:54:09	2189.5	228396600	ALMA KAPPA | Callsign: FMNC | ...	France	002275000	MRCC...	France	Individual call	Routine	J3E TP (SS...
2021-11-24 07:01:53	2187.5	002275100	MRCC GRIS-NEZ	France	002275100	MRCC...	France	Individual call	Safety	Test
2021-11-24 07:02:09	2187.5	002275100	MRCC GRIS-NEZ	France	002275100	MRCC...	France	Individual call	Safety	Test
2021-11-24 07:03:36	2187.5	002275000	MRCC ETEL	France	002275000	MRCC...	France	Individual call	Safety	Test
2021-11-24 07:19:18	2187.5	636011280	UNITED SPIRIT | Callsign: ELYB2 | ...	Liberia	002275100	MRCC...	France	Individual call	Safety	Test
2021-11-24 12:14:50	16804.5	256858000	MSC ATHENS | Callsign: 9HA402...	Malta	002275000	MRCC...	France	Individual call	Safety	Test

Those who have had the opportunity to try and test it in comparison with other GMDSS decoders says the best accuracy in decoding, with fewer errors than all competitors. In addition, not least, is the extreme lightness in terms of CPU usage (in some cases even 5 times less than others!)

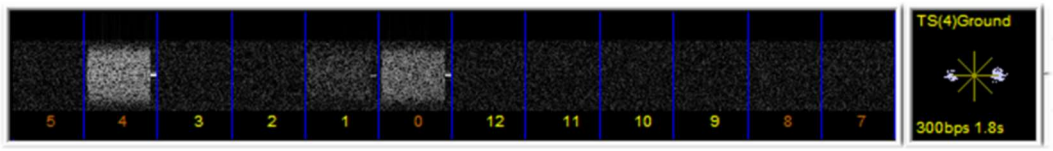


I have write a PDF guide that can be downloaded here:
https://blackcatsystems.com/download/BlackCatGMDSSGuide.pdf

HFDL at 300 bps

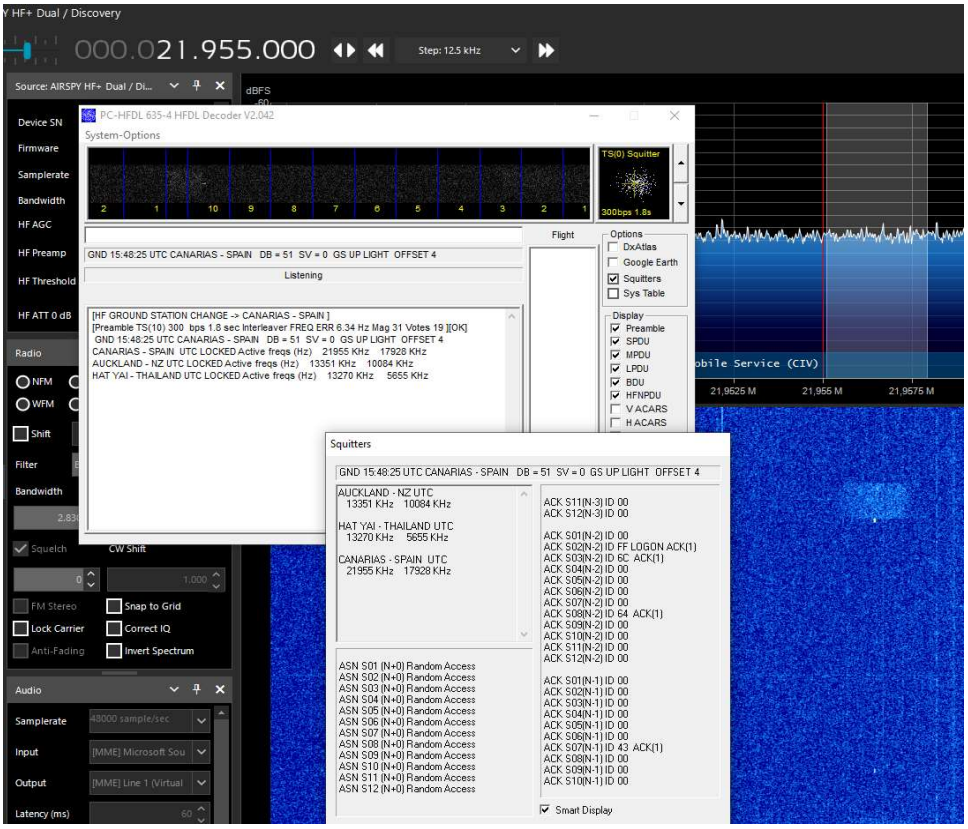
Airspy HF+ Discovery e decoder PC-HFDL

PC-HFDL is a windows based decoder for the ARINC 635-3 HF data-link protocol. Is based on a number of interconnected ground stations. Each ground station transmits a frame called a Squitter every 32 seconds (see screenshot). The Squitter frame informs aircraft of the system status, provides a timing reference and provides protocol control.



Each ground station has a time offset for its Squitters this allows planes to jump between ground stations when trying to log on to the best one. When passing traffic Time division Multiplexing is used (TDMA) this prevents two aircraft transmitting at the same time causing collisions.
The program uses the system table (now version 51) to determine the frequencies being used are. This information is transmitted by the HFDL groundstations.

Net of current worldwide stations: AGANA – GUAM, AL MUHARRAQ – BAHRAIN, ALBROOK – PANAMA, AUCKLAND – NEW ZEALAND, BARROW – ALASKA, CANARIAS – SPAIN, HAT YAI – THAILAND, JOHANNESBURG - SOUTH AFRICA, KRASNOYARSK – RUSSIA, MOLOKAI – HAWAII, MUAN - SOUTH KOREA, REYKJAVIK – ICELAND, RIVERHEAD - NEW YORK, SAN FRANCISCO – CALIFORNIA, SANTA CRUZ – BOLIVIA, SHANNON - IRELAND

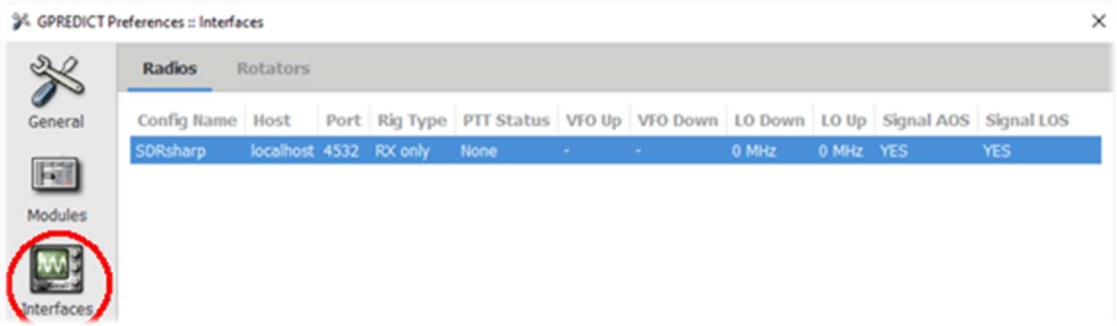


The “Squitters” display write information in a separate dialog box. “Smart display” shows next timeslot allocations.
This software has been designed as a decoder and does not carry out extensive logging or analysis of received information.

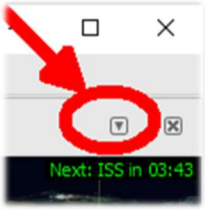
ISS reception and satellite tracking

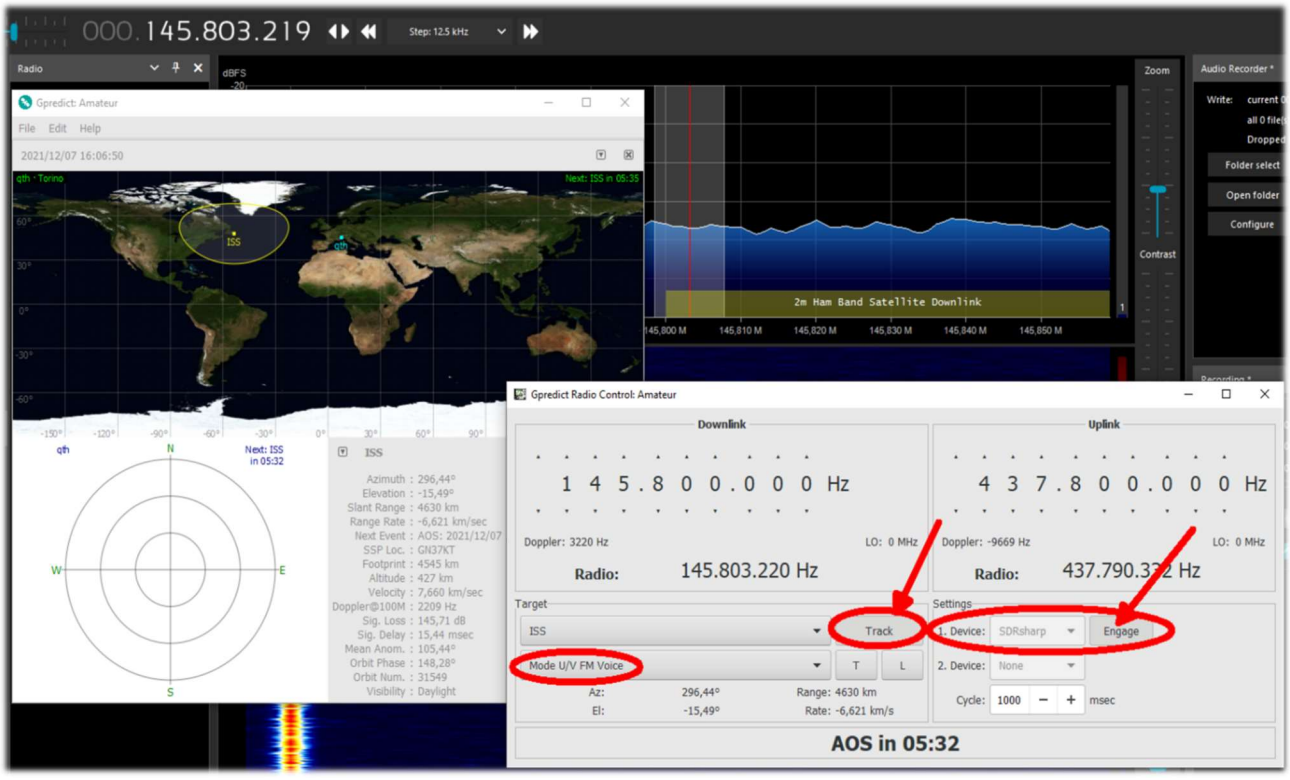
Airspy R2 + Gpredict & plugin Gpredict Connector

The ISS and other amateur radio satellites are not difficult to receive and it is sufficient even only a discone antenna or a vertical antenna for the 2 meters band... the most important thing is to use a good software for the calculation of satellite passages and their automatic tracking to compensate for the frequency shift due to the doppler effect in many cases very marked.
It will be necessary to look for and download the GPREDICT software (for example the release "gpredict-win32-2.3.37.zip") and install it...
For the first configuration: set your coordinates in EDIT / PREFERENCES / GENERAL / GROUND STATIONS while in INTERFACES / RADIOS you will have to create a line like the following for our SDRsharp with Localhost and port 4532:



For subsequent use: always update the TLE data in the EDIT / UPDATE TLE DATA FROM NETWORK menu or provide an automatic update in ED in EDIT / PREFERENCES / GENERAL / TLE UPDATE
To configure tracking: in Gpredict, click on the "Module options/ Shorcuts" icon (highlighted here at the side), select a satellite (in our case the ISS) from the CONFIGURE menu and then access the RADIO CONTROL panel to set some fields to confirm the type of traffic to be monitored (e.g. "Mode U/V FM VOICE") and then click on the "TRACK" and "ENGAGE" buttons...

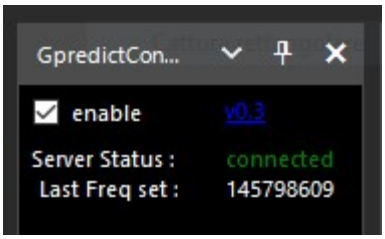




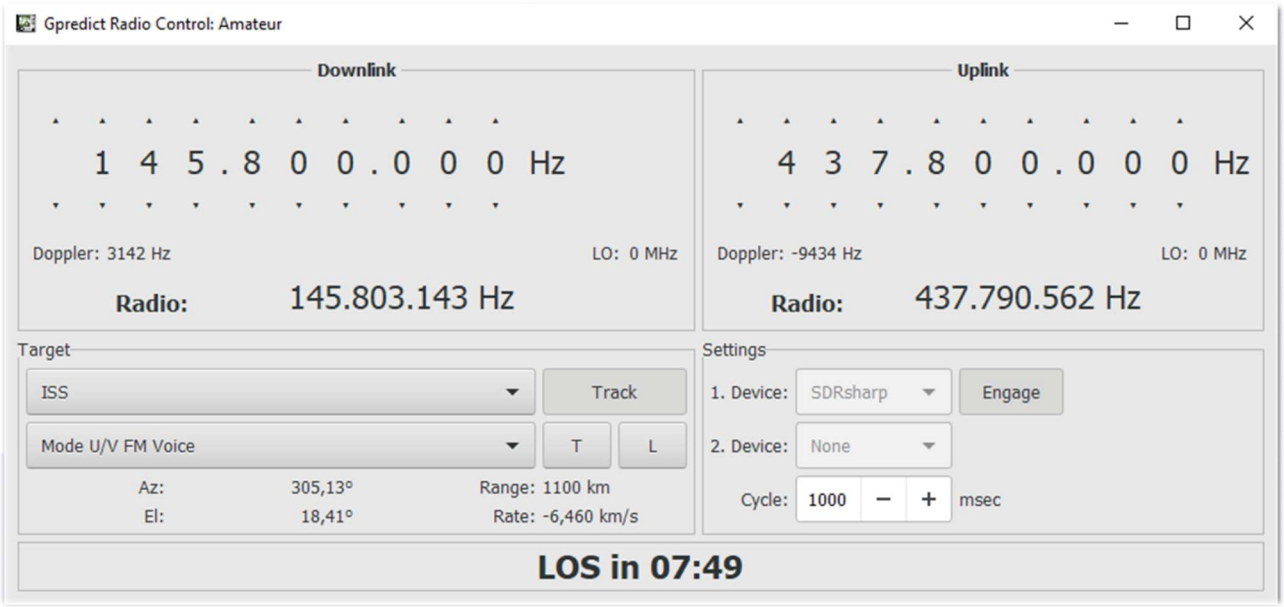
Let us now see on the SDR# side what needs to be done.

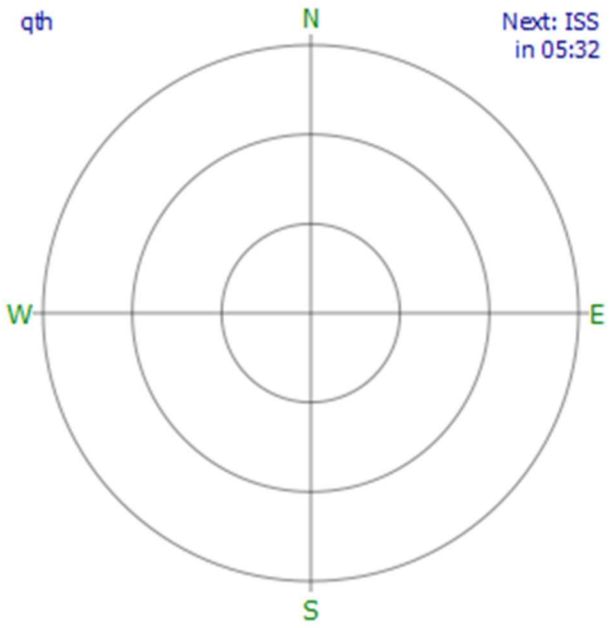
We will use the free plugin "GpredictConnector", which can be downloaded here:
https://github.com/alexwahl/SDRSharp.GpredictConnector

By extracting the DLL in the usual directory, it enables automatic dialogue with Gpredict as soon as the “enable” flag is enabled.



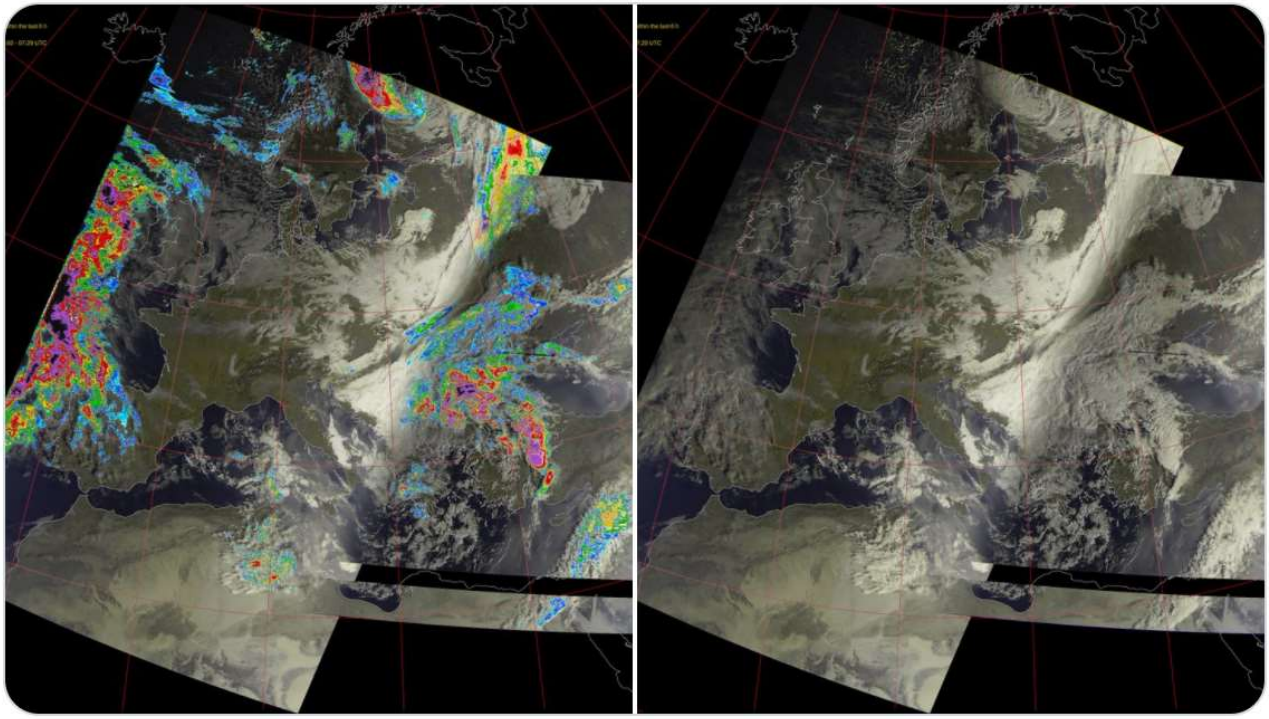
If everything is working properly the plugin will connect to port 4532 and "Server Status" will not only display "connected" in green colour, but SDR# will and autonomously follow the downlink frequency of the satellite with appropriate VFO changes to compensate for the doppler effect, as well as providing multiple tracking information and AOS and LOS times.





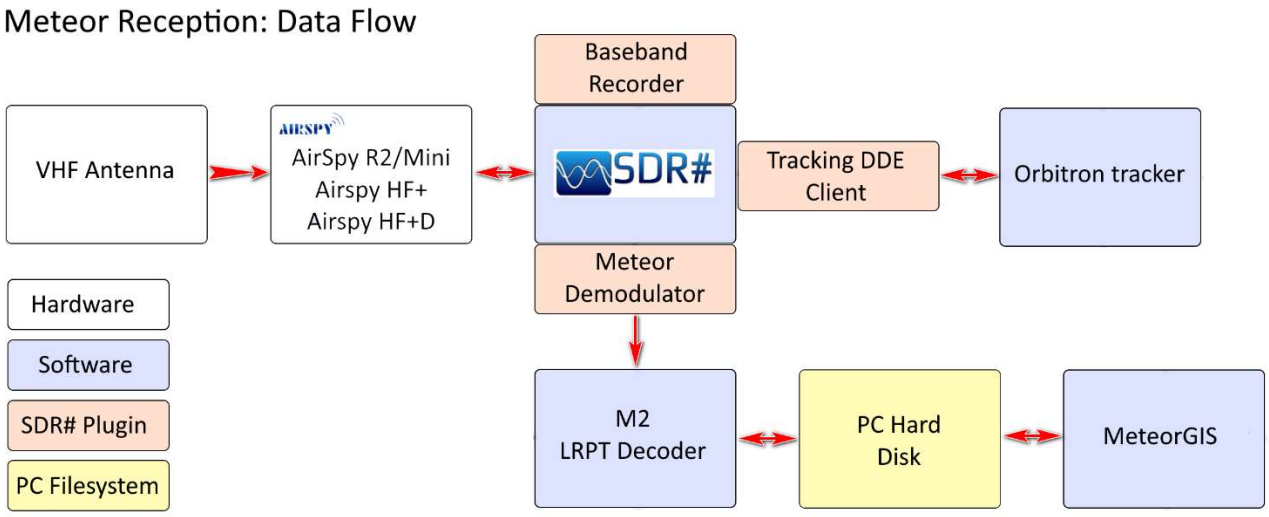
METEOR-M N2 Reception of quality images
Airspy HF+ Discovery and many software

It is not easy to condense the whole process in a few lines, but the final result of Marco Melandri (BlackApple62) in receiving images from the METEOR-M N2 satellite is definitely of very high



quality...

LRPT images received at 137.1 MHz in RGB+Rainfalls/RGB mode with Turnstile antenna and SPF5189 RF Low Noise Amplifier. These are the software used: Tracking DDE v1.2 + Meteor Demodulator v2.3 + LRPT decoder v2019.9.14.0056 + Postprocessor MeteorGIS v2.24.



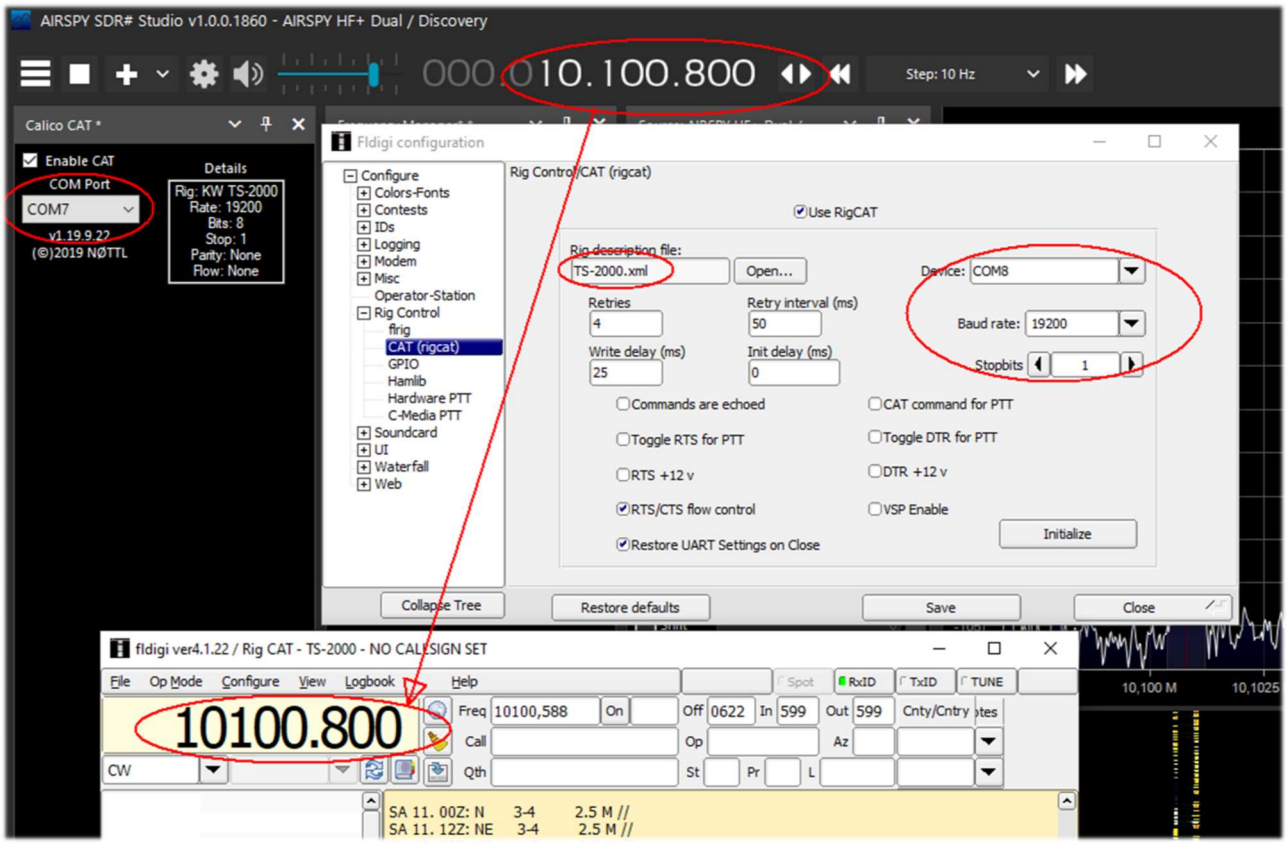
The "data flow" kindly granted to me, represents in a very simplified way how data pass from radio reception to decoded images on disk. A more detailed description of the events in each software module would be needed, starting from the AOS phase of the satellite, up to the LOS and finally to the writing of the processed images, but it would take a dedicated guide, so for those interested this is an introduction: http://happysat.nl/Setup_Meteor/Setup.html

Modem multimode
Airspy HF+ Discovery, software Fldigi

Fldigi (short for Fast light digital) is a free program that allows a computer sound card to be used as a two-way data modem. The software is used by radio amateurs all over the world on both HF and V-UHF (for the fastest modes) even with only a few watts of RF power. Many modes are supported: CW, Contestia, DominoEX, Hell, MFSK, OFDM, Olivia, PSK, QPSK, 8PSK, RTTY, THOR, Wefax, Navtex/Sitor-B, etc.

You will need to install a CAT management plugin e.g. CalicoCat and configure it as mentioned in the Plugins section above (in my case on port COM7).

Now Fldigi has to be installed and configured as in the screenshot: I downloaded the specific Rig file "TS-2000.xml", enabled the "Use RigCAT", assigned the COM8 port to 19200 baud, 1 Stopbit.



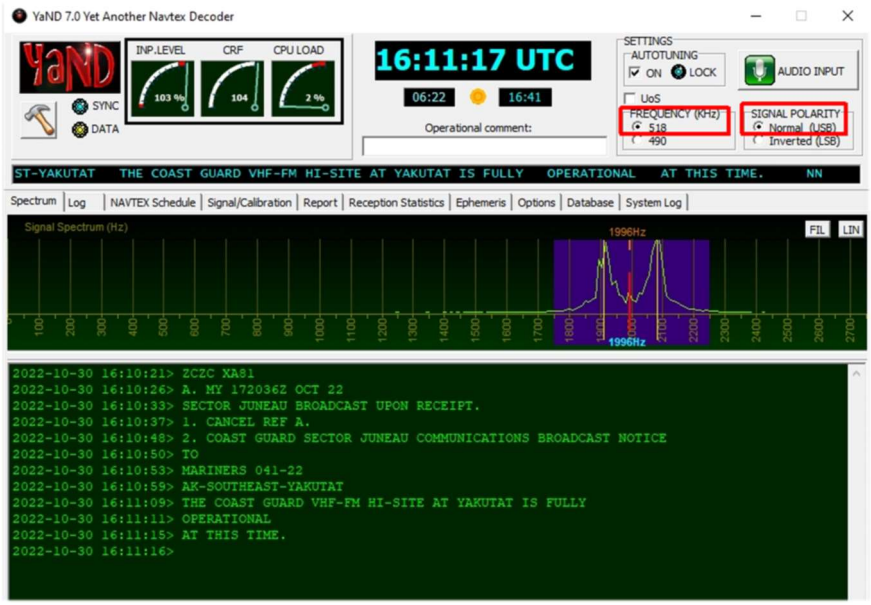
In the meantime, the CalicoCat plugin will make the two softwares talk to each other, and any change of VFO (or change of emission mode) in one of the two softwares will be reflected in the other... *However, I have encountered an annoying bug that causes SDR# to crash immediately: just put FSK mode in Fldigi. Therefore, try the other plugin "SerialController"...*

In the previous example, the RTTY-ITA2 50 baud signal from station DDK9 Hamburg Meteo is decoded (via VAC) on 10100.80 kHz (note the two VFOs perfectly aligned!). In SDRsharp remember to use the "CW" mode.

http://www.w1hkj.com/

NAVTEX (NAVigational TEXt messages)
Airspy HF+ Discovery, software YAND / FRISNIT decoder

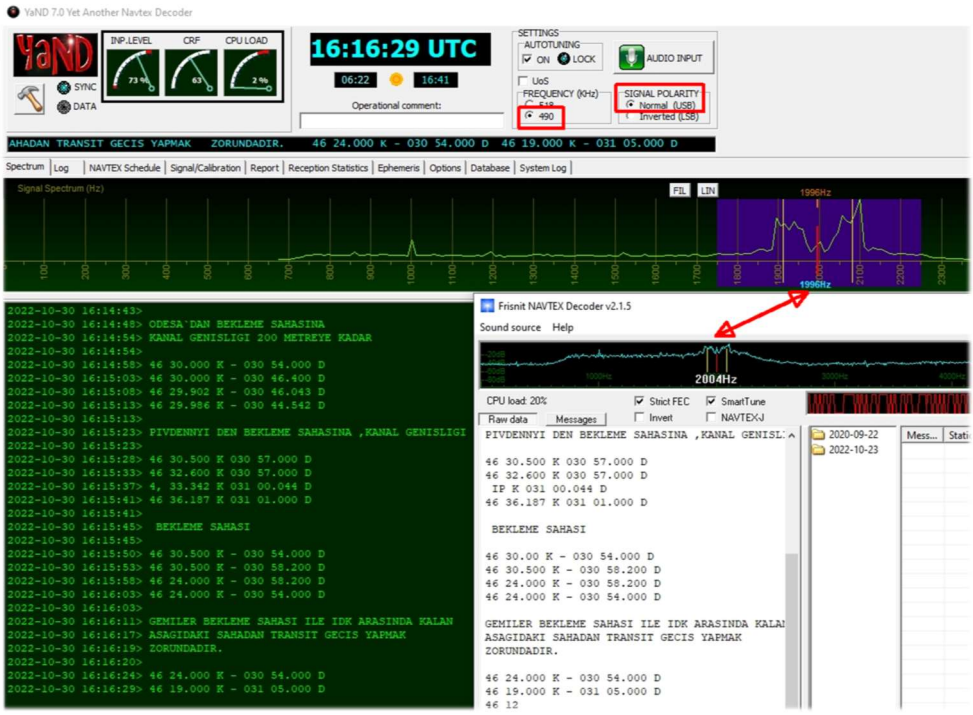
The service, I believe well known to most, has long been developed to broadcast free of charge on medium waves, navigational bulletins and weather information for the use of ships, vessels and those who go to sea. There are two types: the international broadcast on 518 kHz (in English) and the national on 490 kHz (in local/regional languages). These services use the frequencies indicated at certain times with transmissions made by fixed stations in a particular area (called NAVAREA) because the globe has been divided into areas: for example, Italy, in the Mediterranean, belongs to NAVAREA III.



Technically, the modulation is of the BFSK (frequency-shift keying binary) type at the rate of 100 baud and shift 170 Hz.

Taking advantage of a remote SpyServer in Canada I tuned to 516 kHz (in USB) to have the center frequency at 2000 Hz in the "YAND v7.0" software (set as indicated in the red boxes and collaged via virtual sound card). Those who want to use LSB the tuning frequency will be 520 kHz.

In this example, with SpyServer in the Nordic countries, I received the Navtex signal in the regional language, tuning to 488 kHz (in USB) with "YAND" and another excellent decoder named "FRISNIT NAVTEX Decoder". Given the excellent level of the received signal, the results of the two decoders were practically equal.



I have read online that this system too, after decades of honored activity, may soon be replaced by a new digital mode, which could place images and new services alongside text.

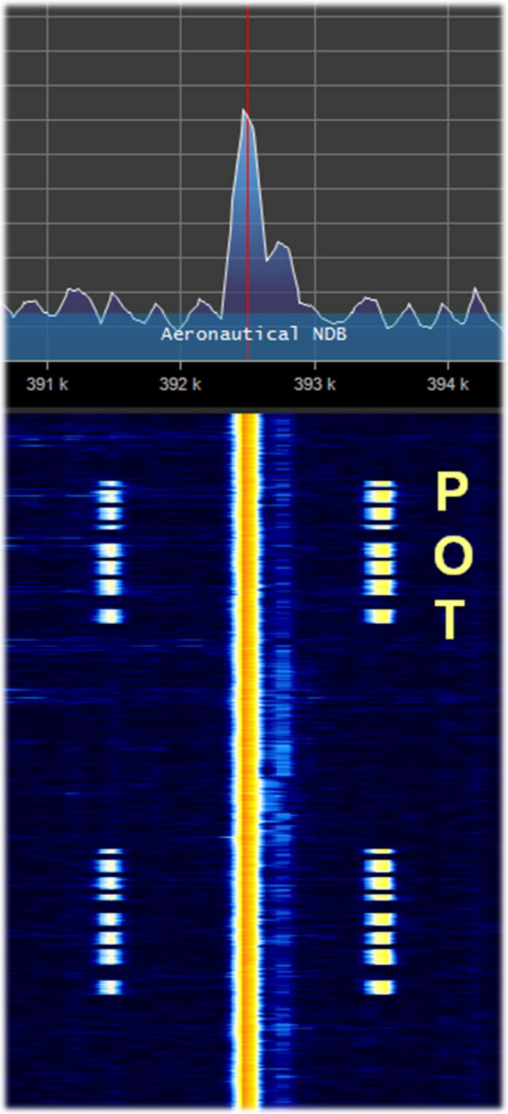
NDB's, dinosaurs in extinction...
Airspy HF+ Discovery

Non-Directional Beacons, better known as NDBs, are beacons used for many years for instrument air navigation or maritime radionavigation.

In the course of 2021, the Italian Air Navigation Authority has planned to phase out NDB, L and VOR type radio beacons at italian airports.

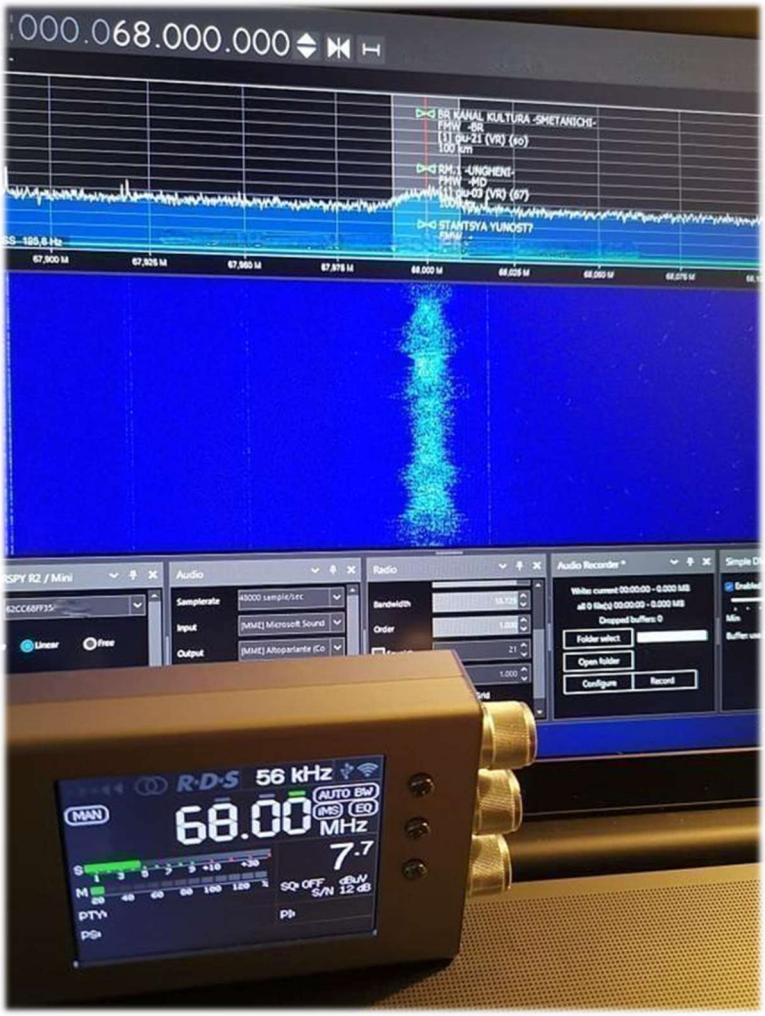
The NDB works in medium waves (between 200 and 1750 kHz), transmitting a continuous wave in vertical polarisation, on which an amplitude modulation of an audio signal is superimposed, through which the instrument communicates its identification in Morse code.

Here is an example of one of the last NDBs still receivable at the moment: 392.5 kHz and with “TOP” Morse code identification (Poirino/Torino - Italy) remembering that decoding starts at the bottom...



OIRT with TEF6686 and Airspy
… in FM an arduous comparison

In summer 2023 I had the opportunity to compare a TEF6686 receiver with the beloved Airspy. The opportunity was provided by the very long and fortunate days of E-sporadic propagation.
The amateur radio operator or SWL who is fond of FM-DX is always looking for the distant and rare signal, which, however, appears only under special conditions of the time of year and the solar cycle, and in our case that of the OIRT band from 65.8 to 74 MHz. The acronym comes from the translation from the French of International Organization of Broadcasting and Television and later merged into its Western European counterpart, the International Union of Radiophony (UIR). At that time it was the prevailing FM range for broadcasting in the countries of the former Soviet bloc, while there is very little of it left now: only from a few stations in Eastern European countries and Russia (such as Radio Rossii, Radio Stalitsa, Radio Mayak and Radio Brest...) which are gradually converging into the usual 88-108 MHz band.



I was saying about the reception that is more likely in the summer months during the warm hours and favored by the E-sporadic type propagation that allows to receive signals of fair/good intensity and without any interference. However, signal polarization changes unpredictably and abruptly and this is where a good receiver is needed to cover that band and mode. The best would be to use a directive antenna pointed eastward than can be done with an omnidirectional one.

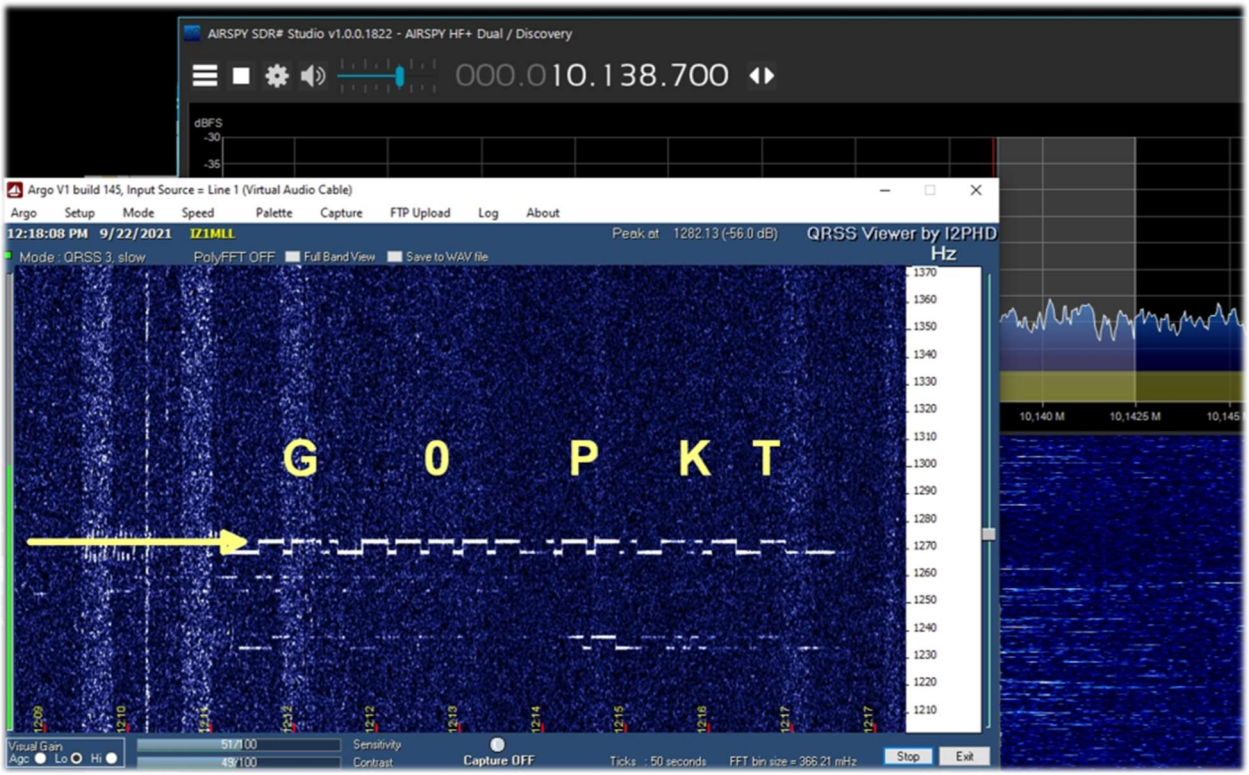
The Airspy is obviously the top but it is related to computer use... recently, on the other hand, the small and portable NXP TEF6686 with contained battery and speaker has appeared on the market. I will not expand here to list all its noteworthy features such as its extreme sensitivity and selectivity also thanks to automatic or manual BW filters in addition to its excellent performance on strong signals in combination with higher performance external antennas. RDS hookup is immediate (even on difficult signals) and with the latest firmware v2.00.15 it has a wealth of detail and RDS functions that we often do not find even in professional decoders.

Question: who won between the two? At the level of pure signal and intelligibility they were on a par, certainly the TEF has on its side the extreme portability while the Airspy with the LI (Listen Info) plugin remains unbeatable by providing on the Waterfall and/or RF Spectrum the indication of the possible broadcaster received...

QRSS signals not listenable and nor visible in RF spectrum! Airspy HF+ Discovery and software ARGO

Hams like to experiment and a lot, especially in the study of propagation...

The QRSS is a very special morse signal, transmitted so slowly that you cannot hear it by ear (a "dot" takes six seconds to be sent while a "line" takes eighteen seconds) and with very low power using frequency shift coding.
With this system you do not make conversation (in Q code called “QSO”) but you can analyze the propagation, test antennas or specific software. In my case I used the software "ARGO" but I suggest also "FSKview" for the visualization of the spectrograms of FSK signals.

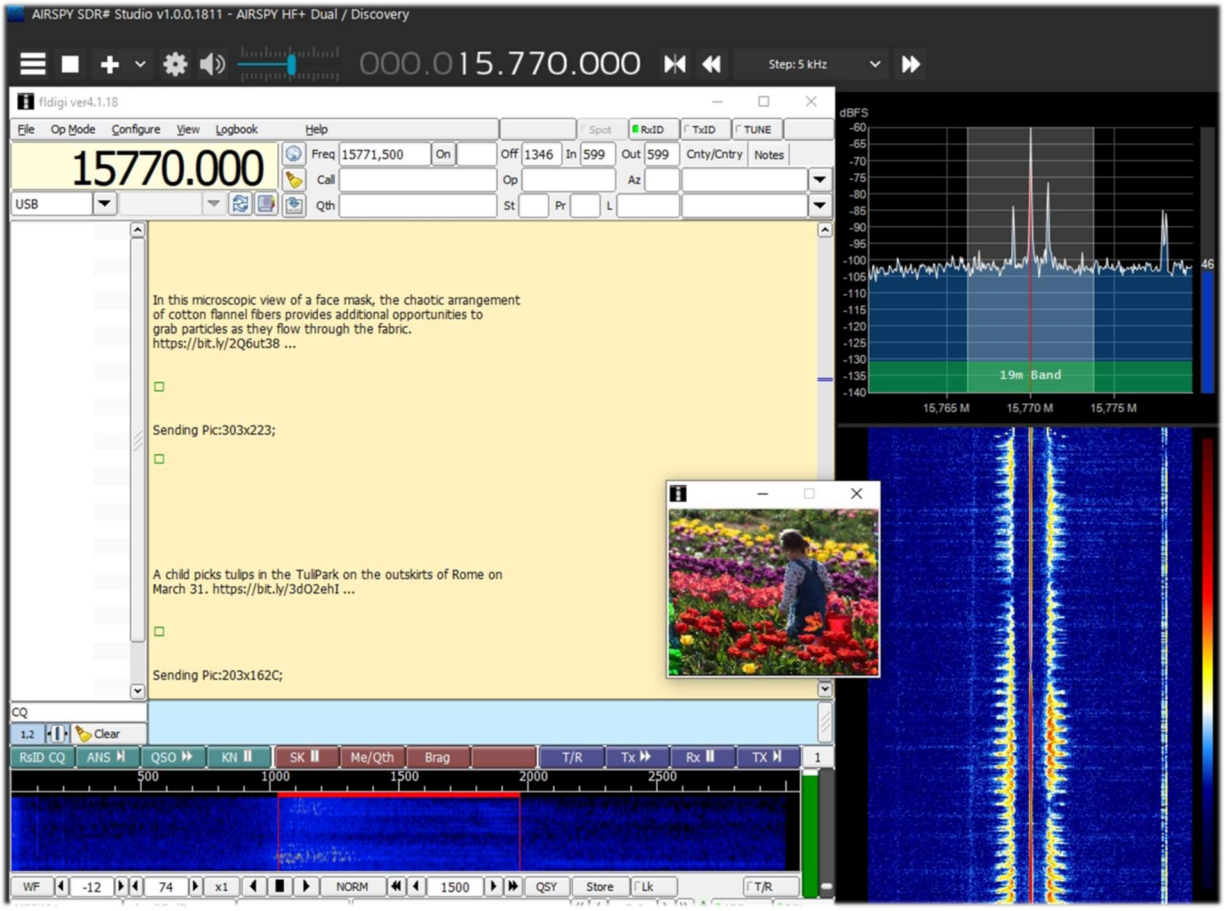


My SDR was tuned to 10138.7 kHz in USB and after several minutes I received and decoded the english beacon G0PKT as highlighted in the first line. Note that in the waterfall and in the RF Spectrum on the right no signal is visible...
From the net says that the beacon is active on 30 meters with the power of about 250 mW.

Try it on the other bands in WSPR and QRSS mode as well!

Radiograms (bulletins and images)
SDR# + Fldigi software

Using an Airspy HF+ Discovery, tuned in my example to the 15770 kHz frequency on certain days and at certain times, it is possible with the Fldigi software (previously mentioned) to receive curious transmissions, RadioGrams, i.e. digital text and images (MFSK-32/64 mode) via the analogue radio transmission...



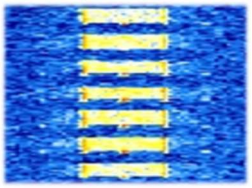
https://wiki.radioreference.com/index.php/Shortwave_Radiogram_Gateway

https://swradiogram.net/

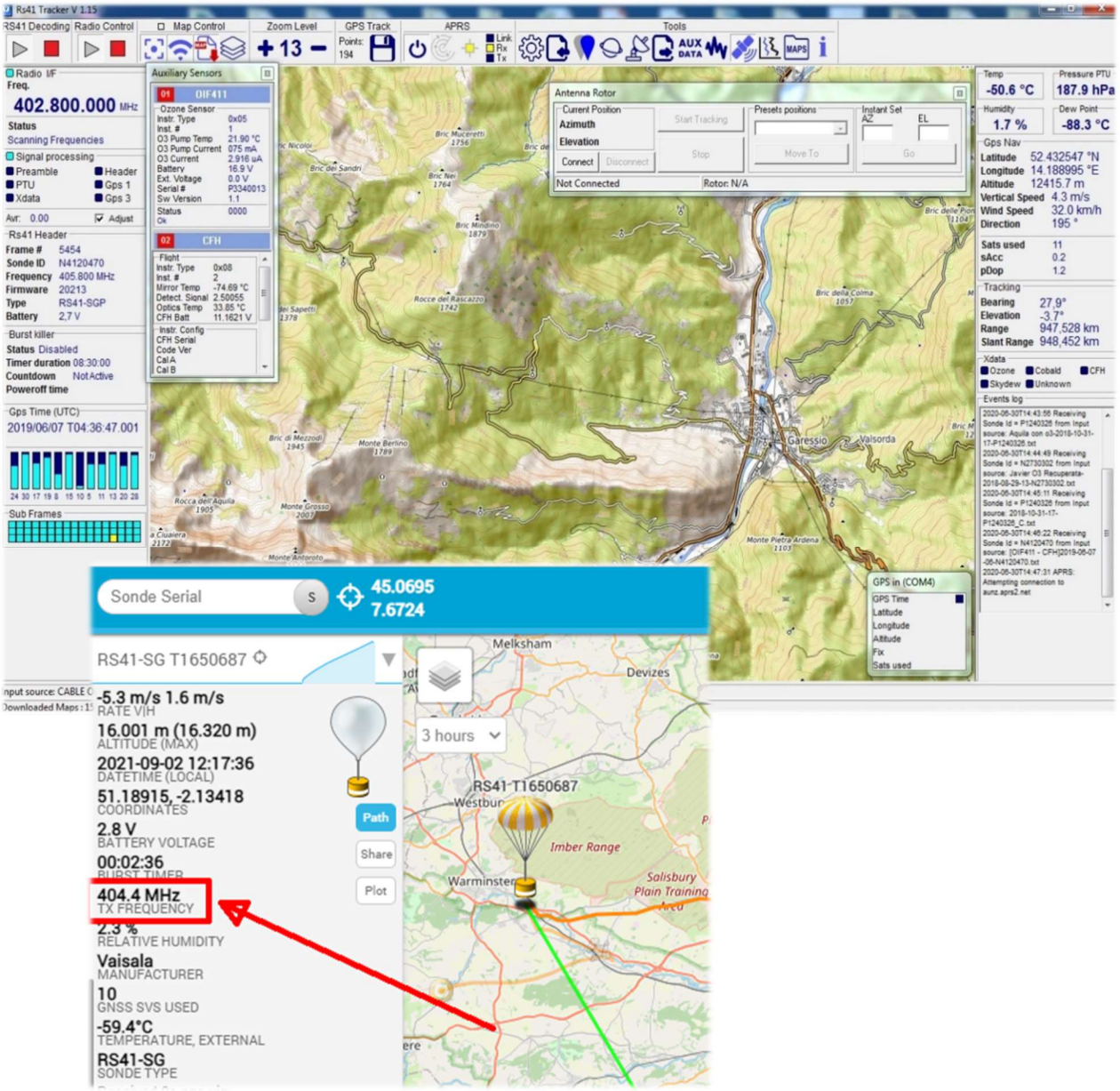
Radiosondes in UHF
Airspy R2 + software RS41 Trakers

RS41 Tracker is an external software, developed by Diego (IW1GIS), capable of decoding real-time telemetry from Vaisala RS41 radiosondes. Used in conjunction with an SDR it allows you to display the positions of the radiosondes on a map and control parameters such as height, temperature, wind speed/direction and burst killer information, etc.

Link: ***http://escursioni.altervista.org/Radiosonde/***

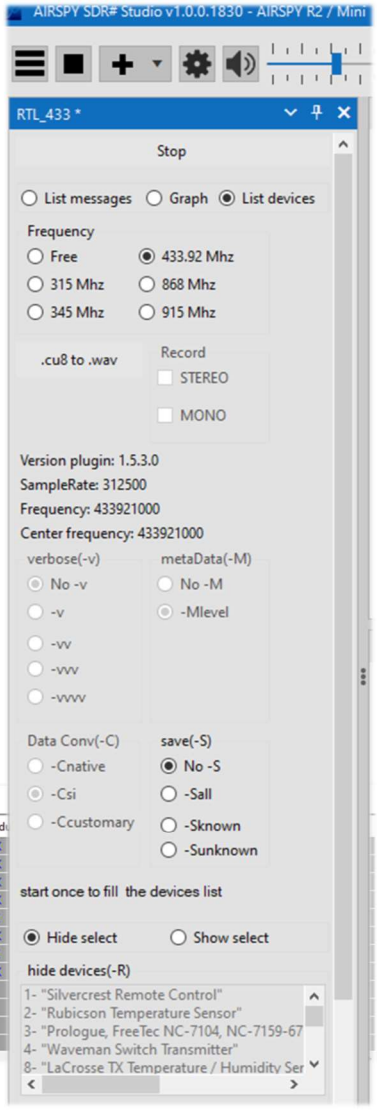


By tuning into UHF (here at the beginning of the 400 MHz band) at set times and with a bit of luck it is possible to receive directly signals like this and using a virtual audio cable send them to the decoder.



By referring to this informative link, you can also find the UHF frequency for transits in your area: ***https://tracker.sondehub.org***

RTL_433 for reading tyre pressure, weather sensors, etc. etc.
Airspy R2 and RTL_433 plugin

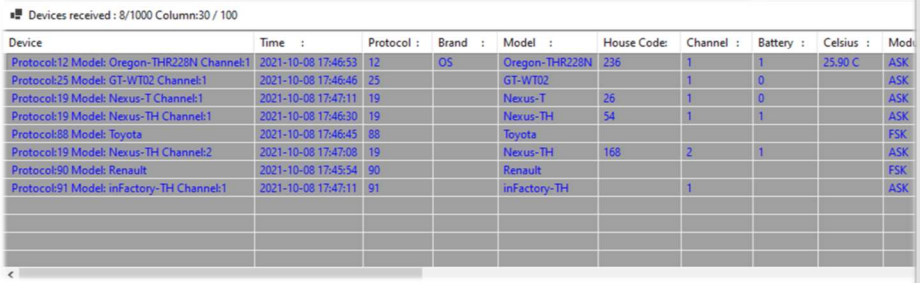


It is therefore possible to decode hundreds of sensors that detect temperature/humidity, weather data, energy consumption, tank level, etc... etc... and why not the TPMP ones, i.e. the tyre pressure and temperature monitoring system of some car models!

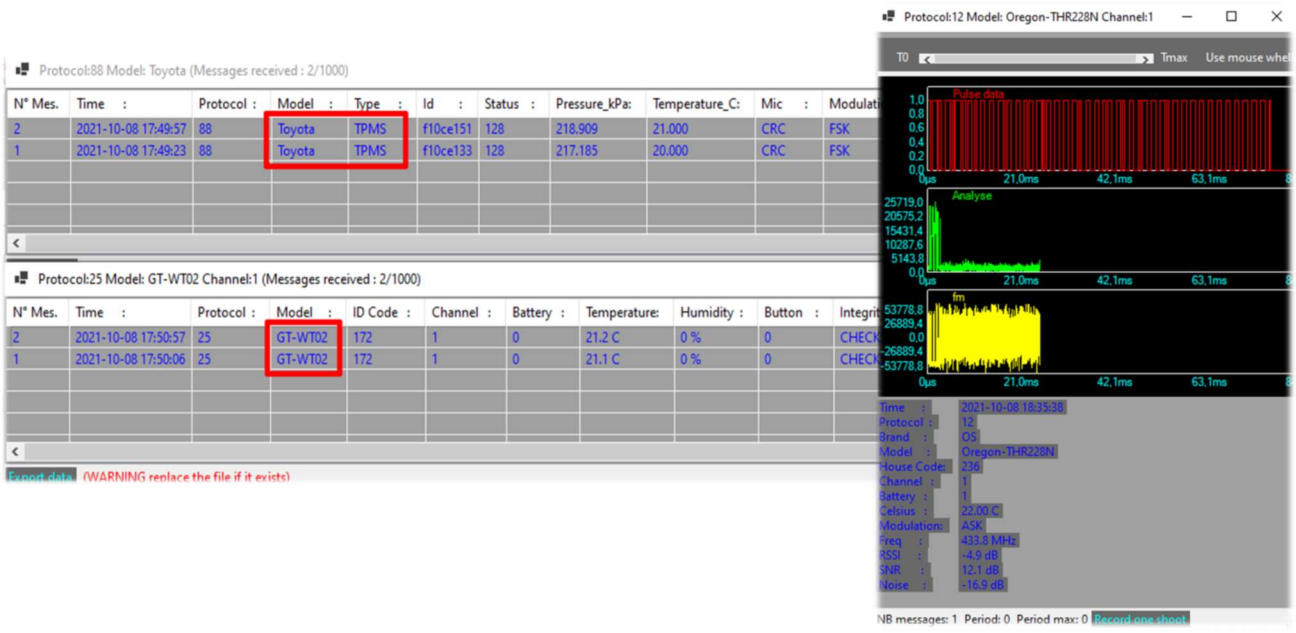
The plugin, with all the necessary instructions, can be downloaded free of charge, here:
here:https://marco40github.wixsite.com/website/plugin-sdrsharp-pour-rtl-433?lang=en

You can start by trying in RAW mode and with a bandwidth of at least 200k, disabling squelch and any other audio filters...

It can be configured to receive the overall message **list of all received devices**, with the possibility of exporting the data to files.



Or **single lists** such as the following with a Toyota TPMS and of an outdoor temperature sensor (model GT-WT02) or a **graphic window** of a termosensor (Oregon THR228N).



Satellite tracking: ORBITRON & plugin “Satellite Tracker2” Airspy R2

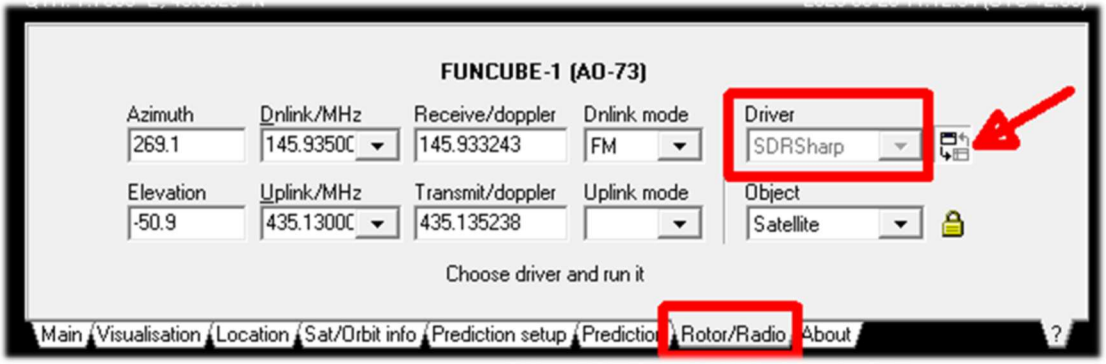
For all nostalgic guys (including me) of the dated, but very reliable, freeware satellite tracking software Orbitron, I propose the settings to interface it to SDRsharp.

The only assumption is that you use a version of SDR# prior to v1400.

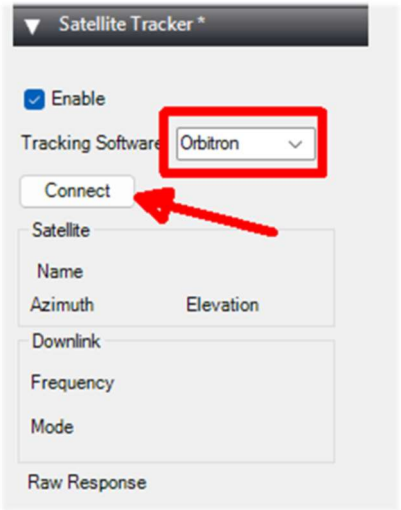
Start by downloading the plugin "SatelliteTracker2.zipper" for example from this link:
http://public-xrp.s3.amazonaws.com/SatelliteTracker2.zip
extracting its contents to the SDR# directory and write the following line in the Plugins.xml file:
<add key="SatelliteTracker" value="SDRSharp.SatelliteTracker.SatelliteTrackerPlugin,SDRSharp.SatelliteTracker" />

Let us now deal with ORBITRON and its minimal configuration.
First, these lines must be inserted at the beginning of the SETUP.CFG file in the CONFIG directory
[Drivers]
SDRSharp=SDRSharp.exe

Run the program and click on the "Rotor/Radio" tab. Select the "Driver" field to obviously choose "SDRSharp" and the path to its executable file.



Choosing a satellite will automatically set all the fields in the screenshot.
At this point clicking on the icon with the two little terminals directly launches SDR#, really a nice convenience...

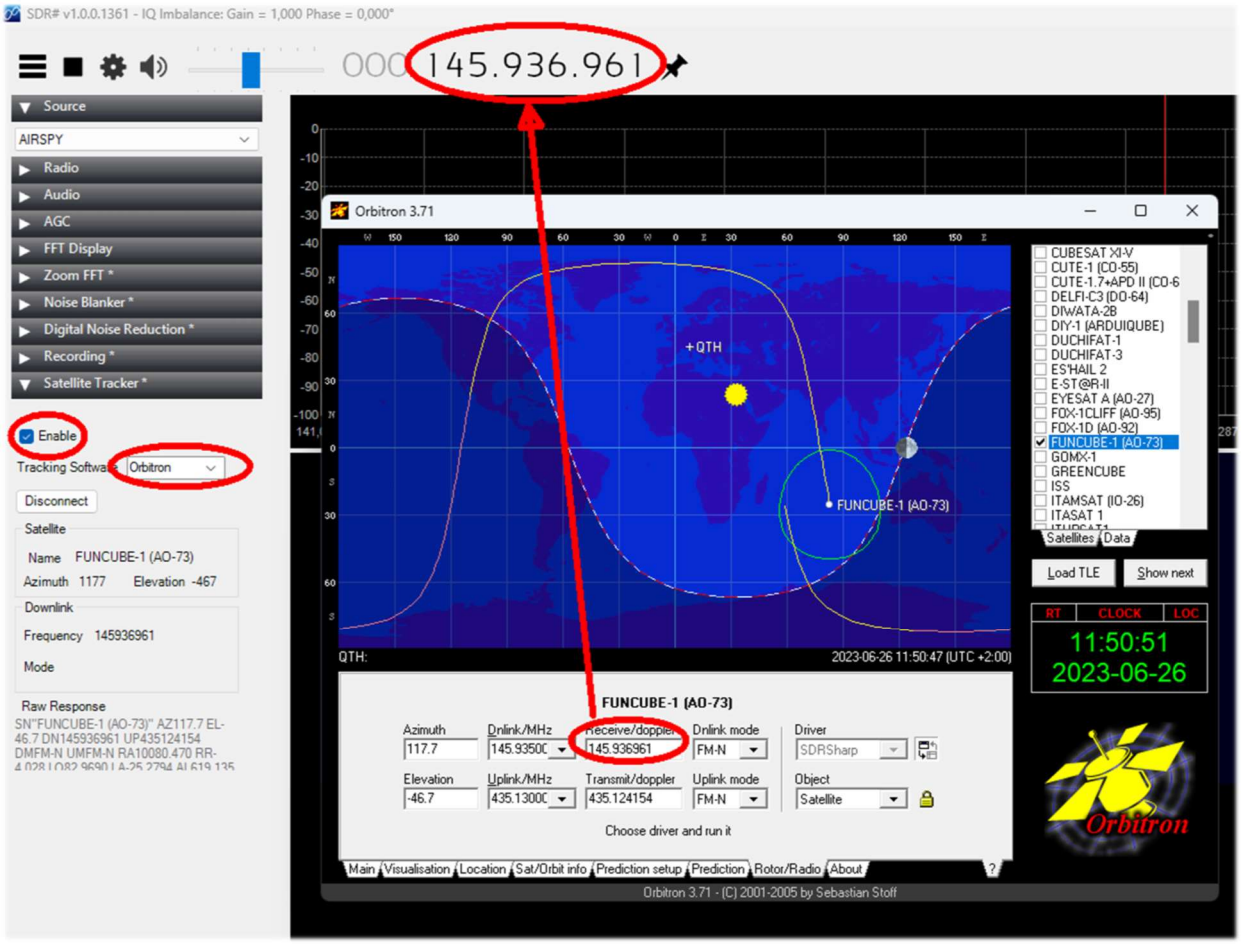


In SDR#, we go down and expand the section of our "Satellite Tracker" plugin.

Click on "Enable," select "Orbitron" from the drop-down field, and then click the "CONNECT" button.

In one to two seconds the connection between the two softwares takes place:

- The plugin itself will start displaying all the tracking data.
- The VFO frequency will be automatically corrected for the doppler effect.

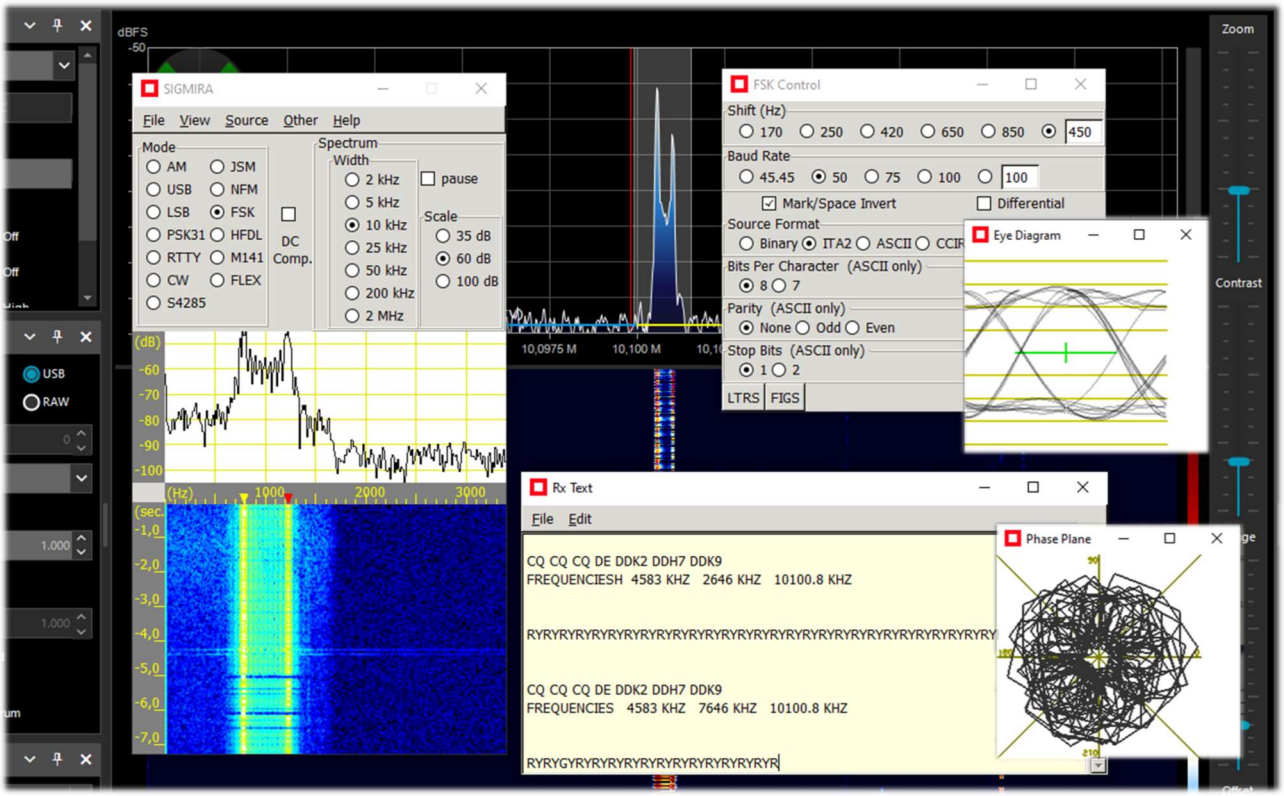


The screenshot shows the following things:

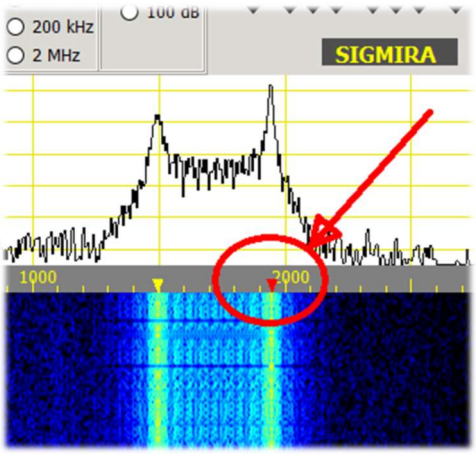
- Using outdated SDR# v1361
- In ORBITRON, after properly updating and loading the TLEs, I selected from the vertical table of satellites for example the FUNCUBE-1 (AO-73) which highlighted in the part below all the tracking data: Azimuth, Elevation, frequencies and reception/transmission mode
- The VFO of SDR# was immediately latched by the plugin and the Receive-Transmit/doppler frequencies updated in real time

SIGMIRA: multidecoder with database
Airspy HF+ Discovery

In the panorama of free decoders there is an very interesting software for Windows.
It is called SIGMIRA, it allows demodulation of the following modes: ALE, CW, FLEX, FSK, HFDL, PSK31, RTTY, SITOR-B, JSM-SLOT MACHINE, STANAG-4285 and the display of the spectrum in real time, waterfall and phase (Phase plane).
It accepts signal input via sound card (for conventional receivers) and via VAC as well as direct connection with some SDR devices.

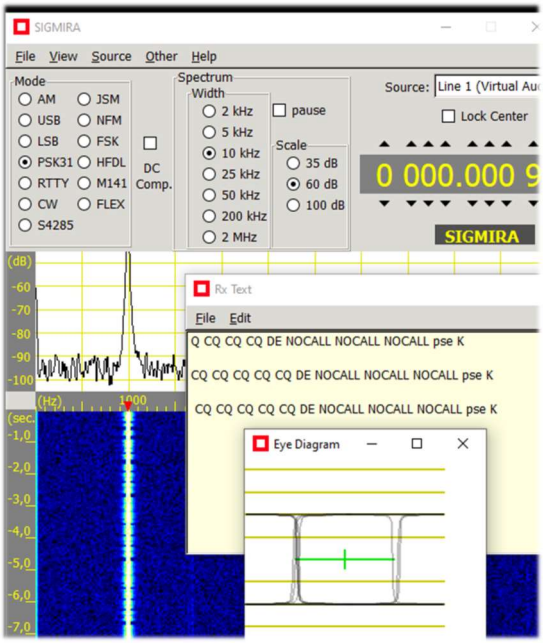


Here we see SIGMIRA with many of its windows open (selectable from the VIEW menu), struggling with an RTTY signal (50 baud reverse, shift 450 Hz) tuned through the HF+ Discovery (in USB mode) and its decoding of messages in the window "Rx text".



In the decoder, I chose the mode and parameters of FSK, then to click on the waterfall at the right signal marked with red triangle.

The yellow one will automatically adjust according to the preset shift: yellow and red correspond to the Mark/Space of the FSK signal.



You can also use as signal source: WAV files or external inputs like VAC.
In the example on the right I have used an old demo WAV file of the ham radio mode PSK31.

In this case the only red triangle should be placed with the mouse on the signal column to have immediately the decoding in the "Rx Text" window and display its shape in "Eye Diagram".

Another unique feature of SIGMIRA is its internal database with over 2000 frequencies in all modes of emission and different categories of users ("Cat1" column).

Signal Database

File

New | Delete | Scan | Freq Lookup

Frequency	Mode	Description	Cat1	Scan Cat.	Parameters
7,449570	hfdl	hfdl?, psk? much prior to 0812	fixed		
7,455000	fsk	150222_0711 utc 2311 lcl	fixed		850,50
7,456000	fsk	fsk, wide ?	fixed		
7,465000	am	broadcast, religious, s9+10 081225_2232	broadcast		
7,470000	am	ch faint 081228_0750	broadcast	ch	
7,480000	am	faint 081230_0105 utc 1705 lcl	broadcast		
7,505000	am	religious childrens' story, s9+30 070218_0115	broadcast		
7,520000	am	fr?, faint 081225_2233	broadcast		
7,527000	m141	PtReyes and coast guard 110324_0108 utc 1808 lcl	maritime	ale	
7,530000	am	asian 051231_0806	broadcast		
7,532000	usb	looks like 40kHz wide psk, 200824_0507 utc 2207 lcl	curious		
7,540000	am	asian 051231_0806	broadcast		
7,545400	usb	two way sp, 081226_0843	fixed		
7,555000	am	asian faint 051231_0806, sp s9+5 081225_2234	broadcast		
7,570000	am	faint 081230_0104 utc 1704 lcl	broadcast		
7,593000	fsk	fsk, wide, pauses 051126_0021 cratt2, 850s/75b, s9+10 061125_1135, 081225_1740	fixed		850,75
7,597000	fsk	s9 091126_2043 utc 1243 lcl	fixed		850,75
7,620000	am	faint 051231_0806	broadcast		
7,630000	usb	SAC?, eng. 7 groups of 5 letters and numbers,"Trilake 22, out." 1505 16_1355 utc 0655 lcl, Washington Cap 4602			
7,643000	usb	signal or noise?, 15khz wide, sinusoid sweep, 150224_0247 utc 1847 lcl	curious		
7,668276	ofdm	ofdm, 12 carriers, 75 baud approx 110307_0600 utc 2200 lcl	curious		
7,681000	am	numbers reported 061111_2359	numbers		
7,683000	usb	40 khz wide psk, 200727_0525 utc 2225 lcl	curious		
7,688000	usb	numbers, chinese, USB with carrier, 121013_1318 utc 0618 lcl	numbers		
7,730000	usb	broadcast, deu, religious, s9, 081222_2126	broadcast	deu	

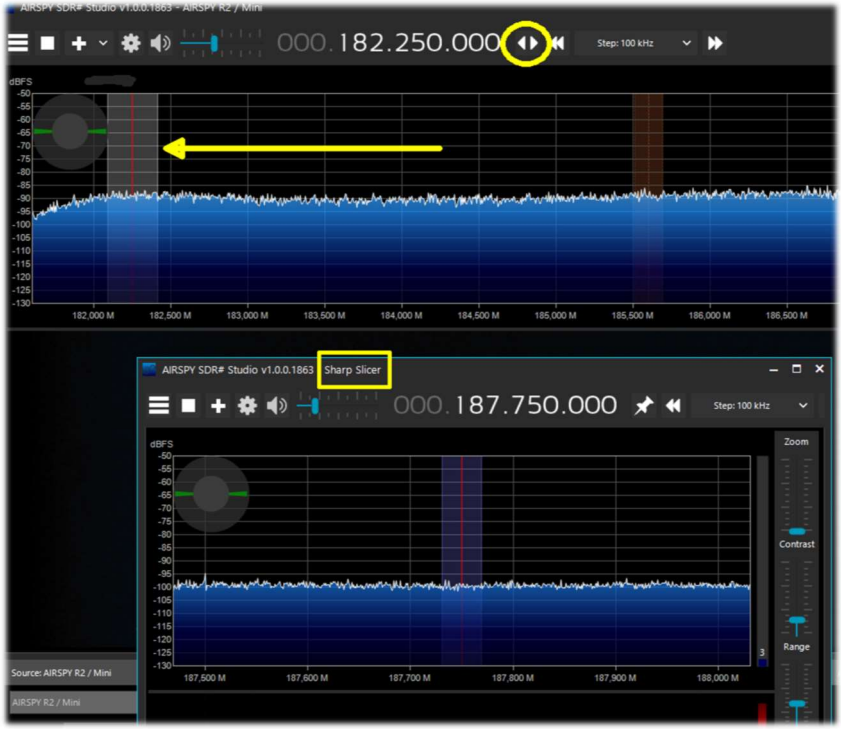
Link: ***http://www.saharlow.com/technology/sigmira/***

SLICE: its profiquent use!!
Airspy R2 with plugin “PAL/SECAM/NTSC TV”

I would like to point out Oscar EA3IBC's interesting video on making the most of the possibilities offered by the SLICE during a tropospheric propagation *(see the "Slice" item in the "Main Settings" chapter):* ***https://twitter.com/ea3ibc/status/1543670847625469952***

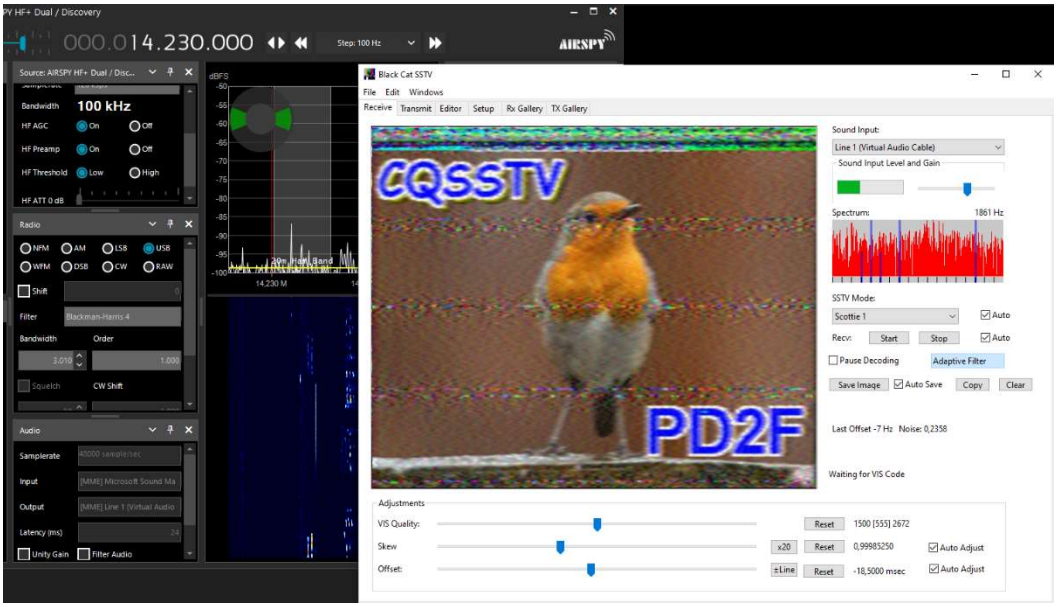
The video shows simultaneous reception of RTA1 Algeria's E6 television channel: with video carrier tuned to 182.250 MHz (with the appropriate plugin) and, thanks to Slice, also audio at 187.750 MHz.

In order to succeed at the first shot at the intent, I suggest using "Free Tuning" tuning (see the chapter "Main Settings and Controls") thus dialing the video frequency 182.250 MHz having the care to place in the waterfall as much as possible on the left side to have sufficient bandwidth...
At this point by adding a Slice (new VFO) we have a way to reach the audio frequency at 187.750 MHz.



SSTV …the charm of Slow Scan TV
Airspy HF+ Discovery and Black Cat SSTV decoder

The SSTV transmitted in HF by radio amateurs around the world has a very special charm and always arouses in me astonishment both as OM and SWL. Very often the signals are very bad and interference does not allow to receive good images, but sometimes with a little luck and good propagation there is plenty of time to be able to receive and decode a good image. Obviously we need a very sensitive decoder with advanced features such as Black Cat SSTV (for Windows and macOS): ***https://www.blackcatsystems.com/software/sstvhtml***



The developer has built many SSTV software in twenty years with a focus on decoding weak and difficult signals. It's easy to write an SSTV decoder that works with a strong signal, but he decided to write a new SSTV application from scratch, with an emphasis on performance under weak signals.

The decoder has an extremely sensitive VIS detector, with an adjustable threshold depending on personal tolerance for false triggering. The images are automatically adjusted, after reception, also for Skew and Offset using all transmitted information, for an almost perfect lock even with extremely weak signals. Images can be automatically saved to a directory of your choice and there is a built-in gallery to view received images.

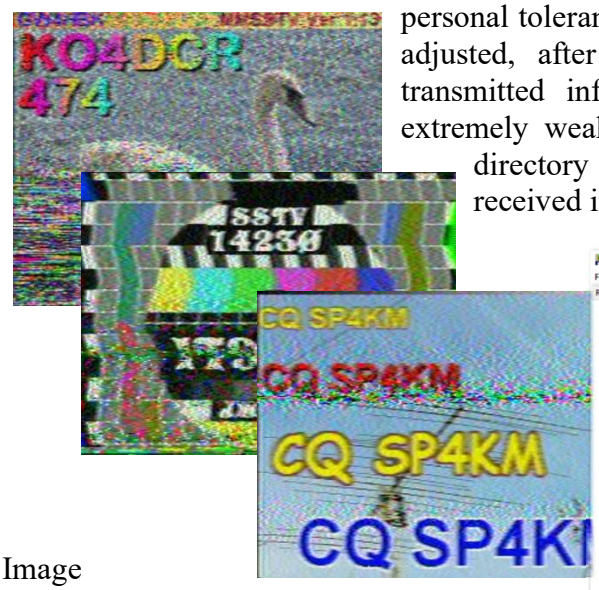
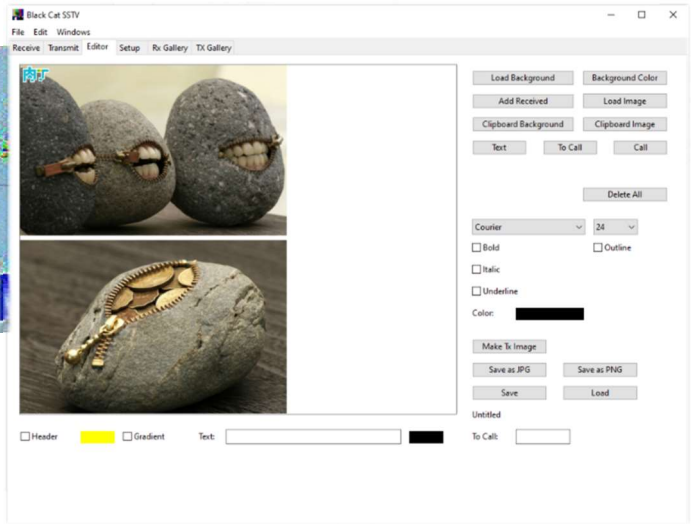


Image transmission is also supported and there is a basic editor (see side screen) to prepare images for transmission.

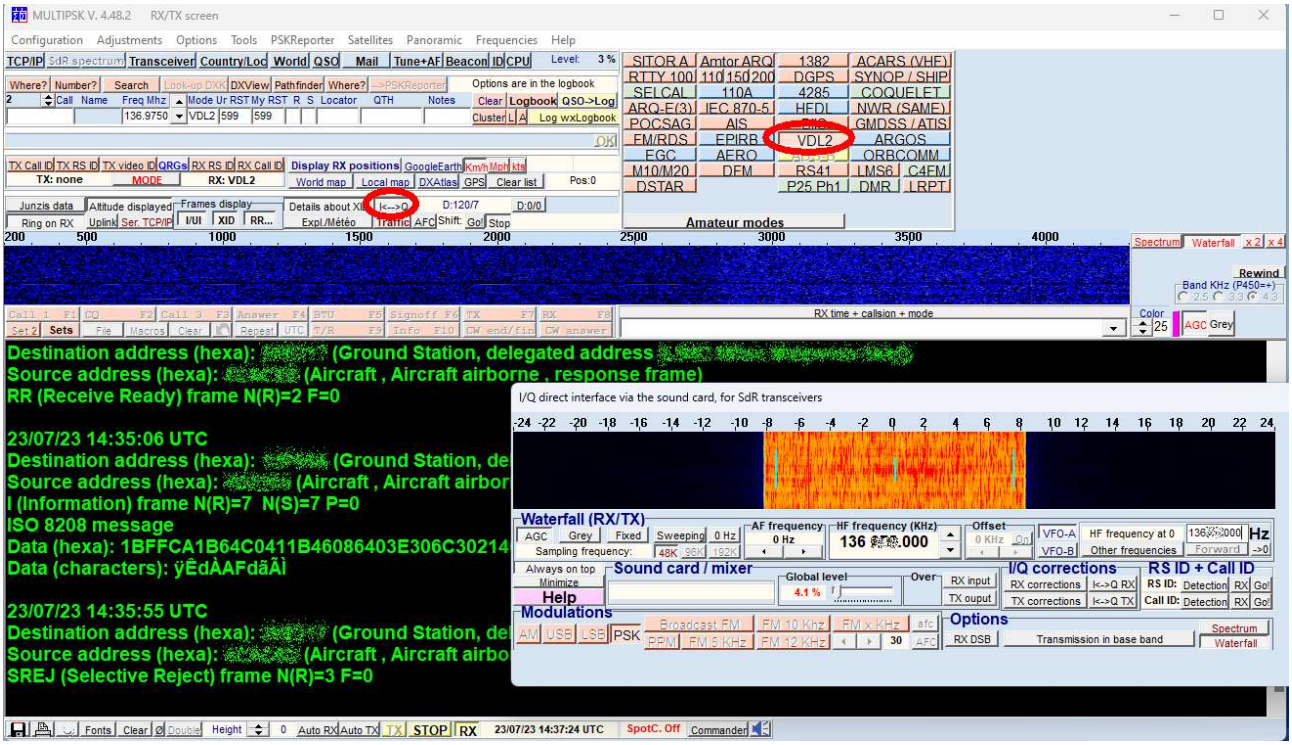


VDL-2 (Vhf Data Link mode 2)
Airspy R2/HF+ Discovery and MULTIPSK/KG-VDL

This system is in operation in Europe since many years (successor to ACARS) and provides better VHF links in terms of volume and wider variety of messages. These are some of the most salient technical features:
Channel allocation: 25 kHz
Modulation: D8PSK - Differentially encoded 8 Phase Shift Keying
Symbol rate: 10500 baud
Bit rate: 31500 bits/s

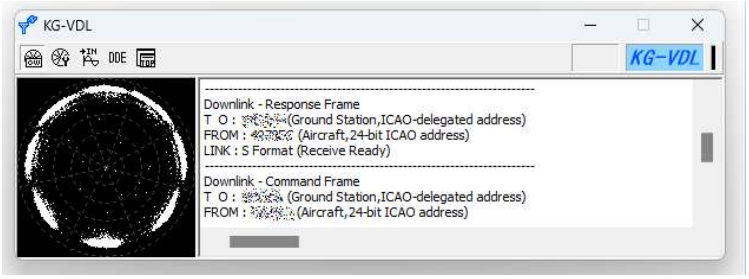
Initially decoding was the purview of only a few professional decoders of unattainable cost, then things changed and it is possible to decode the very fast bursts with, for example, Multipsk or the Japanese KG-VDL, on one of several frequencies available on the 136 MHz band (where receptions up to several hundred kilometers are also possible).

Depending on one's receiving system, to decode VDL-2 smoothly, it is necessary to demodulate the signal to RAW/USB, use a bandwidth of 20/25 kHz, perhaps invert the IQ channels, and exclude or adjust the receiver's AGC as best as possible. These some tests performed...



For MultiPSK it is necessary to set in SDR# the RAW mode, AGC active and no flag under Unity Gain.

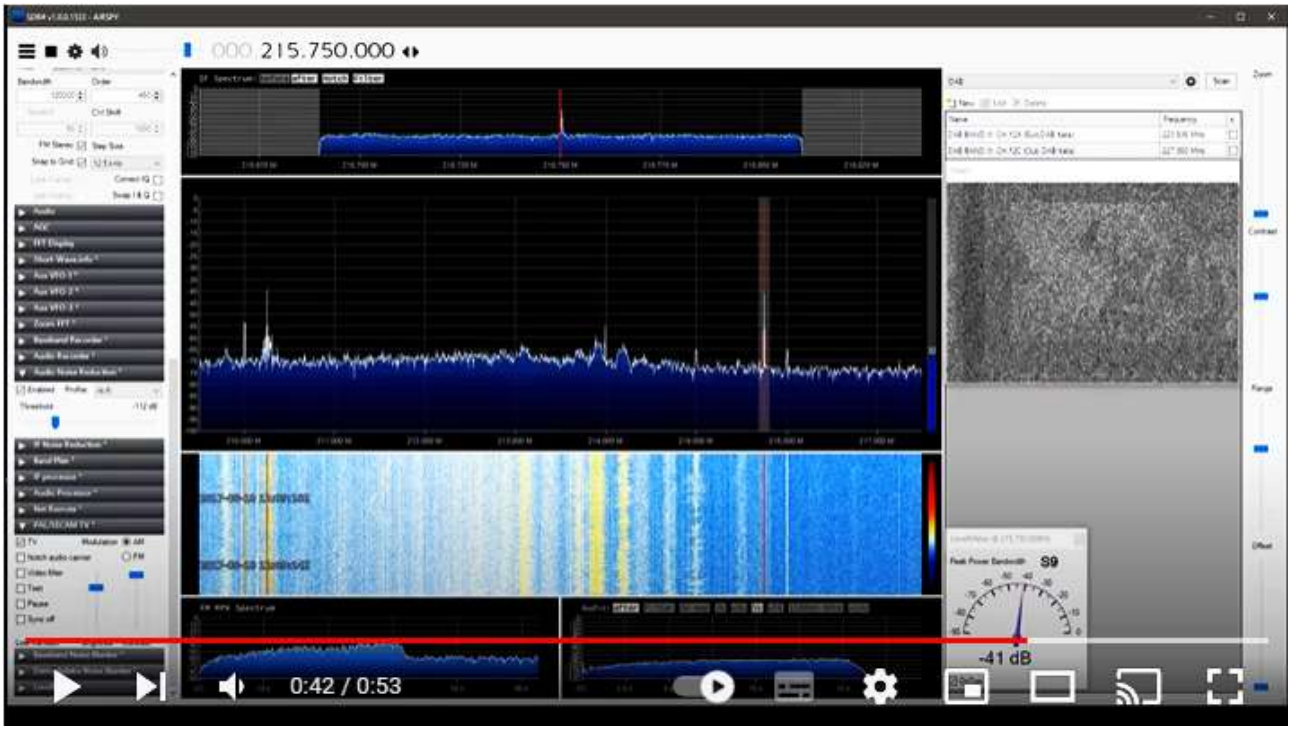
For KG-VDL it is necessary to set the USB mode.



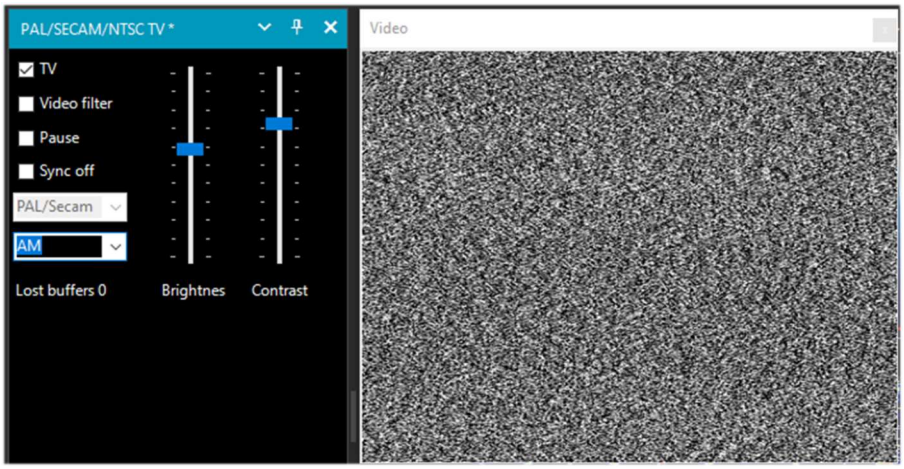
VHF and TV ...the fascination of extreme dx!
Airspy R2 + plugin PAL/SECAM TV

Talking with my friend BlackApple62, an his interesting VHF reception from August 2017 emerged, which I repost here...

It is related to the analog signal (video and audio) from Algeria for him more than 1000 km away. Reception at 215,750 MHz was with a simple SW oriented dipole.
The AM video signal does not synchronize well and is barely visible on the right side of the image.



Here the video: ***https://www.youtube.com/watch?v=ng2EFIwaoRA***



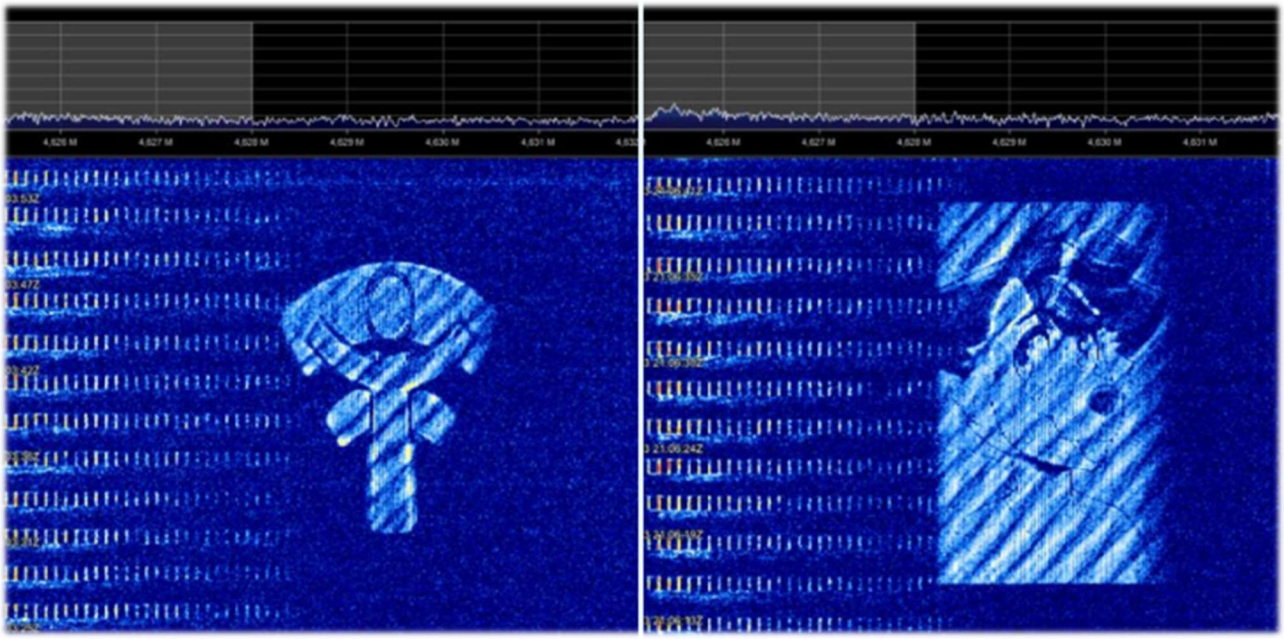
Really a pity not to have more analog PAL signals to test with Ian Gilmour (MM6DOS) excellent plugin, but the last word is never said in propagation season...

Waterfall draw …drawings directly in the SDR waterfall Airspy HF+ Discovery

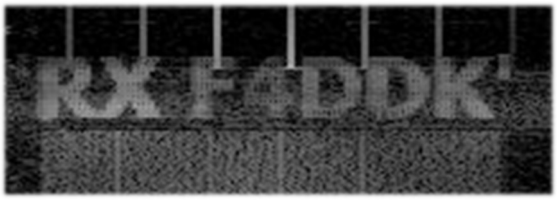
I have received many of them over time and as many examples can be found easily on the net. They have nothing to do with the usual ham radio modes such as SSTV or similar.

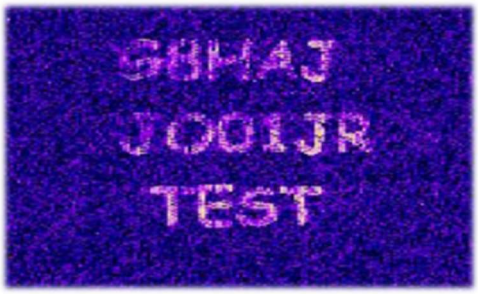
There are some software (like SonicPhoto) that will convert a JPEG image or short text into sound. Then after a few more steps you can proceed to transmit the file over the radio so that the sound will draw the image directly onto the recipient's waterfall.

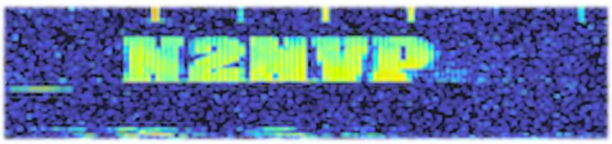
Here are a couple of images received from a friend on shortwave...



And a few callsigns transmitted with this curious system…







Here is a video of the procedure with all the necessary steps:
https://www.youtube.com/watch?v=6szieDSud-o

Things to know for don't lose the head...

It can sometimes happened that after particular changes or risky actions the program will fail due to internal (or often external) code problems. Many things have changed since the v177x (including scheduled Windows updates…), sometimes if something crashes, it is for external problems at SDR# code. All errors are automatically detected and recorded in the "crash.txt" file in the program directory...

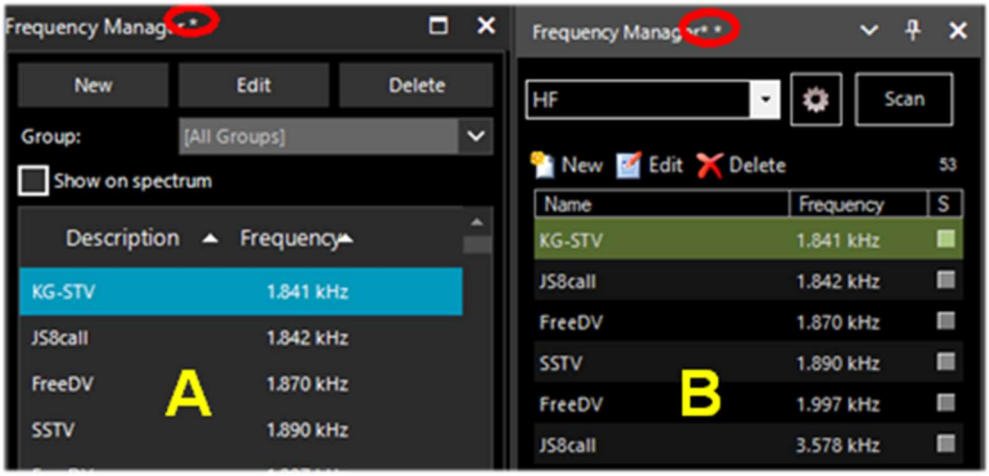
Maybe the only thing to do, if the program is very "customized", is to copy again the file "SDRSharp.exe.config" from the original distribution package. You will lose some customizations (example of the "audio recorder" panels) but this way it will start again for sure. So I suggest to save this file in a moment that everything works so that you can reuse it when you need it. Or even to diversify SDR# installations on your HD and keep a "test directory" to test and verify the new plugins or own customizations.

Running old plugins that are no longer compatible can also lead to some headaches and initial misunderstanding.

For this we are helped by the text file "PluginError.log" (possibly present in the SDR# directory), which collects track of errors arising when the plugin fails to load.

In the next personally verified example, the PluginError.Log file was helpful in understanding a problem. Here's how.

Since v1890 it is possible to jointly load plugins that have the same name (third-party example in addition to internal ones). Such as the Audio Recorder and the Baseband Recorder. But also the Frequency Manager (standard) and the one from author "Thewraith2008" can coexist at the same time as seen in the following screen (note also the presence of a * in the header of the standard one (figure A) and ** for the one from "Thewraith2008" (figure B).



However, two friends reported to me that with v1891 the (B) was no longer loading on some computers with italian language OS. With the invaluable help of "Prog", interpreting the following PluginError.log file, it turned out that the problem was related to the decimal separator of the international settings, ***thus a critical issue internal to the plugin itself and not due to SDR#.***

*** Plugin Load Error - 2022-08-24 16:53:04.519

Config Key 'SDRSharp.FreqMan.FreqManPlugin,Plugins\SDRSharp.FreqMan.dll'

Type 'SDRSharp.FreqMan.FreqManPlugin, SDRSharp.FreqMan, Version=1.1.9.0, Culture=neutral, PublicKeyToken=null'

Message 'Text "Microsoft Sans Serif; 8,25pt" cannot be parsed. The expected text format is "name; size[units[; style=style1[; style2; ...]]]". (Parameter 'value')'

Stack Trace

at System.Drawing.FontConverter.ConvertFrom(ITypeDescriptorContext context, CultureInfo culture, Object value)

at System.ComponentModel.TypeConverter.ConvertFromString(String text)

at SDRSharp.FreqMan.FrequencyManagerPanel..ctor(ISharpControl control)

at SDRSharp.FreqMan.FreqManPlugin.Initialize(ISharpControl control)

at SDRSharp.MainForm.InitializeSharpPlugins()

In other cases and situations it has been verified that some problems came from too many devices plugged into the same powered HUB. *Therefore it is preferable to connect the devices directly to the native USB socket!*

Another suggestion is to avoid the simultaneous use of 4 or more RTL-SDR devices (4.8 MSPS) on a single USB2 bus. Preferable then is a USB3 card...

The Microsoft .NET Runtime can also sometimes cause problems when starting SDR#, especially if you have previous versions installed on your PC (perhaps a mix of x86 and x64). It is recommended to use a good uninstaller to do a complete clean up and reinstall the software from the following Airspy link: **https://Airspy.com/?ddownload=6293**

After some specific Windows 10 updates it happened that you could no longer send audio to external decoding programs (e.g. Fldigi, HFDL, WSJT, etc.). *I suggest checking this:*

- Select **Start** > **Settings** > **Privacy** > **Microphone** . In **Allow access to the microphone on this device,** select **Change** and make sure **Microphone access for this device** is turned on.
- Then, allow apps access to your microphone. In **Microphone settings**, go to **Allow apps to access your microphone** and make sure it's turned on.

Always about the audio!

You might get further improvements (and even less criticality) moving to 16-bit audio. There is no audible advantage to 24-bit audio, so why bother with it? All SDRs presume 16-bit audio in their demodulation path, use float (32-bit) rather than double (64-bit) computations to reduce computer load.

*In Windows10 select **Audio Control Panel > Properties > Advanced** to check the features in playback/recording and of course also remove the flag from "Enable audio enhancements"…*

Check your computer's performance

A number of utilities (command-line) have been developed to help detect and resolve some performance issues often related to USB controllers/drivers.

WINDOWS

Download the latest version of the software: ***https://github.com/Airspy/Airspyone_host/releases***

- Open a console (cmd.exe) and run: `Airspy_rx -r NUL -t 0`
- Let it run for 30 seconds, then close it with `Ctrl + C`
- If the average throughput is below than 10.0 MSPS then either the USB controller has problems or the CPU can't process the data at this rate.

Possible solutions:

- Try another USB port (avoid external HUBs and port repeaters)
- Update the USB drivers (prefer OEM drivers to generic ones). For more details see also: ***https://github.com/libusb/libusb/wiki/Windows***
- Check antivirus or any other software at the same time with heavy loads on the CPU
- Use a PCIe USB 2.0/3.0 controller

LINUX (Debian/Ubuntu) Ubuntu possibly with the distro 14.04 LTS.

- ### Building Airspy, gr-osmosdr and gqrx:

Download the Airspy-git repository, compile it, install it
Download the gr-osmosdr repository, compile it, install it
Download the gqrx repository, compile it, install it
Get pulseaudio using standard Arch way
Configure pulseaudio (add user and group…)
Thanks to SEGFAULT post ***http://Airspy.com/?topic=linux-Airspy-gqrx/#post-658***

- ### Performance problems:

- Build the host tools following "How to build the host software on Linux": ***https://github.com/Airspy/host***
- Open a shell and run `Airspy_rx -r /dev/null -t 0`
- Leave it running for 30 seconds, then `Ctrl+C`
- If the average throughput is below 10.0 MSPS then either your USB controller has problems or your CPU can't process the data at this rate.

Possible solutions:

- Use another USB port (avoid external HUBs and port repeaters)
- Update your kernel
- Use a PCIe USB 2.0/3.0 controller

For further technical details:

https://github.com/Airspy/Airspyone_host/wiki/Troubleshooting

Wiring of the devices and their positioning

The advent of 3D printers provides the ability to create very custom accessories and storage boxes. However, the general consensus seems to be not to use any form of support/mounting that would limit heat dissipation, perhaps inside small plastic enclosures even for weather protection when used outdoors or in an attic.

At the limit just a small piece of double-sided Velcro to stop it on the receivers shelf, but for my part I prefer to leave them free on the table of the radios, maybe in the vicinity of a small fan properly turned on just in the hottest summer months to help the cooling of the outer shell.

Another issue concerns the "micro USB" cable and connectors that must be arranged for minimum tension, pressure and torsion so that they do not create mechanical stress on the connector itself and the underlying PCB to which they are soldered.

Rigid cables are not a solution because they tend to lift the connectors from the boards and the soldering and tracks on the PCB are insufficient to maintain contact for long periods of stress. Here's a good advice: **Don't stress SMA connectors with cables intended for boat anchor radios.**

Also not recommended the continuous connection/disconnection of the cable from the device "micro USB" socket (surely preferable to do it from the side of the normal USB socket of the computer).

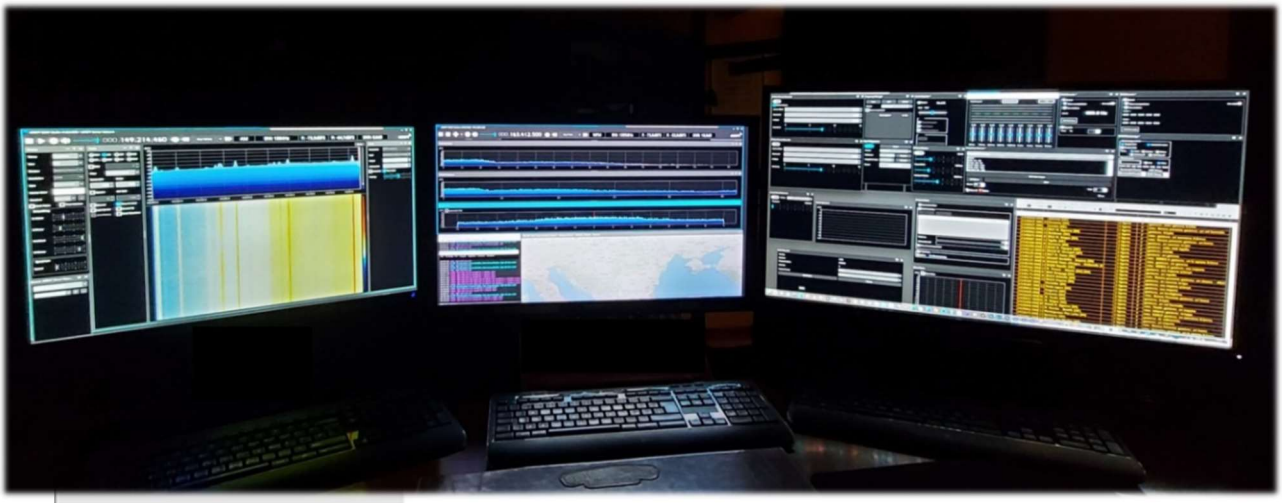
Also the antenna connection would be preferable to make it through short SMA connector cables (male/female) of excellent and thin flexible cable to connect in line your more robust and rigid coaxial antenna cable maybe even equipped with heavy and bulky adapters. All this will help to remove physical stress and allow long life to our small devices...

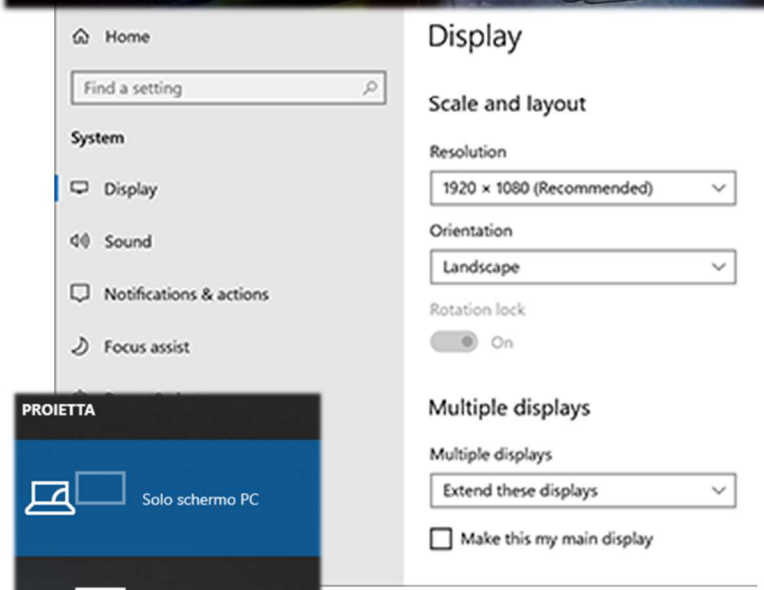
Ideas and suggestions

• Multi-monitor configuration

Dear friend "Pierluigi" informed me about the possibility of using in Windows 10/11 the "extended video" mode that allows interesting things if you have more than one video output port in your hardware. The idea was to be able to use two or more external monitors (even ultrawide) by dedicating a specific function to each! Obviously, one's computer's video card must have multiple outputs. On laptops/notebooks this is not always possible since the output is built-in but HDMI can be used, if present.

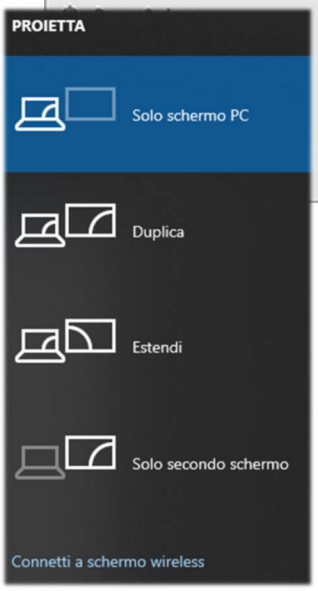
What do you think of such a shack? With SDR# expanded on as many as three monitors!!!





In general in Windows 10/11 these are the steps required:

- Select Start, then open **Settings.**
- In System select **Display** / **Multiple Display** (...picture on the right).
- Use the drop-down list to choose the desired mode.
- After choosing the configuration, select Apply.



Or again to configure an external monitor press **WINDOWS + P**. On my laptop with Windows10 this menu in the upper right corner where I can choose from:

✓**PC schemo only** (the classic mode, that is, display only on the main monitor).

✓**Duplicate** (display the same thing on two monitors).

✓**Extend** *(what we are interested in for displaying our desktop on multiple monitors.)* After "extending" the view, you will be able to move items of interest between the two monitors!).

✓**Second screen only** (display on the secondary screen only).

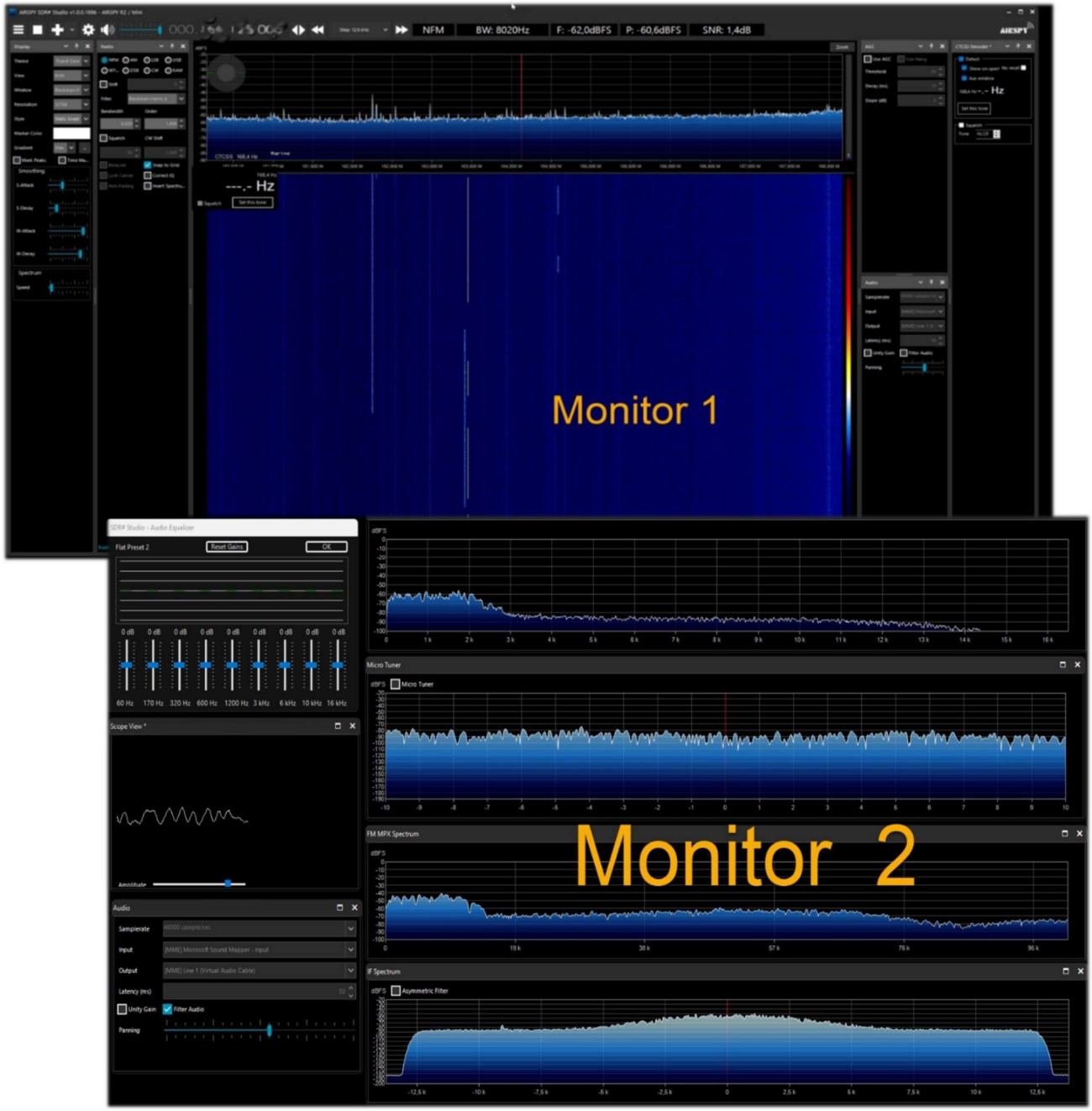
And so here is the idea ...thank you Pierluigi!

Dedicate one of the monitors to SDR# open with the main controls and panels: VFO, RF Spectrum, Waterfall, Radio, AGC, Audio, Recoder (audio and Baseband).

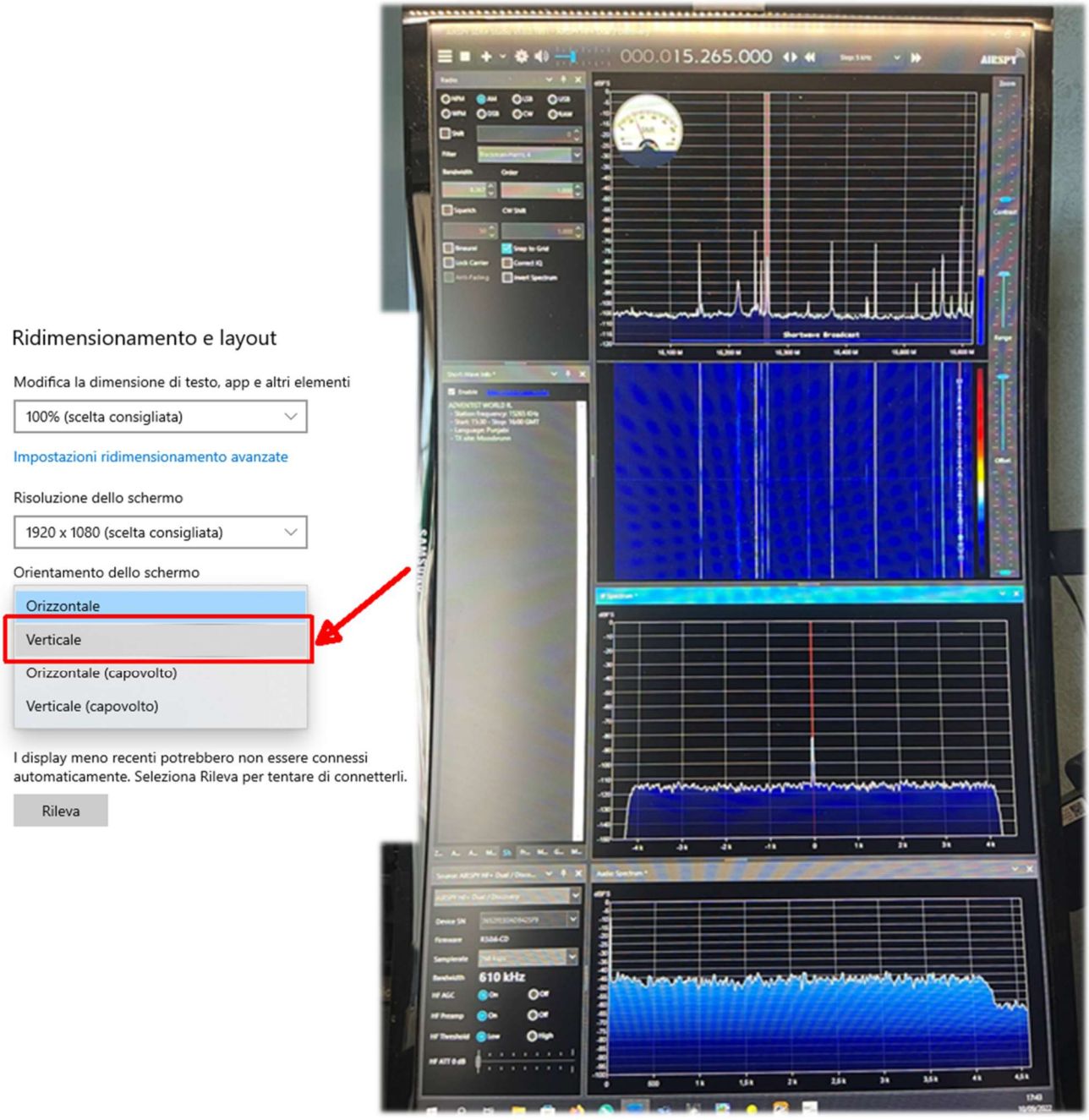
The other monitor, on the other hand, will be useful for us to conveniently display other windows such as Audio Spectrum, IF Spectrum, FM MPX Spectrum, the new Micro Tuner, or external plugins such as Audio Equalizer, Frequency/Scanner Manager, etc. etc.

But now a world really opens up because the secondary monitor (or even a third one!) one could think of dedicating it to various external decoder software (which I mentioned in the chapter "Listening Recipes") or to database/list management software like CSVUB which for richness of data and information like very very large desktops!

In the following screen SDR# on "Monitor-1" the main controls, and on "Monitor-2" the auxiliary windows mentioned above...



But there are also those who have gone further!!! Mr. "Jose Angel C." tried to put his monitor upright with this curious result...



It should be pointed out that verticality can be achieved in two ways:

If you have a monitor that can rotate on its axis (called a Pivot) the image is reproduced by inverting the dimensions.

Or you can use the windows utility that reproduces the image on the monitor by readjusting it. Example with a 28" Pivot monitor you have a perfect A3 size image, but not in the case of the windows utility. However having SDR# several independent windows of the various plugins, these can be arranged as desired.

- **MacOS & SDR**

I receive and turn over to everyone, the following notes from Andy, after we have talked about it more than once throughout our years of deep friendship and knowledge of the world of radio and telecommunications.

The power and richness of SDR# in terms of signal processing and functionality (further extended by the program's plug-in architecture), has always attracted much interest from radioamateurs and enthusiasts loyal to the Apple platform. The Macintosh operating system (System, then Mac OS X, now simply MacOS) has established itself over the years as a platform for music production and scientific application-oriented DSP, but from the standpoint of Software Defined Radio applications, the weight of Apple's environment is far less than the authority and popularity of Windows applications. In the area of multipurpose SDR clients, i.e., compatible with different hardware front-ends, there is no native Mac equivalent of flagships such as SDR# and SDR Radio Console, but neither of old and new workhorses such as HDSDR or SDRuno.

The disappointment among "Macintosh peoples" is all the more bitter when one considers that for some fifteen years, from the 2005 announcement of the switch to Intel chips from the previous RISC PowerPC architecture to the advent of the new machines based on the Arm64 architecture of the Apple Mx family, the three monotheistic computer religions, Windows, Apple and Linux, shared the same low-level code base. Unfortunately (or fortunately depending on your point of view), differences in both low and high level BIOS, interrupt handling, dynamic libraries, languages, development frameworks), meant that software developed for one environment was not always immediately "portable" to and from the other platforms.

Nevertheless, there is some room for maneuver, both on the front of the possibility of "crisply" installing on the Mac a code developed in the Windows environment by recompiling the sources; and through appropriate ways of emulating Windows executables. In a manual devoted to SDR#, the first question concerns this very application: can it also be used on an Apple computer? The answer cannot be unambiguous.

It depends. On an Apple system of the Intel generation, the most immediate route is to create a Windows partition with Boot Camp Assistant, a system utility that basically handles Mac system calls as within a Windows PC. The Microsoft operating system (but the same would be true for a Linux distro) is basically "convinced" that it is running on a non-Apple machine, and native Windows programs run smoothly.

There are, however, some alternative routes, which make it possible to avoid the Boot Camp approach (wasteful in terms of disk space to be allocated to the non-MacOS partition).

The first route is through emulation in a Virtual Machine, through commercial platforms such as Parallels Desktop, perhaps the most well known and widely used Windows emulator today. Attempts to run SDR# with satisfactory results have also been made with another commercial emulator, VMWare Fusion, and with Oracle Virtualbox (a largely open source project under GNU license). In the past there have been experiments with running SDR# through a "compatibility layer" such as Wine, the project that makes a good percentage of Windows applications executable in Posix-compliant UNIX environments.

Finally, up to eight years ago it was possible to recompile SDR# source code (developed in C# on Microsoft .Net framework) on Apple Intel machines, thanks to the existence in the Macintosh world of tools such as MONO (emulator of .NET for MacOS) and the Xamarin development environment, oriented to the portability on Mac and iOS of Windows code. A solution, the latter, not easy to implement and now no longer viable.

Making an eventual coexistence even more complicated were two major evolutionary steps in the Macintosh operating environment. In moving from Mac OS X to later versions (Big Sur MacOS 11 and Monterey MacOS 12) but especially in choosing the new arm64 architecture, Apple somewhat turned the tables on emulation and recompilation.

Gradually, Parallels Desktop and VmWare Fusion have adapted to the new architecture, and today especially Parallels is able to provide a virtual machine for Windows 10/11 that is defined as very performant. At present, the VM approach is the only one that offers any hope of using SDR# on the more powerful Apple M1 systems. Conversely, on Intel machines predating the Big Sur operating system (but perhaps also Catalina 10.15), it remains possible to resort to Boot Camp (the safest method), virtual machines, or even recompilation of pre-2014 SDR# sources.

Looking forward, one can envision for SDR# a future of convergence and interoperability centered on the cross platform availability of the Microsoft .Net 2020 development environment Microsoft in fact announced that with .Net 5 it was officially on the road to unification of .Net by merging the .Net Core and MONO/Xamarin into a single base of libraries and development tools. However, victory should not be sung too soon. If it is ever feasible, it will still be years before the SDR# executable return key can even be pressed on the Mac keyboard.

P.S: ...and in the meantime?

If the roads thus far to the finish line have been blocked or made too slow and circuitous, radiant enthusiasts and Macintosh enthusiasts can still enjoy the opportunities associated with the development of explicitly native SDR code that is avowedly cross platform or otherwise adaptable with relative ease to the Apple environment.

Along with general purpose SDR clients such as GQRX and such as the recent SDRplusplus (SDR++), the latter explicitly inspired by SDR# even in its modular structure, there are a great many projects on the net dedicated to specific functionalities-first and foremost, decoding of digital modes-now covered by the many software modules developed by "prog "s" of SDR#. These are usually applications and utilities that originated in the Linux environment and are redistributed in executable or recompilable form on Macs as well.

These programs, however, are beyond the specificity of this manual, and it is not appropriate to dwell on them here. Suffice it to say that the most motivated and trained users on the use of command lines can find ways to escape from the ancient "radio isolation" of the pupil ("Apple") of their eyes.

Many thanks to the various friends (Andy, Ciccio, Gabriele, etc.) who over time have introduced me to aspects of an O.S. unknown to me.

I recently read a post in the forum...

I finally found the configuration that works on my Mac M1:
- ✓ *Airspy HF+ Dual/Discovery*
- ✓ *SDR # studio v 1.0.0.1899*
- ✓ *Parallels 18.0.2 running Windows 11*
- ✓ *Mac OS 12.6 Monterey*

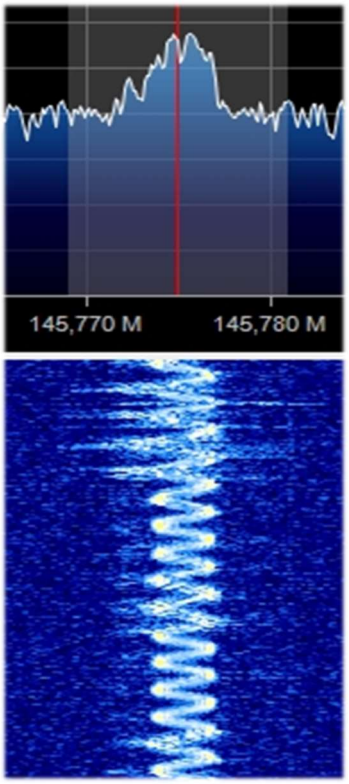
Things I didn't understand...

Over the years I have received many signals like the previous one and very few are easily identifiable at first glance. It must be said that not all signals emitted in HF/VHF/UHF are on purpose because many of them are caused by the most disparate radio interference and disturbances: noise generated by the internal circuits of the receiver itself or by the USB or power supply of your computer, industrial equipment or the many poorly designed or poorly shielded domestic equipment, but also occasional natural phenomena of various entities (solar storms, ionospheric propagation, etc.).

Today, thanks to the use of SDRs, it is possible to have a clear graphical representation of these phenomena and with the use of waterfall, it is possible to visualize and analyze in real time all the received signals including interferences. But being able to make a cataloging is quite difficult if not impossible. Sometimes even a simple switching power supply of low cost radiates signals difficult to identify if not turning off one at a time the various utilities (but what if it were our neighbor?)

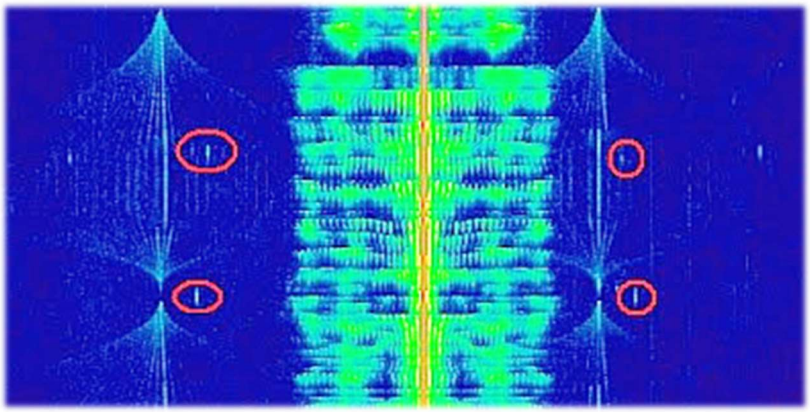
On the net sometimes you come across similar screenshots received from guys maybe from the other side of the world, but no one has yet assigned a unique name, there are those who call them **Squiggles** or **Doodles** or **Ladders** but in the end they are the same things... *What do you think can be a new form of radio listening or why not of what I define "**Waterfall Art**"? Will you give me a hand to collect and try to catalog the most curious and strange?*

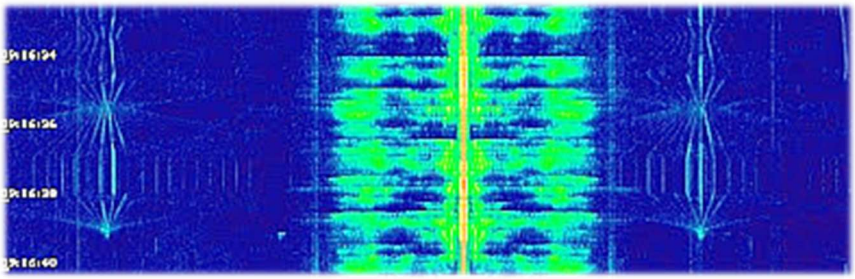
Very unstable speech transmission on 145 MHz



MW signals from the ...Umbrella Corporation?

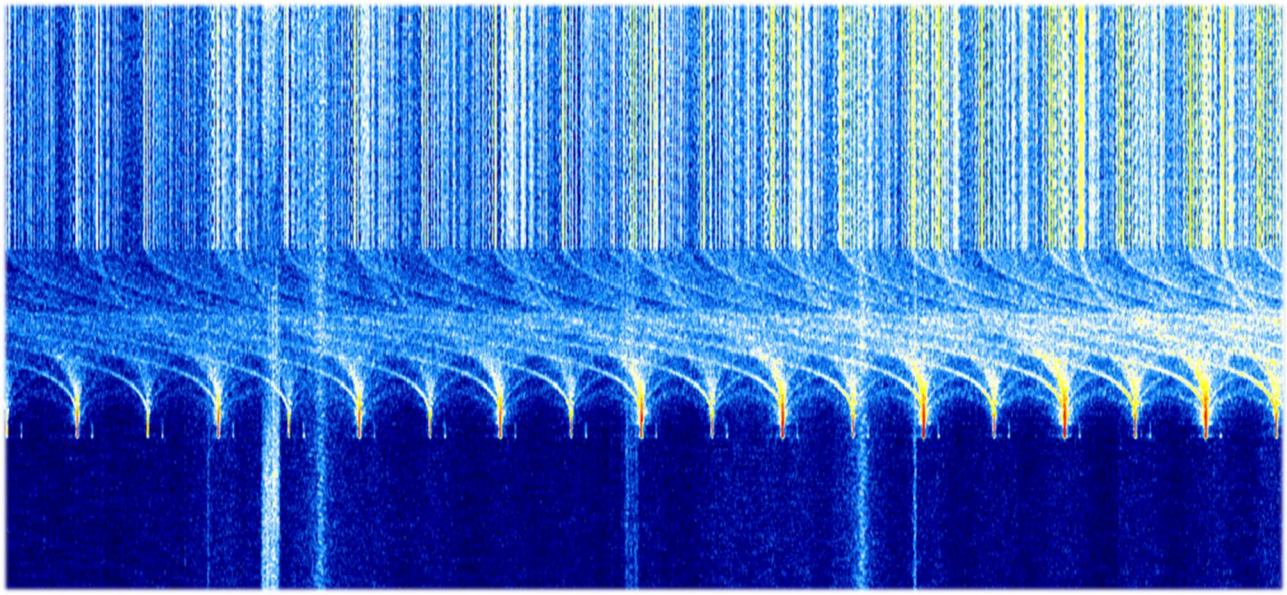
While listening to my local medium wave station I came across some unknown specular signals in the 999 kHz vicinity. Demodulated in USB by ear it sounded like a low audio note that gradually extended its frequencies to open up like a digital umbrella. I had to increase the contrast of the following images a lot as it didn't appear too sharp on screen. After a few seconds the cycle changed and the umbrella closed, leaving a further trace: a very short, higher-pitched audio note that I have circled in red. In about 100 kHz bandwidth the signal was visible 7 times every 16 kHz exactly... Curious indeed, and the definition of the "Umbrella Corporation" - the fictional biotechnology company featured in the Resident Evil video game series - came to my mind.



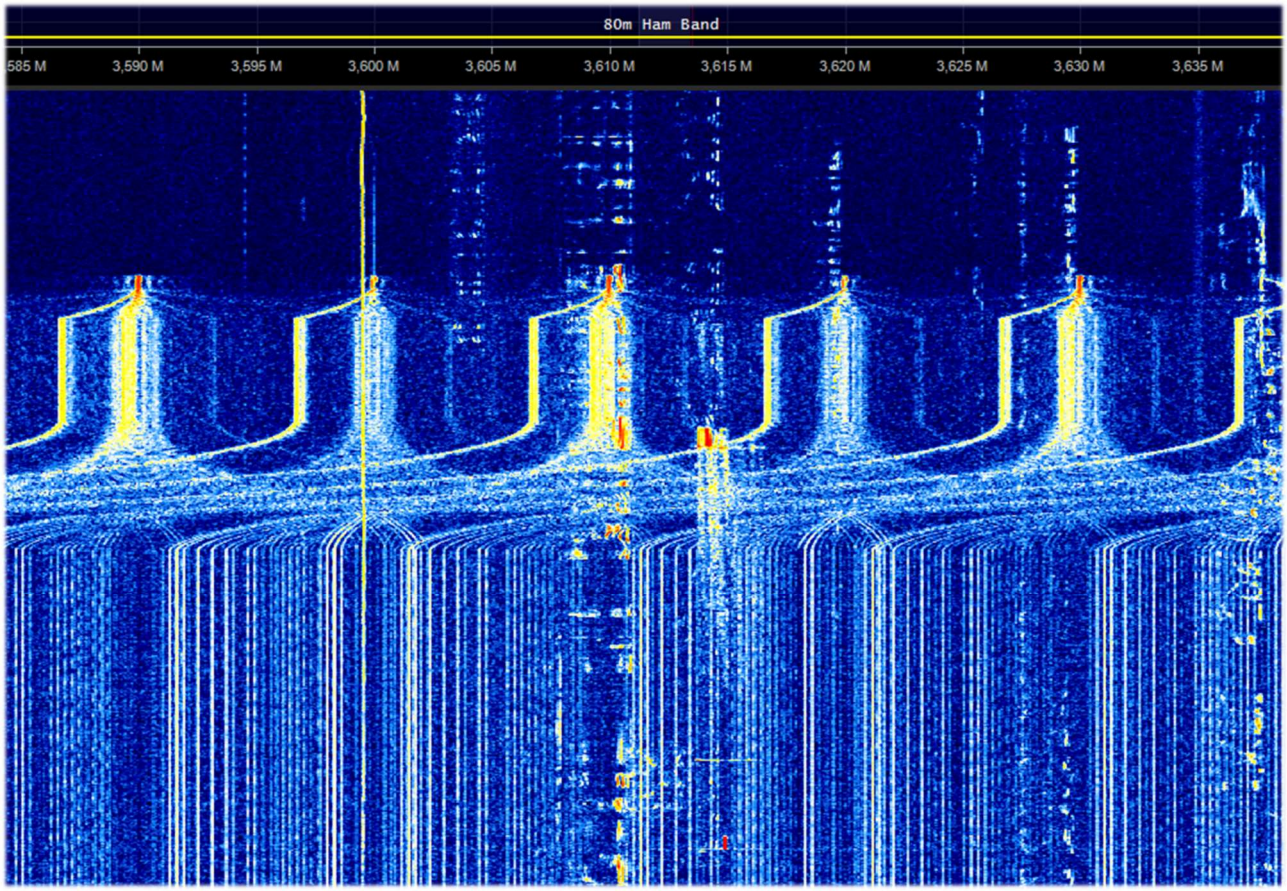


Also this noise has been raging for years in all my HF: it is a cycle of several seconds that opens and closes with the following signals matrix...

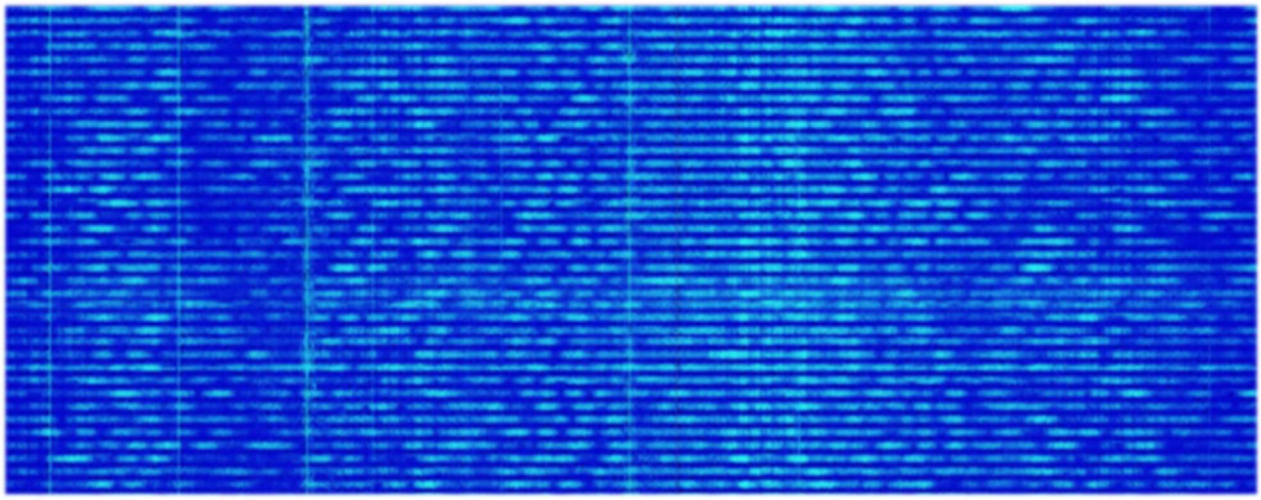
OPENING



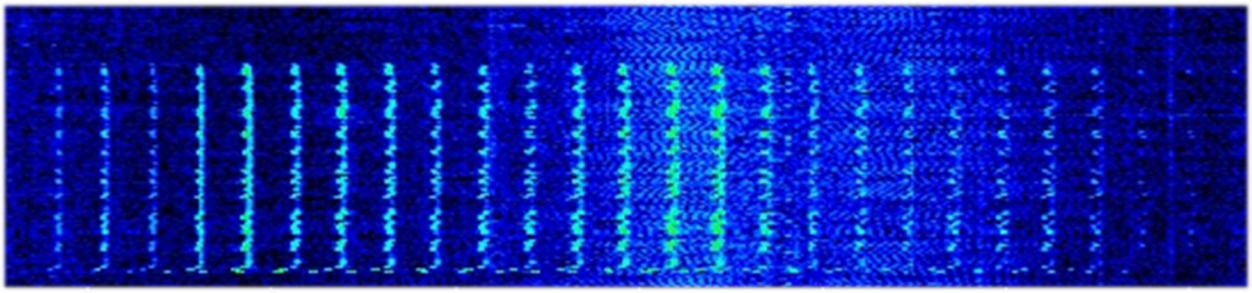
CLOSING



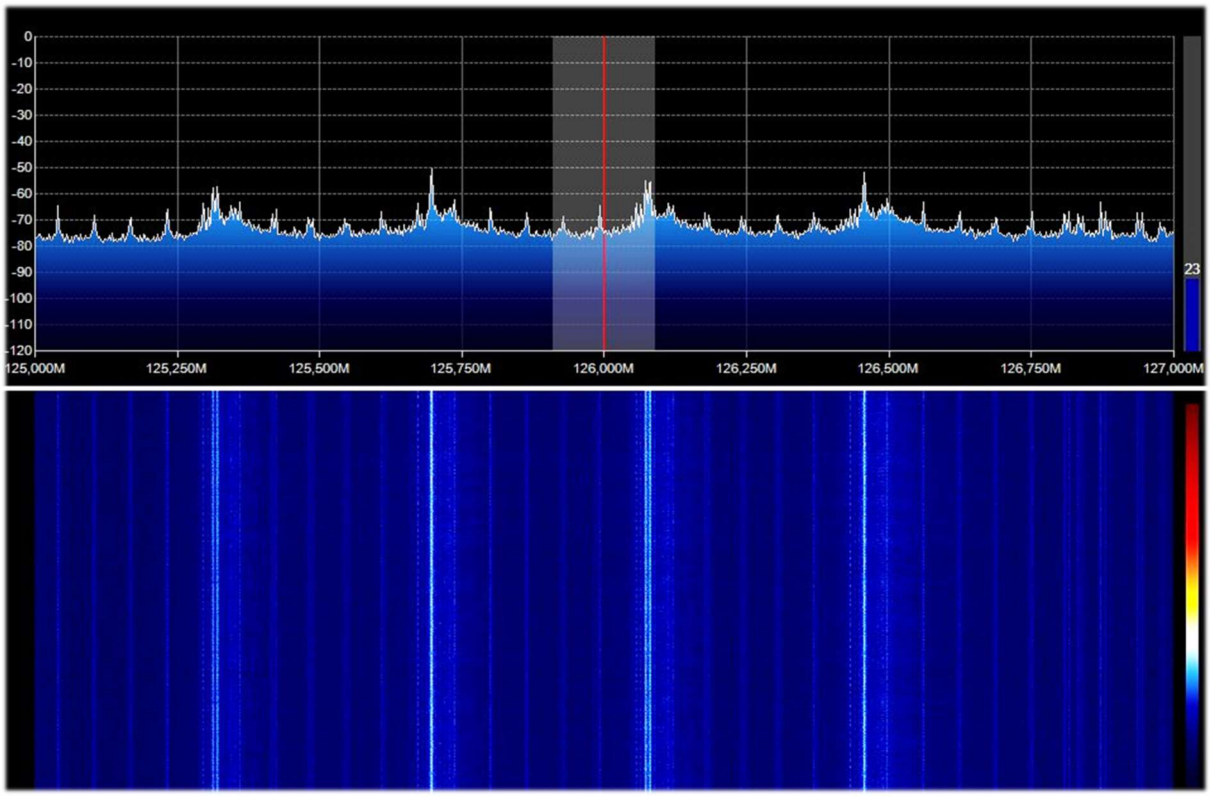
A strange and dense array of signals appeared on HF for a few days between 11 and 13 MHz, perhaps one of the many OTH - Over the Horizon radars?



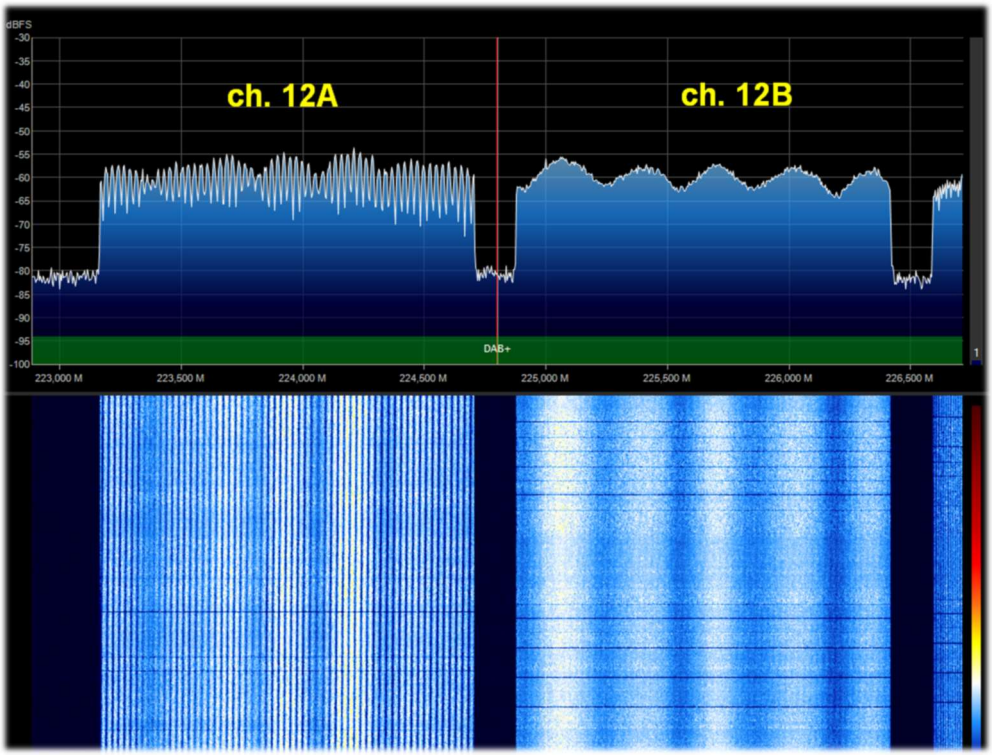
Continuous HF noise from 1.0 to 5.0 MHz from my Atlantis desktop PC power supply.



What about these? All noises coming from the USB or the laptop's internal power supply?

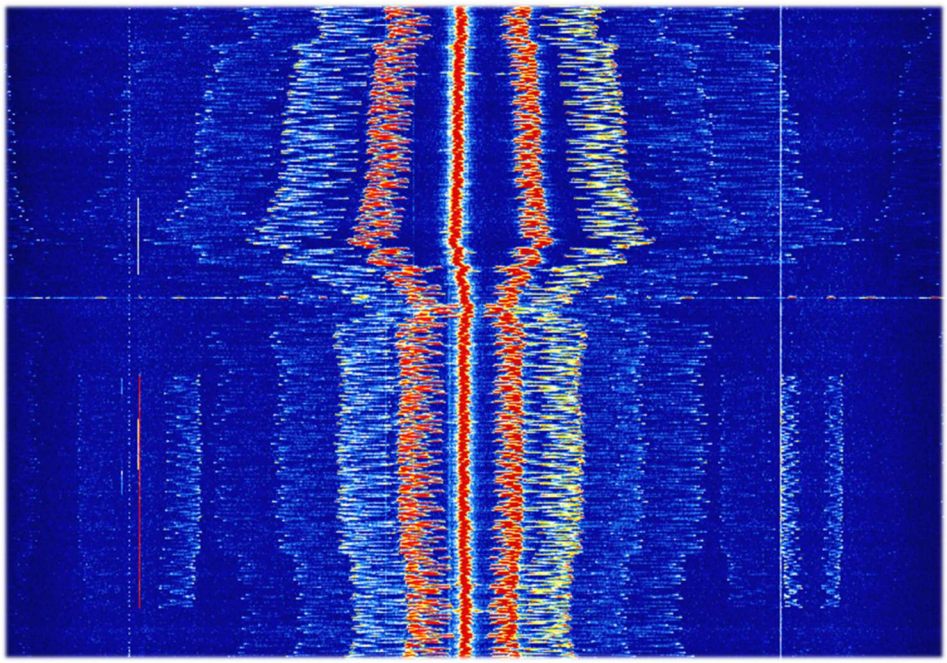


Here everything is ok: …DAB+ signals, but who knows why with such a different multiplexing?

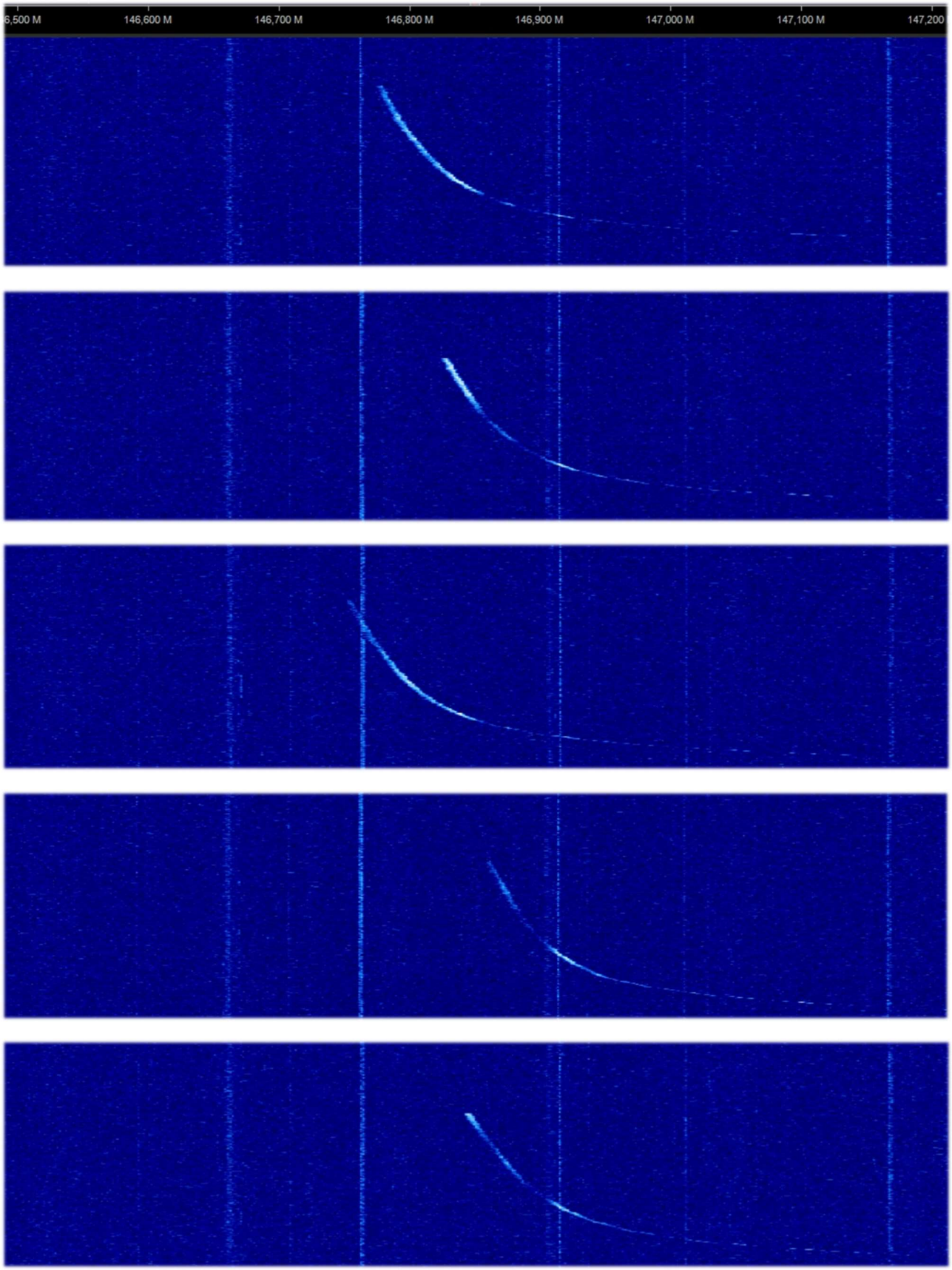


My friend Claudio, a technical expert in the field, explained to me that you cannot see differences in the multipath in the spectrum because there is a scrambling signal that makes them all visually equal. However the difference in the shaping of the signal amplitude depends either on the multipath or on the composition of two or more signals at the same frequency that are periodically summed with a longer or shorter period depending on the difference in path or phase.

Here the problems in UHF return... with these unknown 'sculptures' of pure noise!

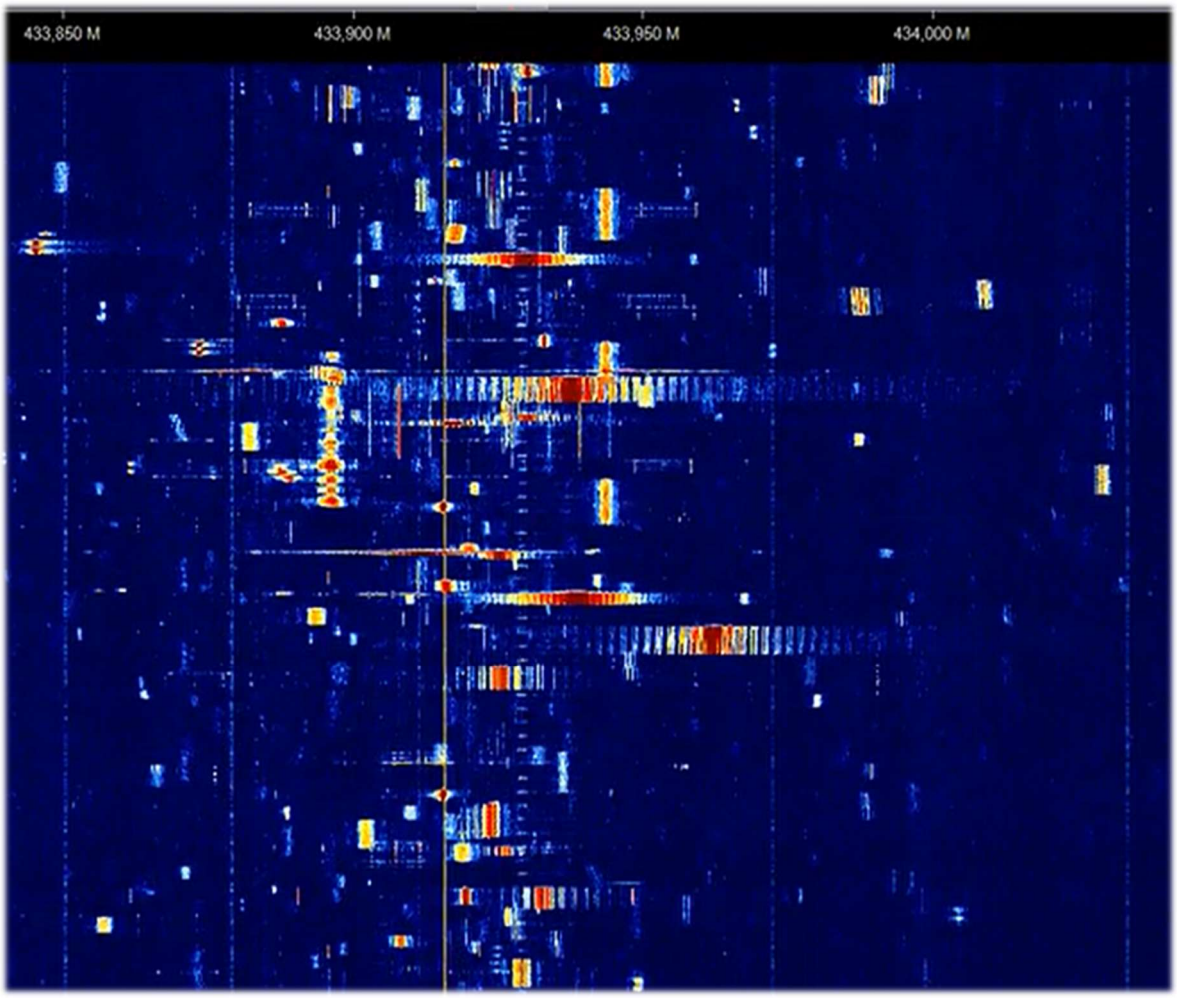


And who generated these curious VHF "whiskers" with signal ranging from 147.1 to 146.8 MHz captured in the time frame of only five minutes? (October 1, 2021 at 13:30 UTC)

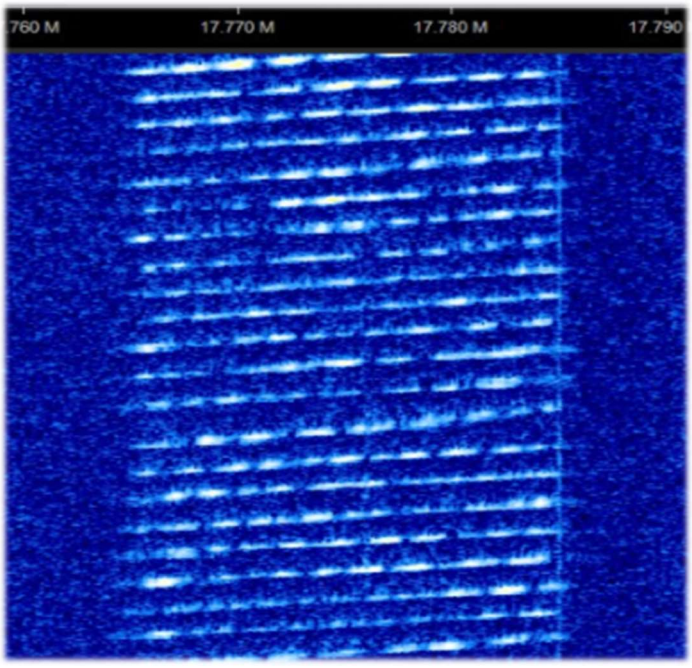


Revealed

A varied palette of digital bursts on 433 MHz, but how many are there…?!

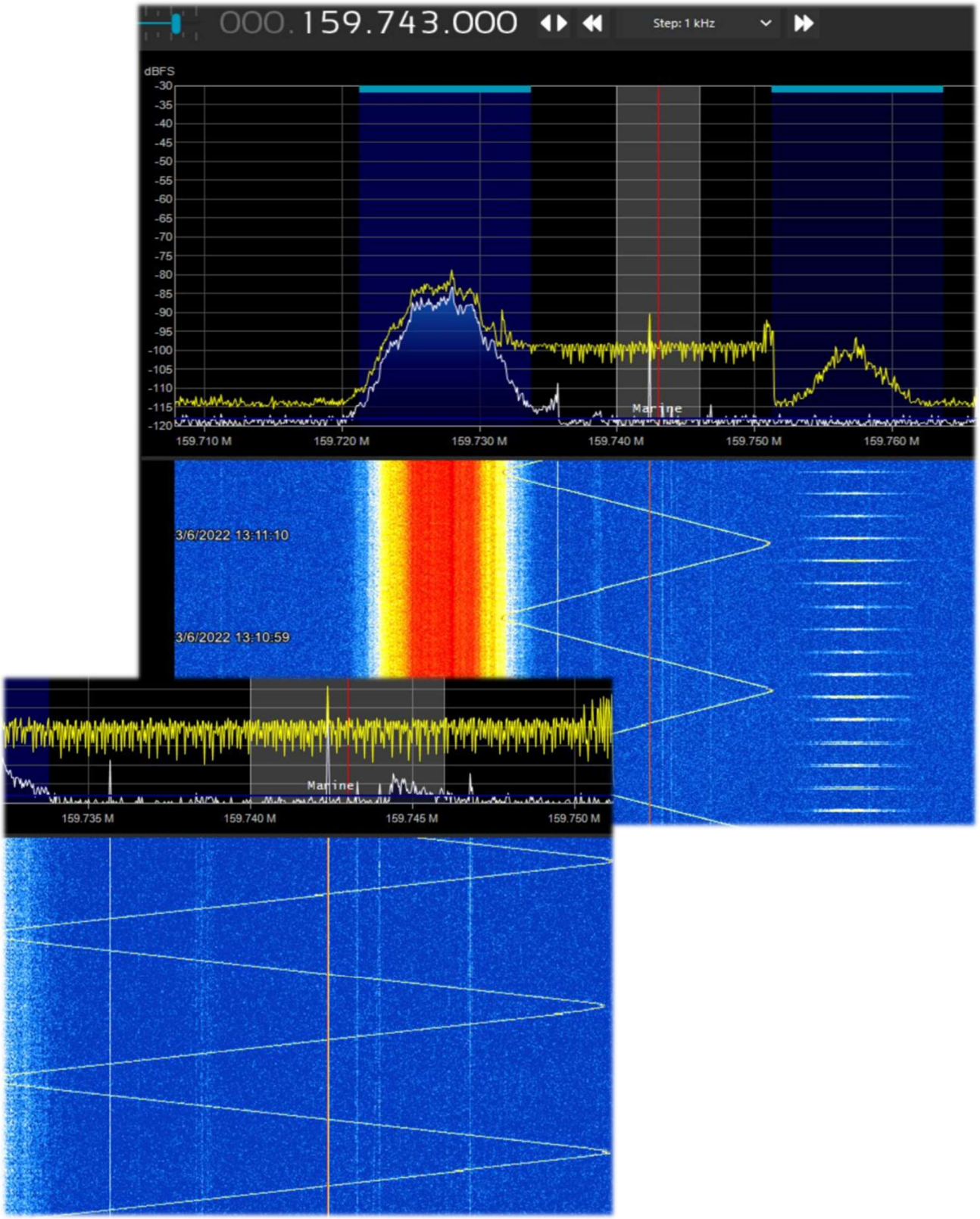


Certainly another of the many OTHRs on HF (here on 17 MHz, 16 meter broadcast band)

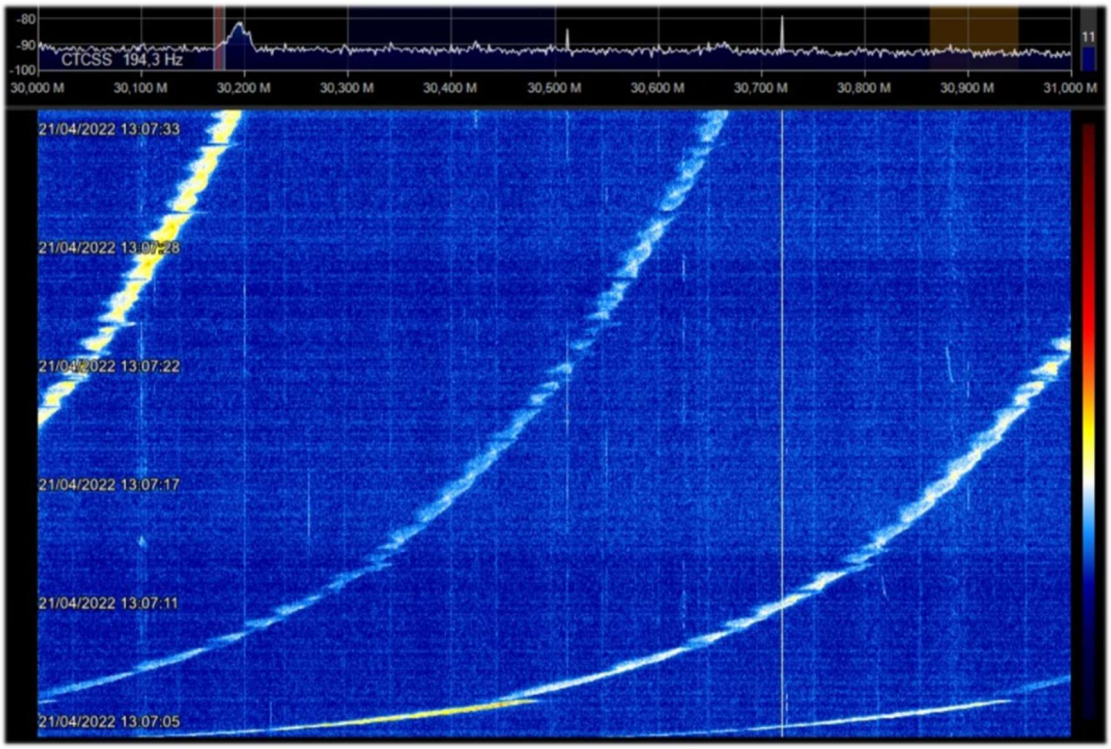


Lee (Maine, border with Canada) sent me these two screenshots of a very strange signal detected on 159 MHz.

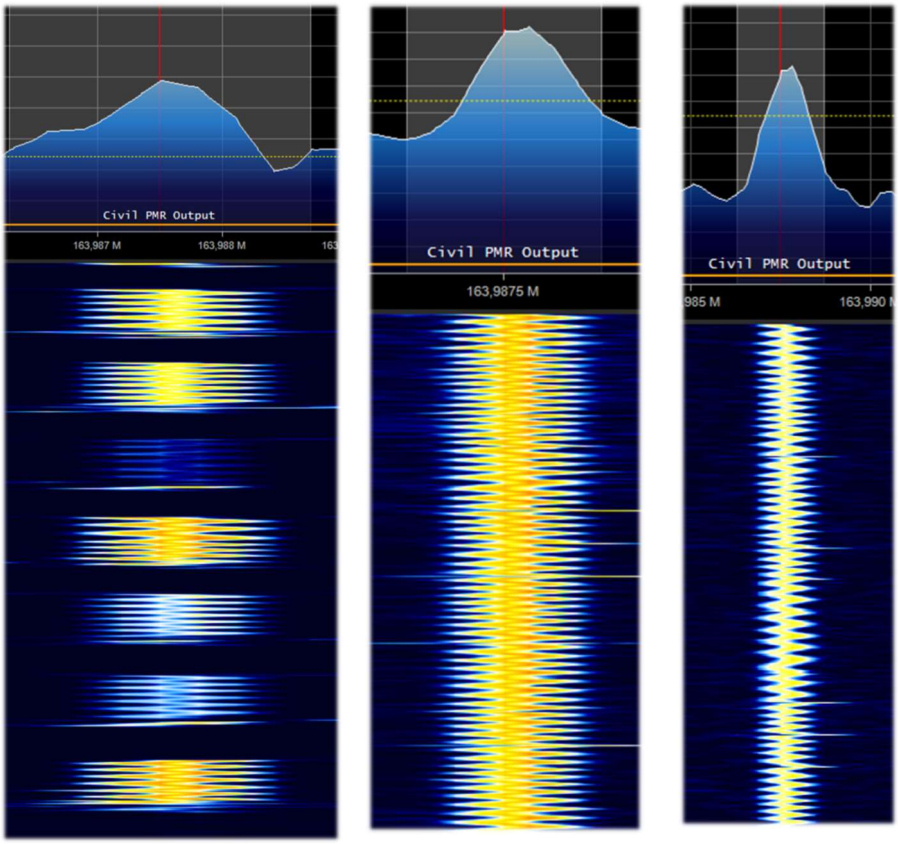
It lasted about half an hour, not visible anywhere else in the spectrum. You can see that the audio signal moved rapidly in frequency in a cyclic fashion and then reversed in the narrow range of the 20 kHz BW, creating that curious jagged appearance.



Roberto (Italy), a dearest guy and fellow radioamateur, sent me the following screenshot (April 21, 2022) detecting curious increasing signals in the 21/50 MHz range (but after a while the frequency stabilizes) also indicating that usually induction welders are retrograde. These instead go forward...

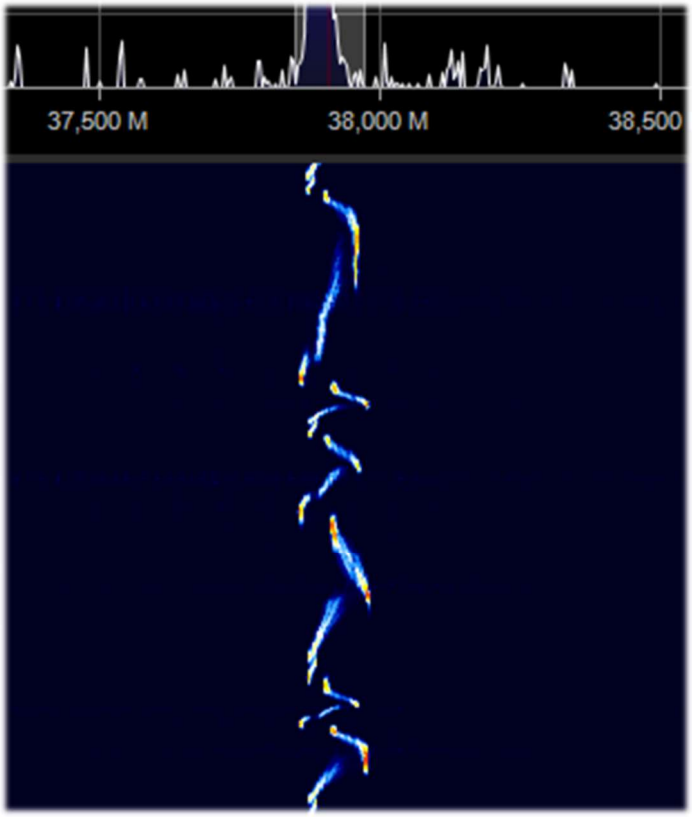


(Italy). A net of local vhf links with a curious problem of hooking/unhooking to the system and with only carriers with hiccupping audio and with continuous triggers that went on for half a day...

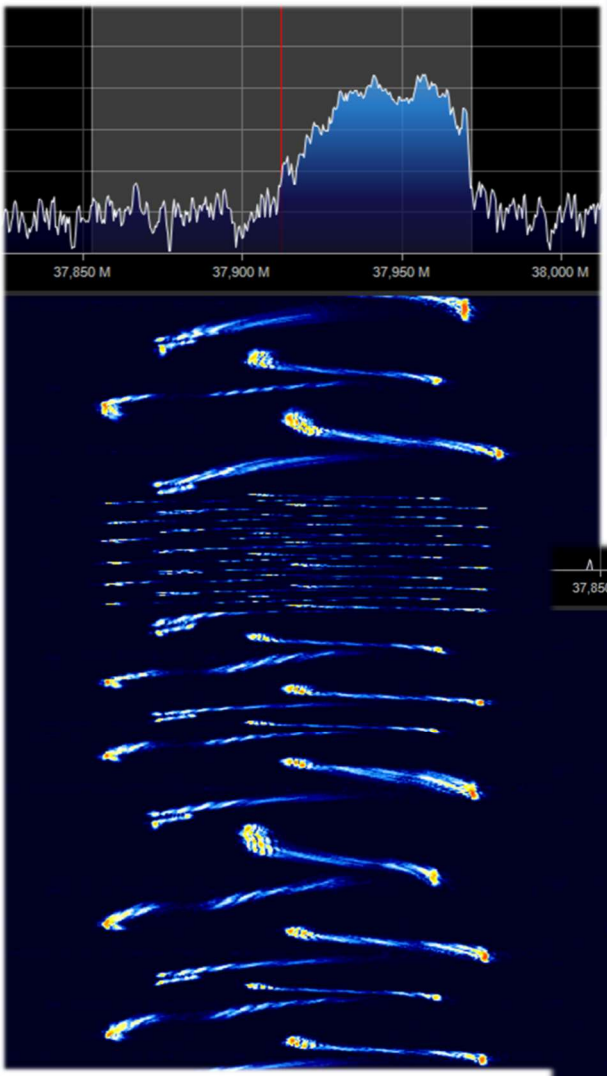


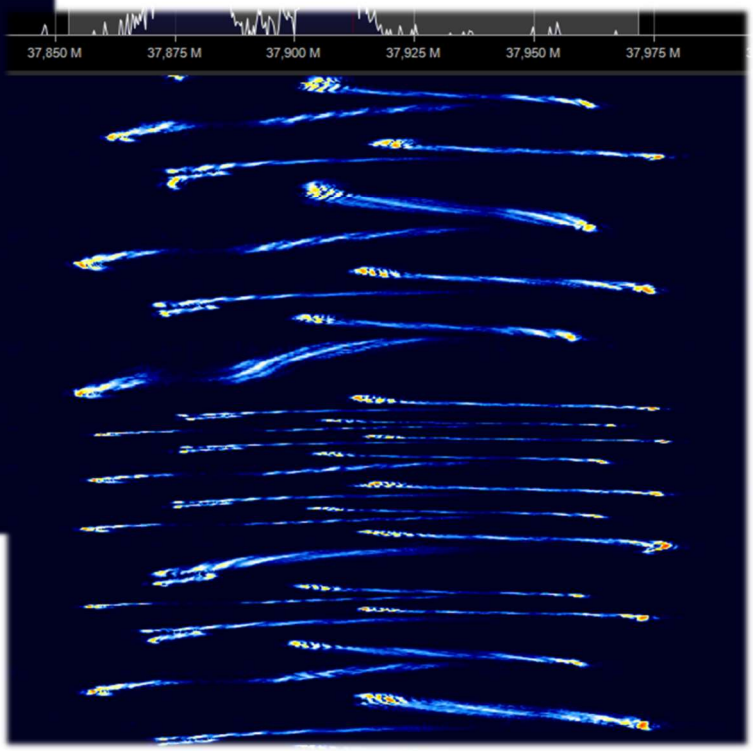
(Italy). Another of my virtual sculptures collected from the ether, perhaps thanks in part to being in a chaotic inner city!

The signal is present fairly strongly at the beginning of 38 MHz with a bandwidth around 120 kHz. Perhaps some switching power supplies in full operation?



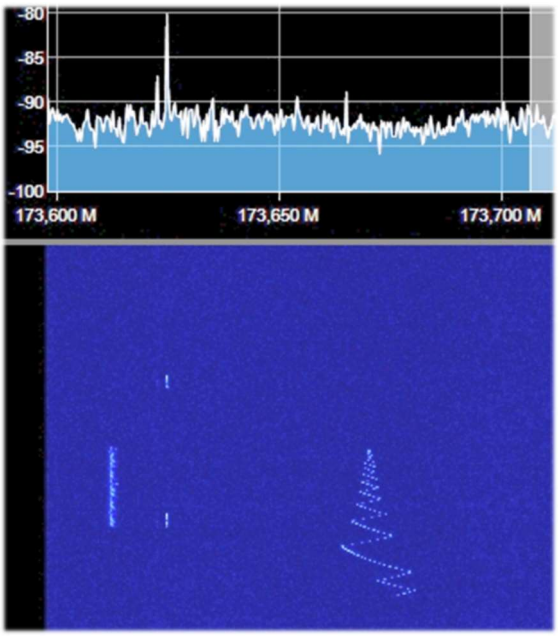
Here are some zooms of greater detail...



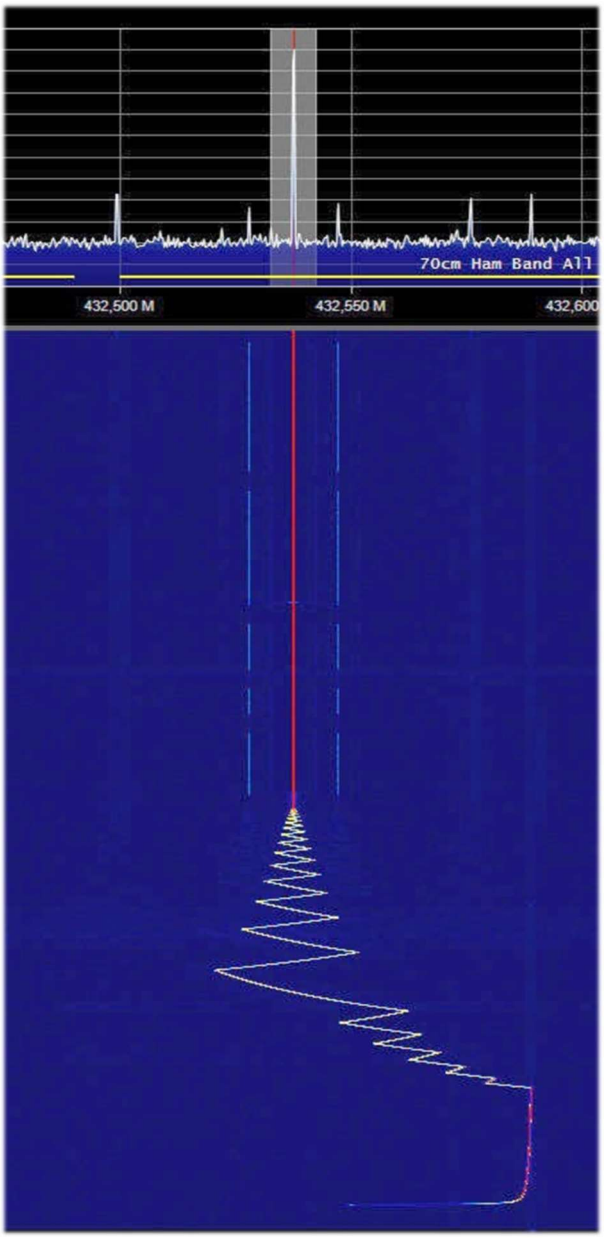


Revealed

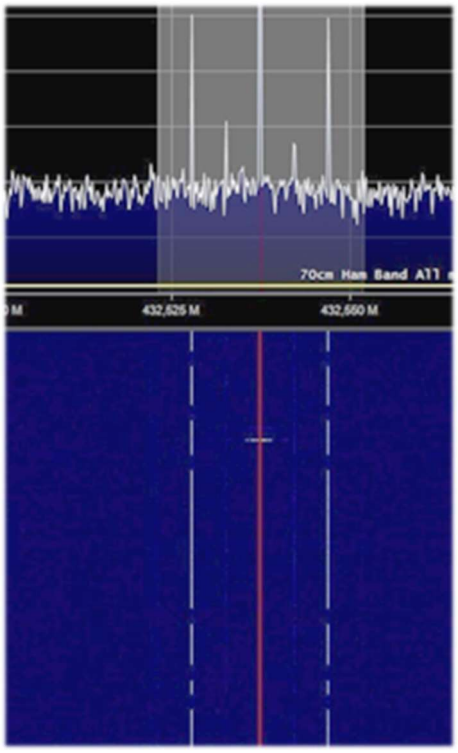
Years ago I came across this curious signal on VHF that had drawn a "Christmas tree" in the spectrogram, unfortunately without discovering the cause until now…



In fact, I just found one that is practically the same and has a much greater signal strength and detail of details.



It begins with a variable note that goes down and up in frequency in a very few seconds, until it settles into what might appear to be an on/off note (with AM and morse center carrier) classic of NDBs still found in longwave...



It doesn't!!!

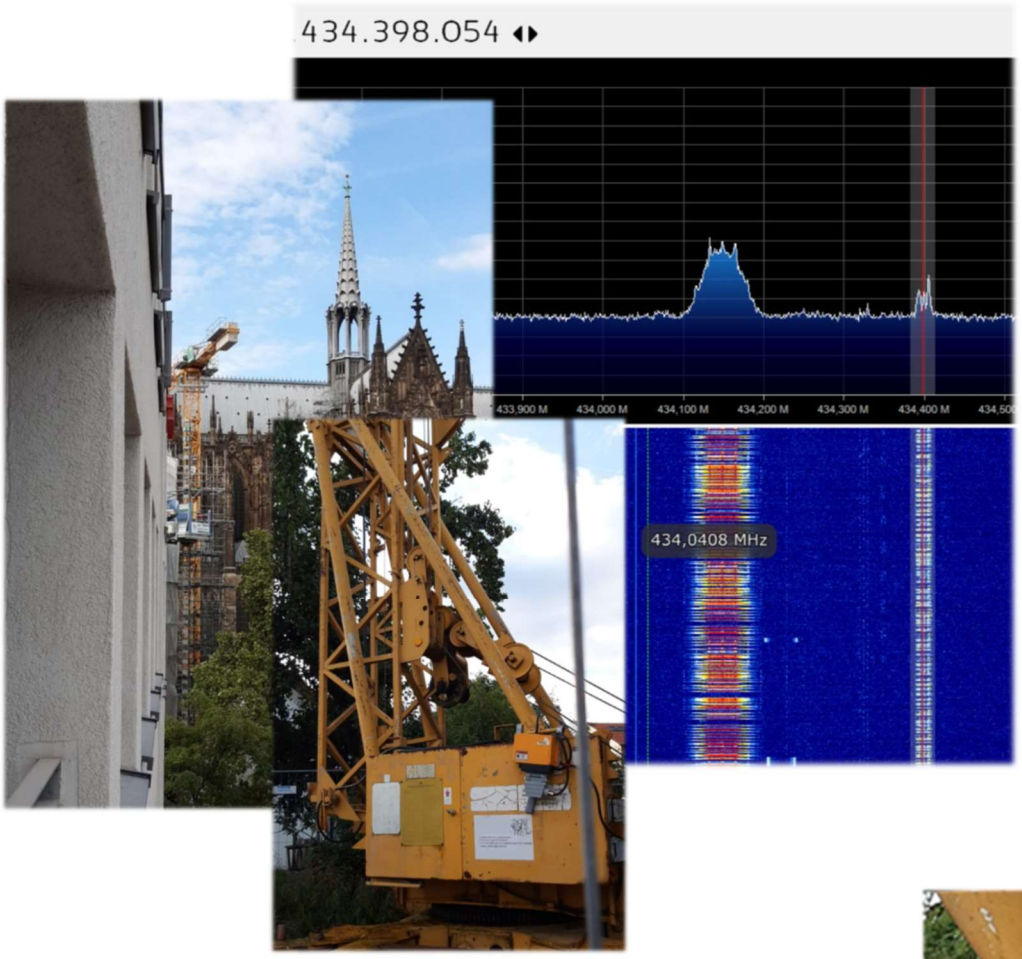
Curious to know what generated it?

It comes from turning on a digital receiver from the AOR that I have on my radio shack on the table. As soon as it is turned on (without any antenna attached, multicoupler or anything else) it initially generates that little tree and then continues with that very curious NDB-like signal...

Austria, Germany, Italy ...there is always a crane nearby!!!

Revealed

Also these digital signals, which have been raging for some years in UHF in the 433/435 MHz band and confirmed by several friends and colleagues, I was finally able to identify them accurately...



The field is industrial radio remote controls that enable mobile, wireless control of machines, vehicles, and site equipment. The range, safety and reliability of the frequency, as well as the ergonomics of buttons, joysticks and housings are therefore crucial for safe and efficient remote control.

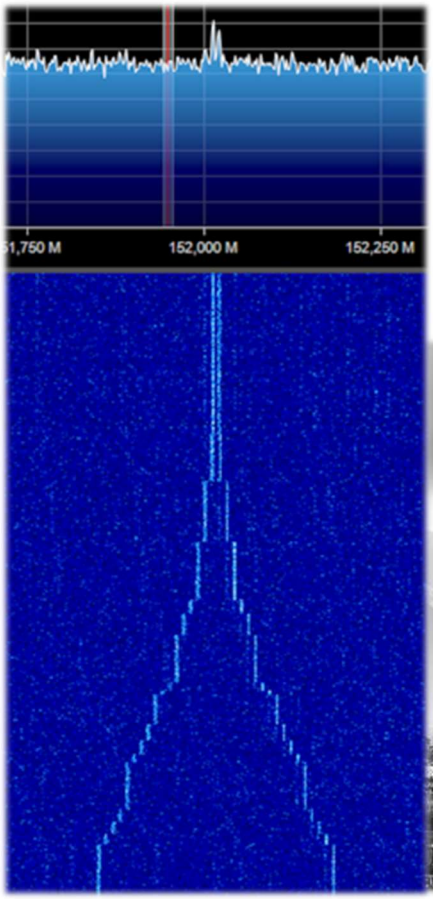
Thanks to Google, I was able to learn that the transmitter sends encrypted data packets to the receiver (like the one on the side) by pressing a single button, switch, or turning a potentiometer. The data packets are first checked for security code by the radio module in the receiver, then evaluated and finally acted upon and also displayed on an alphanumeric LCD display.

There are many manufacturers, each with their own standards and operating frequencies but all traceable to the ranges: 433/435 MHz, 868 MHz, 915 MHz and 2.4 GHz (at least for EU countries).

There are not only cranes to take advantage of these systems... but also mega concrete mixers with articulated cement transfer systems, huge cranes/cranes at ports for loading/unloading containers, etc. etc.

It looks like the Mole Antonelliana in Turin...

(Turin - Italy). Every now and then I come across really peculiar and unique curiosities in our SDR world... like this spectrogram that drew a kind of Mole Antonelliana, a monumental building in Turin, located in the historic center, a symbol of the city and one of the symbols of Italy.

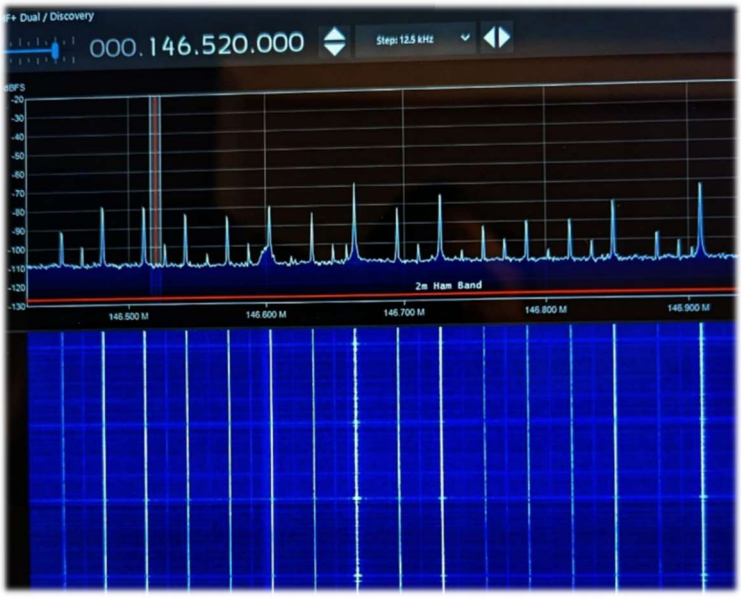


The name comes from the fact that it was once the tallest masonry building in the world, while its adjective comes from the architect who conceived and built it in 1863: Alessandro Antonelli.

Take a look too if it doesn't look a lot like her!!

POE IP camera in VHF…

Revealed



An HF+ Discovery and a YouLoop antenna were able to identify the source of these recursive noises in VHF...

The origin of the QRM is due to a PoE (Power over Ethernet) system.

This technology allows some network devices such as VoIP phones, IP cameras and access points to be powered using the same cable that connects them to an Ethernet local area network. Unfortunately, if shielded cables or appropriate filters are not used, the risk is to receive and display signals like the one depicted opposite.

ADS-B SPY v2.2-RC26

This valuable tool allows the real-time reception of aircraft transponders transmitted on the nominal frequency of 1.090 MHz, i.e. for the acquisition of ADSB by Airspy R0/R2/MINI devices and subsequent visualization on cartography in graphical and tabular format by other software (e.g. Virtual Radar Server, Flightaware, etc...).

Need a dedicated antenna and possibly little and good coaxial cable, but you can initially try with a discone or a bibanda VHF / UHF but better would be to realize the project of a small dedicated collinear copper or brass alloy that offers good gain and reception in the range of a few hundred kilometers ... Please refer to the link: ***http://www.radioamatoripeligni.it/i6ibe/ads-b/ads-b.htm***

On the Airspy website there are brand new updated versions of the ADSB-SPY (for Windows, Linux, Raspberry, Odroid). The Airspy R0/R2 and Mini can be used as high performance ADSB receivers capable of 12, 20 and 24 MHz MLAT. The brand new and original algorithms compare favorably with high-end ADSB receivers turning your Airspy into a self-contained ADSB station with low power requirements.

I will cover the one for Windows here.

airspy_adsb	exe	196.608
flightaware	bat	74
virtualradar	bat	52
libusb-1.0	dll	135.680
pthreadVCE2	dll	61.952
msvcr100	dll	773.968

Once I downloaded the file **Airspy_adsb_win32.zip** I proceeded to unpack the six files in the SDR# directory.

I start in my case by running the Virtualradar.bat file which contains the following line of parameters:

```
start Airspy_adsb -v -e 20 -w 5 -m 20 -l 47806:asavr
```

The meaning of the various commands can be better understood thanks to the help

```
A High Performance ADSB/Mode-S decoder for Airspy
Options:
 -s <serial_number>         Device serial number
 -t <timeout>               Aircraft timeout in seconds (default: 60)
 -g <rf_gain>               RF gain: 0..21 or auto (default: auto)
 -f <bits>                  Forward Error Correction (FEC) bits (default: 1)
 -e <preamble_filter>       Preamble filter : 1..60 (default: 4)
 -C <target>                CPU processing time target (percentage): 5..95 (default: disabled) (adjusts preamble filter while running)
 -E <max_preamble_filter>   Maximum preamble filter when using CPU target 1..60 (default: 60)
 -P <non_crc_preamble_filter> non-CRC Preamble filter: 1..preamble_filter (default: disabled)
 -w <whitelist_threshold>   Whitelist threshold: 1..20 (default: 5)
 -c <host>:<port>[:format]  Add a Push Client
 -l <port>[:format]         Add a Listener
 -m <mlat_freq>             MLAT frequency in MHz: 12, 20 or 24 (Airspy R2 only)
 -n                         Enable Verbatim mode
 -x                         Enable DX mode
 -r                         Reduce the IF bandwidth to 4 MHz
 -R <rssi_mode>             RSSI mode: snr (ref = 42 dB), rms (default: rms)
 -D <comma separated list or "none">  ignore these DF types (default: 24,25,26,27,28,29,30,31)
 -b                         Enable Bias-Tee
 -p                         Enable Bit Packing
 -v                         Verbose mode
 -h                         Display this help screen
Available output formats:
 * AVR         - Raw AVR format
 * AVR-STRICT  - Raw AVR format with only CRC valid frames
 * ASAVR       - Raw Airspy AVR format
 * Beast       - Raw Beast Binary format
```

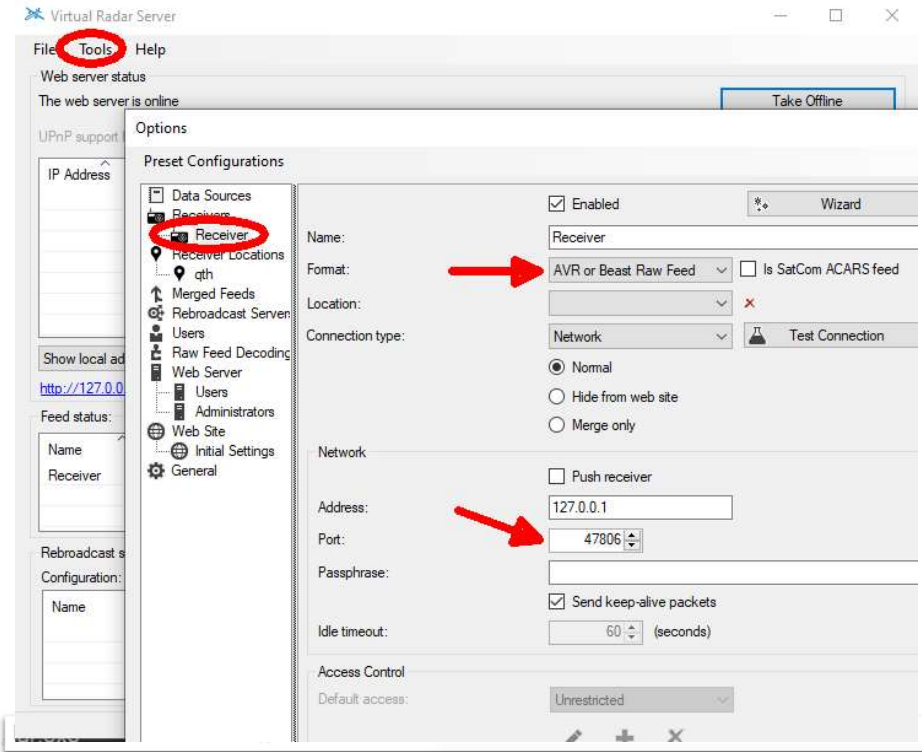
It will then open a window like this, while in the meantime we're going to install and configure the free Virtual Radar Server software:

```
airspy_adsb v2.2-RC26
Listening for asavr clients on port 47806
Acquired Airspy device with serial 644064DC2E836BCD
Decoding started at 20 MSPS (Gain: auto; Preamble Filter: 20.0)
```

https://www.virtualradarserver.co.uk/Download.aspx

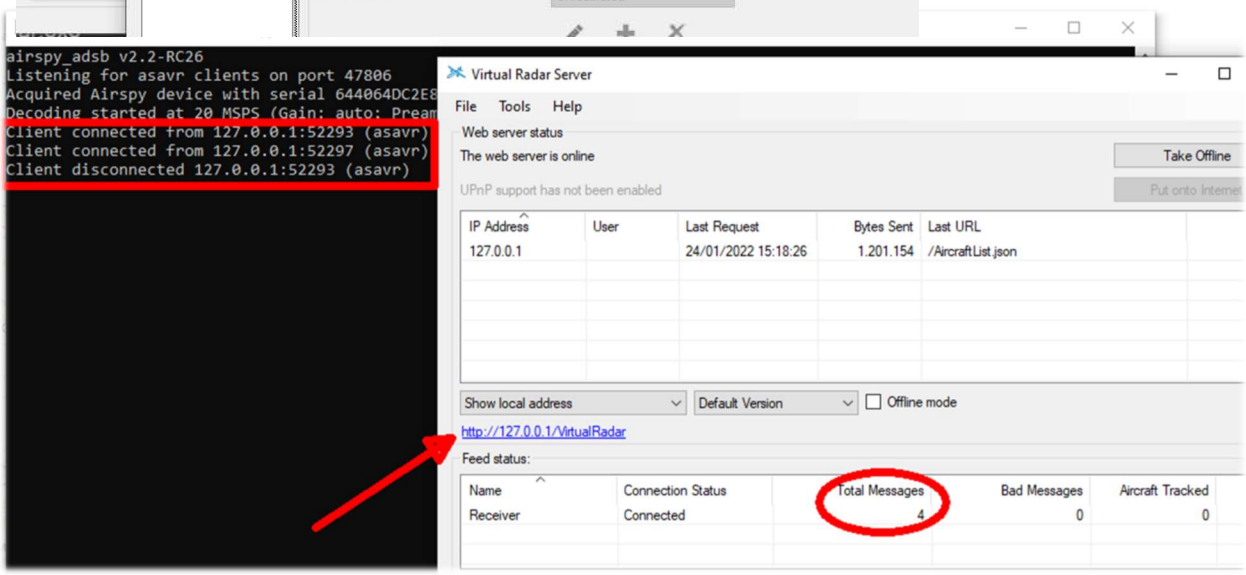
```
*5D896408971F97;011A9470;06;0736;
*20001490D60A4F;011CA3E3;06;076E;
*28000017562C3A;011E55E9;06;07CE;
*5D896408971F97;011E798F;06;06BE;
*20001490D60A4F;0122B27E;06;069A;
*5D896408971F96;01263E06;06;0772;
*5D896408971F97;012A1BF7;06;0747;
*A0001490CC300030A401803BAE8E;0140AF49;06;05F8;
*A8000017A29A5733615C2330159B;0141EC1D;06;0605;
*5D896408971F8A;015E3157;06;0686;
*A0001490201541 33E58820385DE3;0162D911;06;0659;
*5D896408971F8A;0163146C;06;06E7;
*A8000017A29A5933614423B2C94F;016434F8;06;06AB;
*A8000017CC300030A40180021A67;01644F67;06;05D7;
*28000017562C3A;01668291;06;069C;
```

As mentioned at the beginning, ADS-S SPY receives the data in "raw form" like these on the left... which are then sent to port **47806** of the program that will display them at their best!

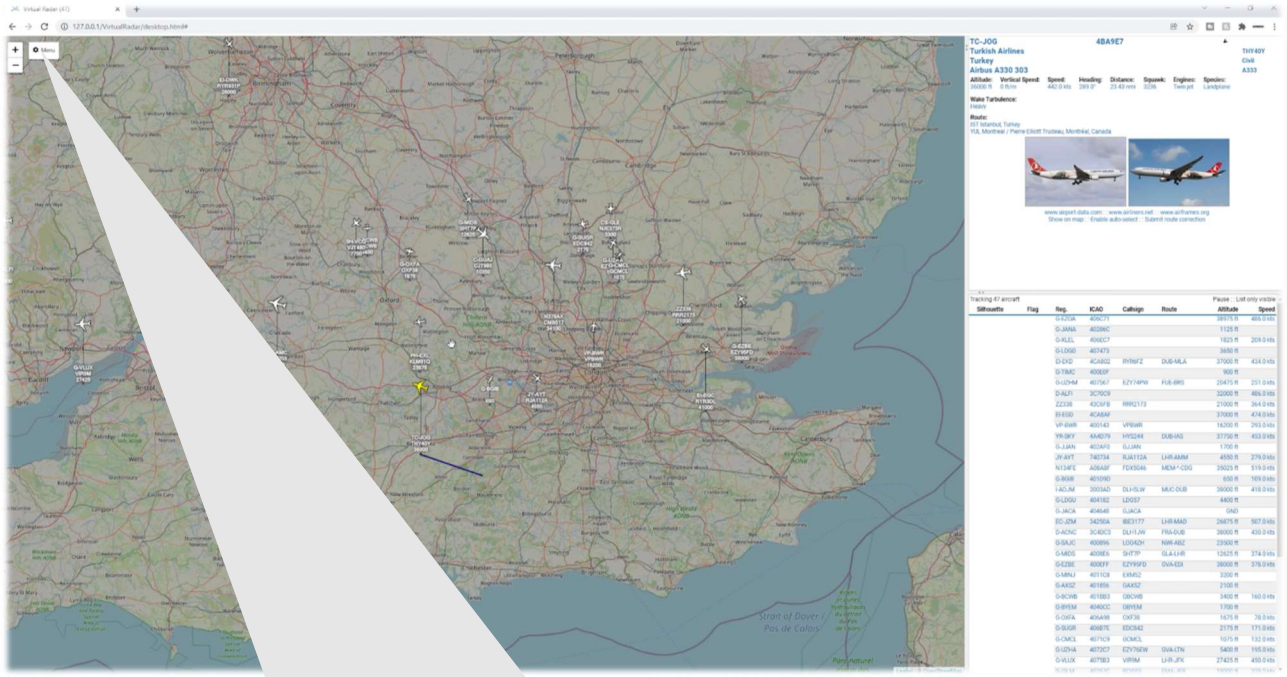


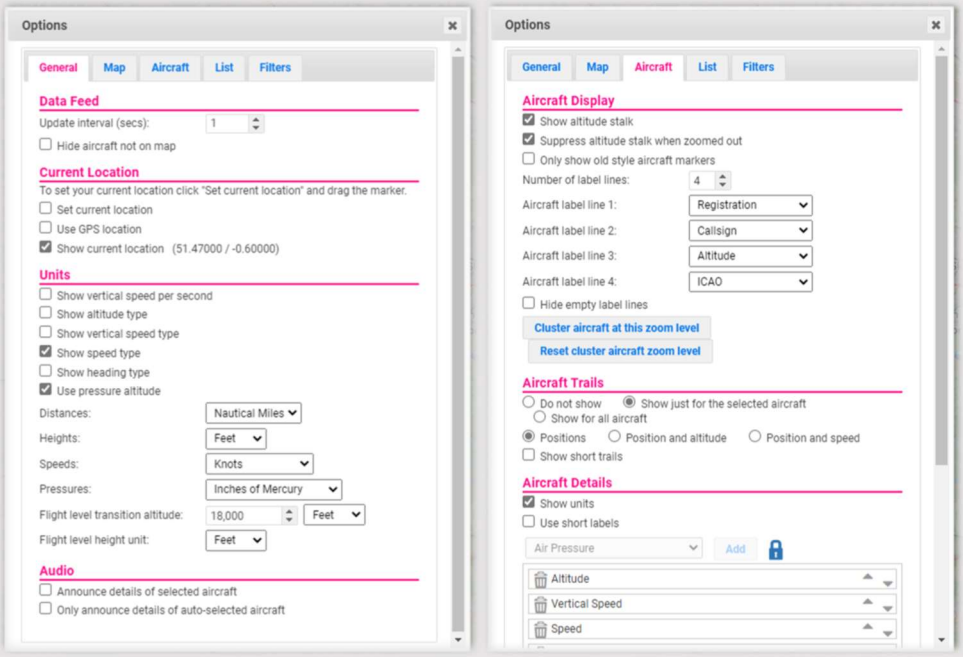
In TOOLS / OPTIONS / RECEIVER we are going to configure the highlighted points...

while in TOOLS / OPTIONS / RECEIVER LOCATIONS I inserted a line with my coordinates.



At this point the windows of the software will begin to populate with data and messages received in real time. Just click on the link highlighted in blue, to open your browser and visualize on a map all the movement in our skies.





Some windows with multiple Virtual Radar Server settings and customizations.

Link download: ***https://Airspy.com/?ddownload=6057***

An excellent Youtube video can be viewed here: ***https://youtu.be/cogNi2lM3gw***

Raspberry Pi

Sometimes it can be useful not to depend on a personal computer running 24 hours a day (CPU/HD/monitor consumption, fan and various electrical noises) or to have the need to remotely locate your receiving station (perhaps in an attic near the antenna cable) and so the use of a Mini-computer can open the way to many projects and applications even in the amateur radio field. The Raspberry Pi (or "single-board computer"), which costs very little and consumes very little power, is ideal both for its high-end technical features and for its extensive software/radio equipment, which also includes all our Airspy devices!

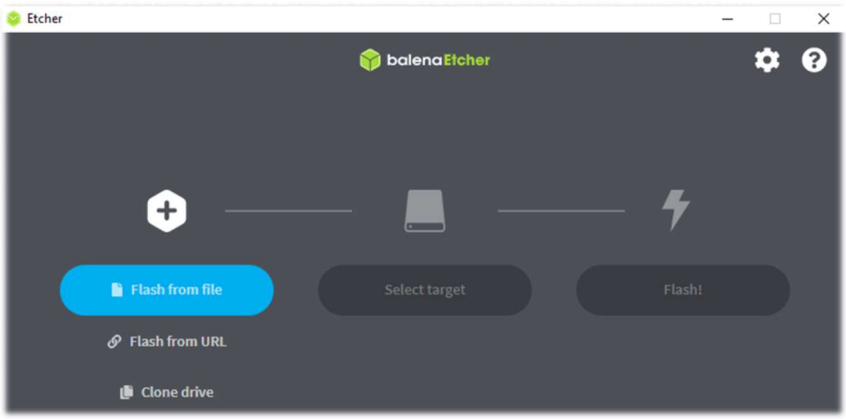
There are certainly other ways, and I will describe one of them in detail, which has led to excellent results, even though it will be the most challenging chapter, and with some possible criticalities. Those who know Linux well can try it differently by following the indications given here: ***https://photobyte.org/raspberry-pi-running-spy-server-as-a-service/***

Prerequisites:

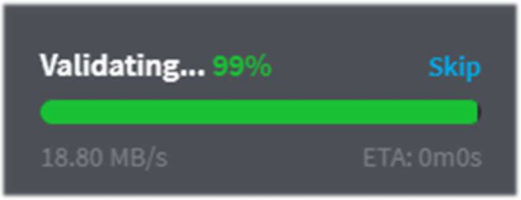
- **Raspberry Pi (with power supply, monitor and keyboard)**
- **a microSD card (of at least 8 GB)**
- **software PiDSR (image file): *https://github.com/luigifcruz/pisdr-image/releases/tag/v5.0.0***
- **portable BalenaEtcher (for flashing the microSD): *https://www.balena.io/etcher/***

This is not the session to describe in detail the various types of Raspberry, there are dedicated sites for every need as well as describing the different distributions and customisations available created specifically for the ARM architecture which is very different from the PC architecture.

In my test, I reused a Raspberry Pi 3 model B that had been lying in a drawer for a long time, in combination with a good external power supply (since it is well known that the Pi3 model is very sensitive to power supply variations). So let's see the bare minimum to get up and running in no time, starting with the software that we are going to download from the links indicated in a directory on our Windows computer (e.g. C:\Temp).



Connect the microSD to the PC and run the opensource and portable software BalenaEtcher. Choose "Flash from file" where in my case I indicated the image file (a "Raspbian" modified with SDR software compatible with each Pi model) named "**2020-11-13-PiSDR-vanilla v5.0.img.xz**", taken from the site of the developer: the radioamateur Luigi Cruz (PU2SPY). Then in "Select target" choose the drive that contains the microSD and finally the third button "Flash!" to start the process. It will take about 15 minutes between writing and verification, do not interrupt it and at the end you will see the following screen:



Make sure that no errors are reported when writing/verifying the image, otherwise reformat the SDcard or use another one. The image has now been created (it is obviously not accessible or viewable via a Windows browser), so it can be extracted from the PC and installed in the Raspberry's slot.

PiSDR's pre-installed software for radio use is very rich indeed, but I have only tried a fraction of it at the moment... The supported SDRs are the following: RTL-SDR, LimeSDR, LimeNET, PlutoSDR, all Airspy (R2, Mini, HF Discovery and HF+), HackRF One, USRP.

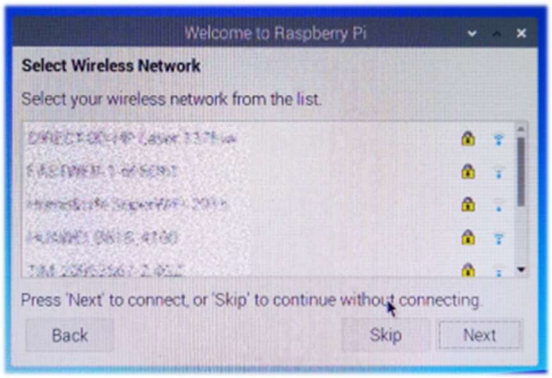
In my case, I connected the Pi to my home WiFi router with a good Ethernet cable (i*f the distance is long, consider a class 7 cable, which is also shielded)* then a video/keyboard and of course an Airspy!

Let's take a look in sequence at the various screenshots that appear on first start-up for configuration:

1) Welcome

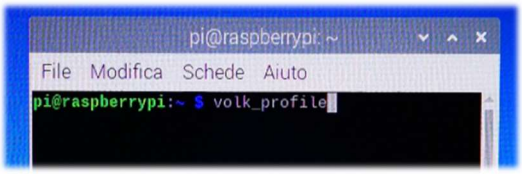
2) Choose country and language

3) Change password

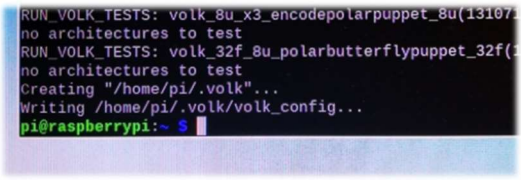


4) Select wireless network

While I left out the window with the request to update the software (which I did not do)...



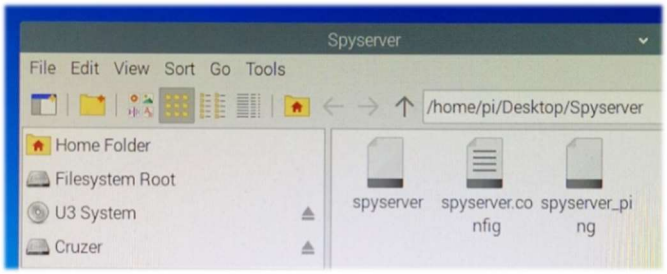
On the developer's website, it is recommended to run the following command from the "Terminal": **volk_profile** which will optimise the system. The Terminal icon is this (the fourth in the top left corner)



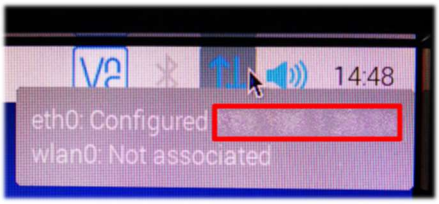
Allow time for the update to take place (several minutes) and then close the Terminal only when you see the usual terminal prompt...

At this point, our new operating system is almost ready. All that remains is to connect to the Airspy website using the "Web Browser" (second icon on the top left), and download to the desktop the file **"SPY SDR Server for 32-bit ARM boards"** from the following link: ***https://Airspy.com/?ddownload=4247***

Once the file **"Spyserver_arm32.tgz"** has been downloaded, I created a folder called Spyserver on my desktop and extracted the three files from it...

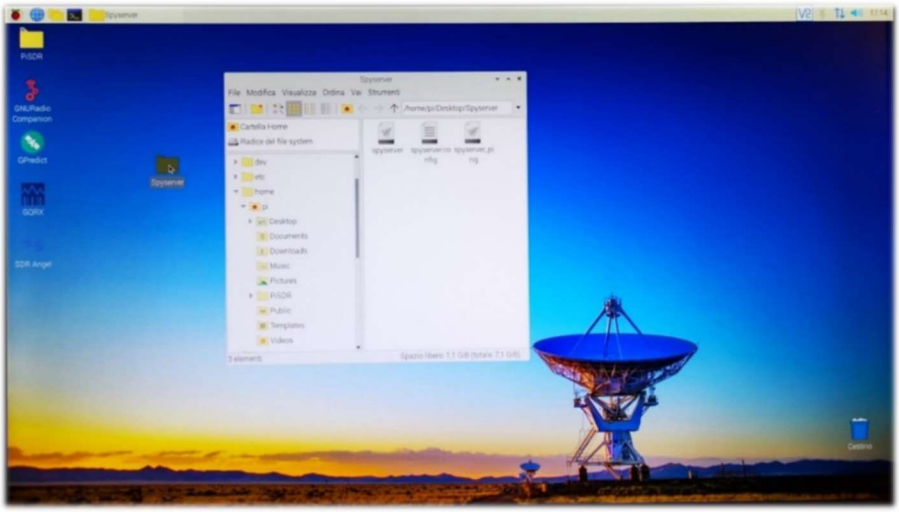


For these operations and to move between the system's folders, I used the third icon in the menu at the top left..

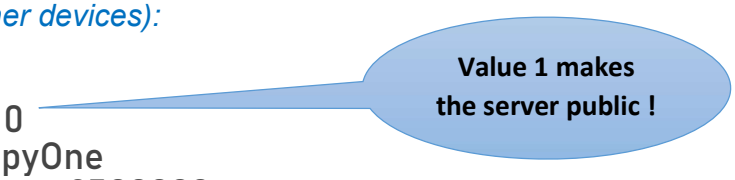


I disabled the Bluetooth icon in the menu at the top right... Instead, I wrote down the number of the IP address assigned by the system that appears by hovering the mouse over the blue Wireless icon at the top right "eth0: Configured xxx.xxx.xxx.xxx". *We will need it shortly afterwards...*

Now we need to edit the file "**spyserver.config**" for our needs. You can either click on it or stand on it and with the right mouse button choose the "Text Editor". We need to edit some values, remove a # (which means to make that line of the script active) and finally save the file, being careful not to change any other parameters for this time.



We will have time later to go back and analyse and better understand all the lines of the "SPY Server Configuration File". These are the lines to be considered for modification and use with an AIRSPY R2 *(read below for other devices):*



bind_port = 5557
list_in_directory = 0
device_type = AirspyOne
device_sample_rate = 2500000
initial_frequency = 101800000 *(optional, it concerns the frequency that will appear at start-up in the VFO of the SDR#: in my case I can see if everything works on the first shot: if in my attic the active antenna and the multicoupler are on and working, if the remote switch is correctly positioned, etc.).*
initial_gain = 10 *(for device: R0, R2, Mini)*

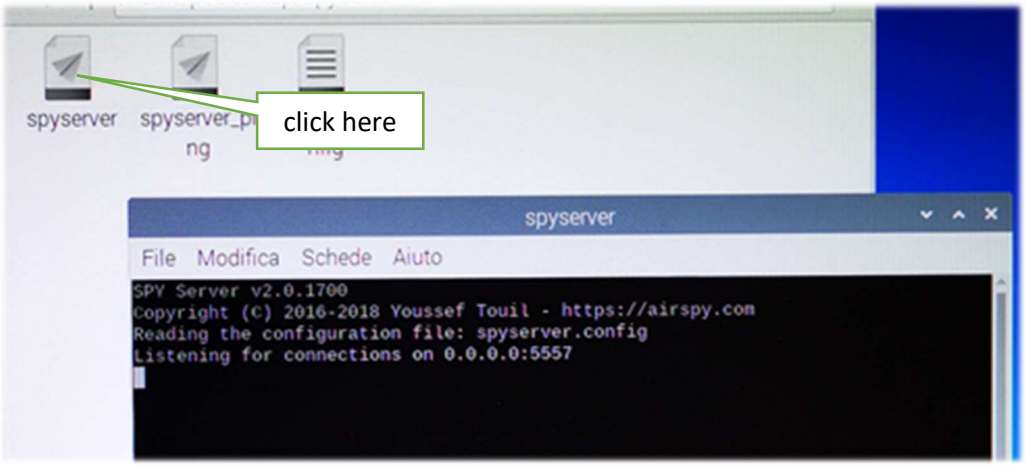
The "Device Type" group has these choices (so indicate your own instead of xxx)

```
# Device Type
# Possible Values:
#   AirspyOne  (for device: R0, R2, Mini)
#   AirspyHF+
#   RTL-SDR
#
device_type = xxx
```

The "Device Sample Rate" group has these choices (indicate value instead of xxx)

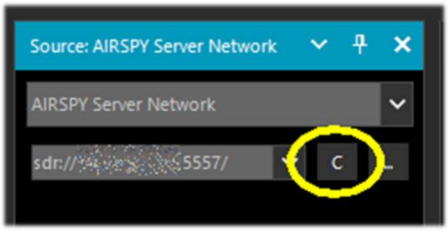
```
# Device Sample Rate
# Possible Values:
#   Airspy R0, R2 : 10000000 or 2500000
#   Airspy Mini   :  6000000 or 3000000
#   Airspy HF+    :   768000
#   RTL-SDR       :   500000 to 3200000
#
device_sample_rate = xxx
```

Now that the file has been properly configured, all that remains is to run it by double-clicking on the "Spyserver" icon and then "Run in Terminal" which will open with a few lines highlighting that it is "listening" while waiting for the client to connect...

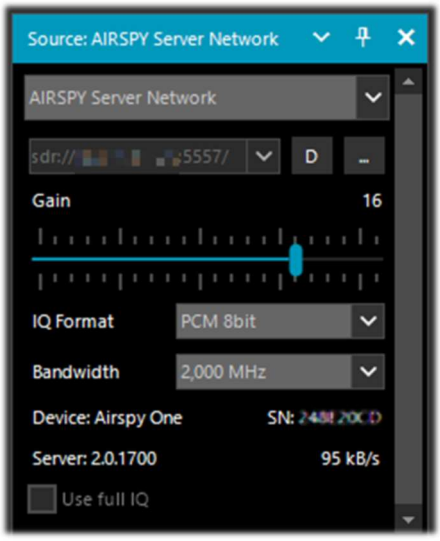


We are finally almost at the end... thanks for your patience!

Now from the laptop that I have decided to use as SDR# Client (wirelessly connected to my home network) it will be necessary to activate the Source field "AIRSPY Server Network" by typing under my ***IP address (previously marked) : port number***, and then press the "C" button.



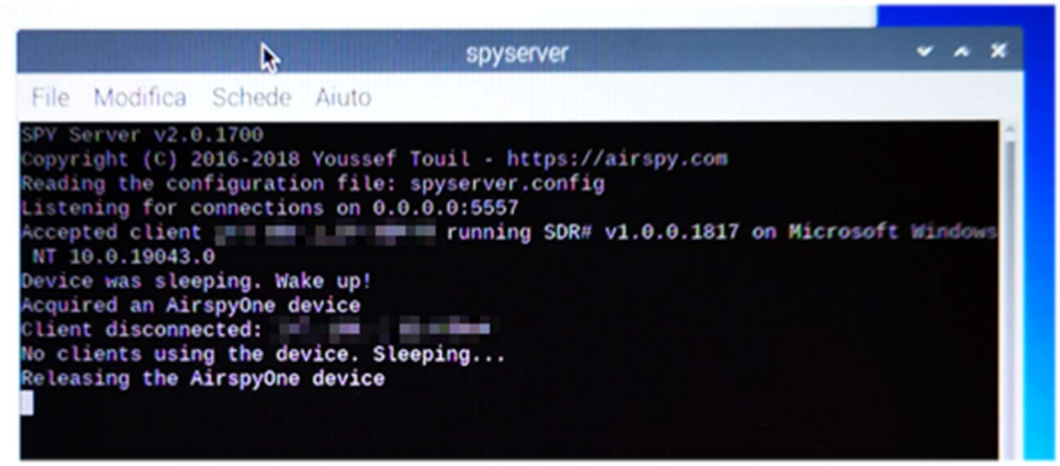
If everything is working properly, the client/server connection is established and the panel populates with more informations. The only thing needed is to adjust the Gain to the right, set the proposed frequency and use it normally: audio, decoding and functionality will be practically the same. For the other options already discussed please refer to the AIRSPY Server Network chapter.



Subsequently, to correctly close the connection it will be necessary to press the "D" button while on the server side, on the Pi, the Terminal will be closed and then the Raspberry from the "Close Session / Stop" menu from the first icon at the top left...

Wait a few moments and then the power can also be switched off...

Looking back at our Raspberry Server we can see that in the meantime the Terminal panel had been populated with more information during our connection.

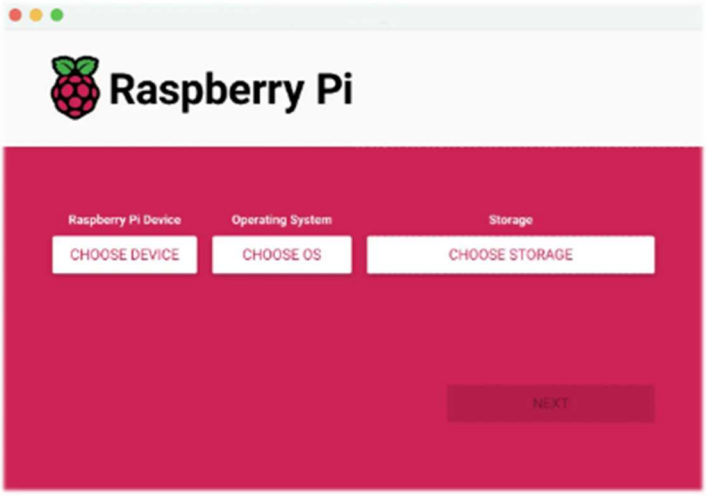


For those who would like to know more, here are some commands to execute in the Terminal that may be very useful:

dmesg	Allows you to read (in the USB lines) the details of your connected SDR
free -h	to see how much RAM in your Raspberry
htop	to monitor system processes in details. To close: CTRL + C
hostname -I	to get the IP address of the our Raspberry

Here a complete list: ***https://www.tomshw.it/hardware/comandi-linux-raspberry-pi/***

Owners of a Raspberry Pi4 (with AARCH64 ARM architecture) can instead download and use the "**SPY SDR Server for 64-bit ARM boards**" from the following link: ***https://Airspy.com/?ddownload=5795***



For those who want to experiment, it is also possible to install a new Raspberry Pi OS (Raspbian) by simply connecting it to the local network and starting it without the presence of the micro SDcard..

After a few moments, a wizard will start (I then inserted the SDcard) that allows you to choose: your device, the operating system/architecture to be installed, storage, language, etc... following the few simple steps suggested.

What if we just wanted to update the firmware of our Raspberry instead?

First, it is advisable to make an external backup of the micro SDcard by gently pulling it out and using an adapter.With Windows, I suggest using the freeware software **Win32 Disk Imager.**

Then from terminal in Raspberry type the following commands giving the time required for download, installation and configuration:

```
sudo apt update

sudo apt full-upgrade -y

sudo apt autoremove -y

sudo apt install raspberrypi-kernel-headers

sudo reboot
```

I have to admit that all this was not immediate, unfortunately you can find very few indications on the net and they are often misleading for your needs, hardware/software available on your computer as well as firewall and antivirus configuration.

Then an important thing was to check which IP address was to be pursued and configured so that everything could talk at its best and without bottlenecks.

For example, your own router can reserve some unpleasant surprises, in my case with the R2 and the 10M sample rate, the transferred audio is hiccuping and ripped (unusable) and I could not understand if it depends on the RAM of my Pi or on other configuration parameters of the Spyserver.config file (for example I tried to change the parameter "Force-8bit = 1") or even from the supplied power supply with a few amperes (symbol A)...

But this is all part of that amateur radio spirit that leads one to experiment with patience and renewed enthusiasm with even the most complex and unfamiliar things.

Tests carried out on a Pi3 have shown that it is possible to run two RTL-SDRs at the same time, provided that the performance is not too exaggerated...
In fact, it is possible to obtain reasonable results by using, for example, AIS and ADS-B decoders in parallel, which do not require the entire stream to be transferred, but only the processed stream...

Main features and differences between the various models Pi…

	Raspberry Pi 5	**Raspberry Pi 4**	**Raspberry Pi 3 B+**
			
RAM	4/8 GB (LPDDR4x-4267 SDRAM)	2/4/8 GB (LPDDR4 - 2400 SDRAM)	1 GB (LPDDR2 SDRAM)
Processor	Broadcom BCM2712 Quad core Cortex-A76 (ARM v8) 64-bit @ 2.4 GHz	Broadcom BCM2711 Quad core Cortex-A72 (ARM v8) 64-bit @ 1.8 GHz	Broadcom BCM2837B0 Quad core Cortex-A53 @ 1.4 GHz
GPU	VideoCore VII @ 800 MHz	VideoCore VI @ 500 MHz	VideoCore IV @ 250-400 MHz
Power connector	5V 4A via USB-C	5V 3A via USB-C	5V MicroUSB
USB 3.0	2 @ 5 GBps	2	-
USB 2.0	2	2	4
Display connector	2x 4kp60 mini HDMI	2x 4kp60 microHDMI	1x HDMI
WLAN Wi-Fi	802.11ac	802.11ac	802.11n
Ethernet	Gigabit	Gigabit / 1000 Mbps	300 Mbps
Bluetooth	5.0	5.0	4.1
Dimension	86 x 56 x 21 mm		

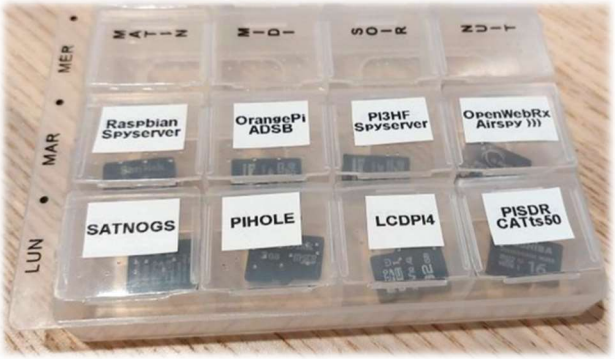
I got to try out a Pi4 with 8GB of RAM which definitely offers more modern equipment, better technology, and even more usage options due to the larger and faster RAM.

These are basically the main differences with the Pros and Cons.

	Raspberry Pi 5 (launched in late 2023)	Raspberry Pi 4	Raspberry Pi 3 B+
Pros	Two to three times more powerful and faster than Pi4; External PCIe v2.0 connection for fast peripherals	Better CPU and memory than Pi 3; Dual 4k monitor support; USB 3.0; 3.5mm audio jack	Single full-size HDMI port; Lower power consumption and board overheating
Cons	No 3.5mm audio jack; Perhaps a bit excessive for normal DIY projects	Higher power consumption and overheating; Absence of full-size HDMI ports; No PCIe connection; New case; Higher cost than Pi 3	Less-than-bright specifications; USB 2.0; No PCIe connection.

In our area of interest, with the same software installed and hardware connected (wireless router and Airspy R2 device) the Pi4 proved to be really strong and finally I was able to manage the SpyServer at its maximum potential at 10 MSPS IQ *(with the previous Pi3 B+ beyond 2.5 MSPS IQ the audio reached the client all fragmented and hiccupping)*.

A few days ago I came across an all too nice Twitter from my colleague Oscar EA3IBC who, in order to properly collect and classify the various micro SDcards in his Raspberry, had a brilliant idea! He gave me permission to share it with you all....



The use of a simple pill storage container...

Or specific cases (including waterproof ones) can be found on the market simply by typing "microSD memory card holder" into Google.

Linux & Airspy

If we wanted to use one of the very many native software for the Linux operating system that do not then have a version in our beloved Windows we have to arm ourselves with a little time and a lot of patience but the results are assured!

There are two possibilities:

- Install a Linux distribution directly on your computer's HD or on its dedicated partition
- use a Virtual Machine in which you will then go on to install the ISO of one of the many Linuxes available. As a reminder in this regard, there is a distro that promises interesting things: DragonOS is an out-of-the-box operating system based on Lubuntu x86_64 for anyone interested in software-defined radios. All installed source software is in the /usr/src directory, with the remaining software installed by package managers.

  Leverages the portability, security and power of Lubuntu Linux as a distribution package and operating environment for a preinstalled suite of the most powerful and affordable open source SDR software. It also supports a variety of SDR devices.

  This is the reference link: https://sourceforge.net/projects/dragonos-focal/

  And a helpful video: https://www.youtube.com/watch?app=desktop&v=lTBtlGGf5KE

So, giving one of the previous two points as assured, let's see, for example, how to install the Airspy package in Ubuntu by typing the following commands directly into the terminal:

```
sudo apt-get update
sudo apt-get install Airspy
```

Should it be necessary to proceed with the uninstallation of the Airspy package, this is the command:

```
sudo apt-get remove Airspy
```

To uninstall it along with its dependencies:

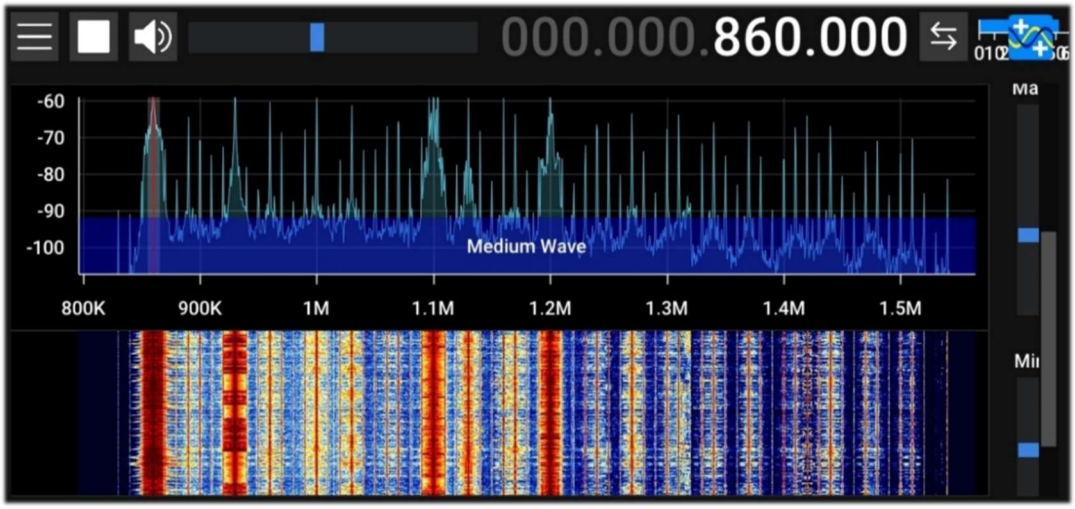
```
sudo apt-get autoremove Airspy
```

If you also want to delete the configuration and data files:

```
sudo apt-get purge Airspy
```

…and if we think about Android?

Now that we have properly configured our Raspberry with the Spy Server we can also try using Alexandre Rouma's open source cross-platform software SDR++ with our Android devices (version 9.0 and up). It can be found here: https://github.com/AlexandreRouma/SDRPlusPlus

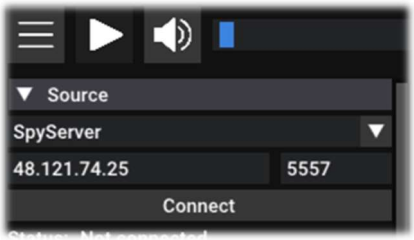


Let's see the whole thing in a few steps by starting with downloading and installing the SDRPP.APK file on our smartphone/tablet from this link since it is not in the "Play Store" market:

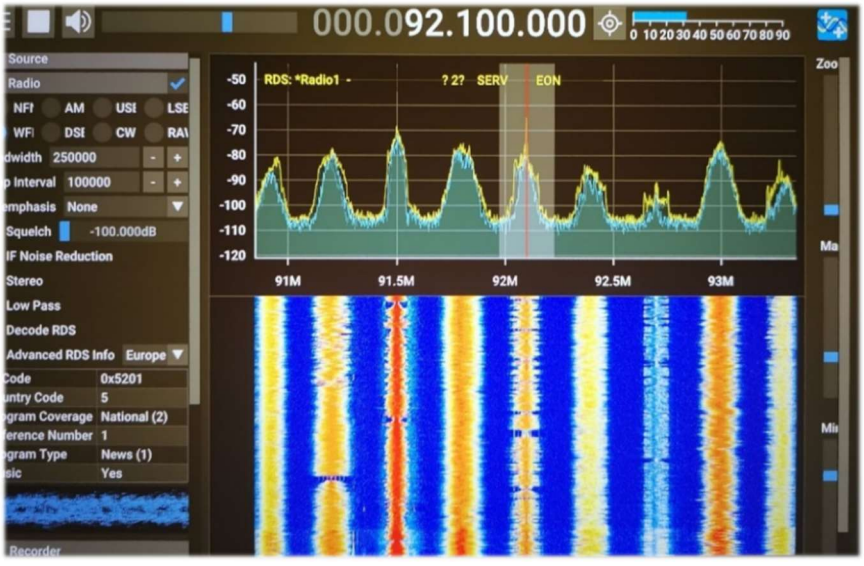
https://github.com/AlexandreRouma/SDRPlusPlus/releases/tag/nightly

First you will need to set the correct resolution ("DPI scaling" in the Display menu) otherwise you will not be able to use it. In my case, with a Galaxy Tab A8 tablet, I set the DPI scaling to 200%.

Then from "SOURCE" you select the last item "SpyServer" by entering the IP and port and then clicking "Connect" to establish the connection...



If everything is configured correctly it will boot up with the "Play" button and you can configure it to the best of your ability: in this screenshot it is at work in band 88-108 in FMW with the related RDS plugin



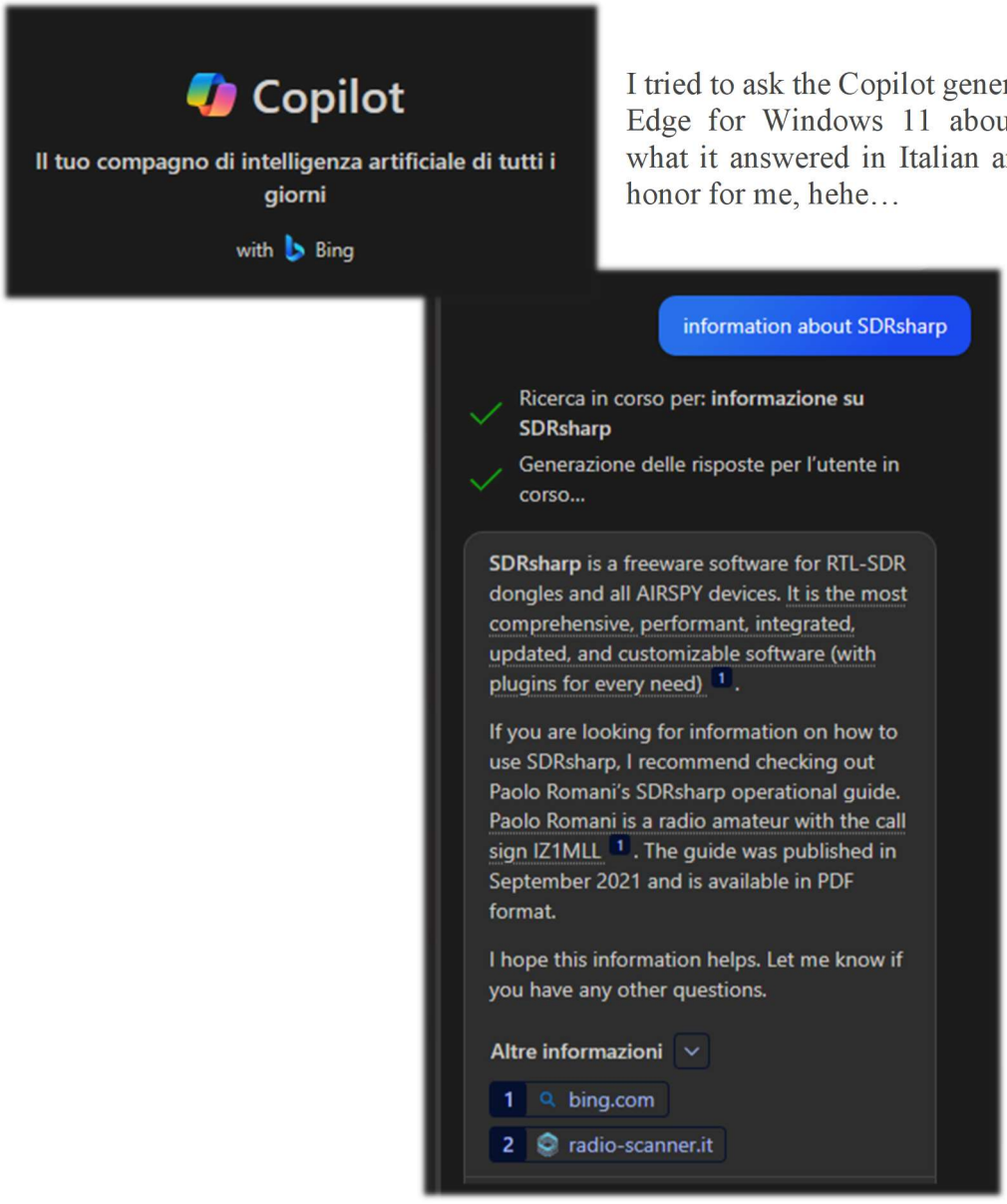
What they say about us…

Our beloved SDR# is well known, used and exploited in videos and some documentaries, etc where it appears prominently and immediately recognizable...

I have started collecting the following and if you have others to suggest they will be welcome!

---------- Artificial Intelligence (generative) ----------

AI (Artificial Intelligence) is very fashionable these days.

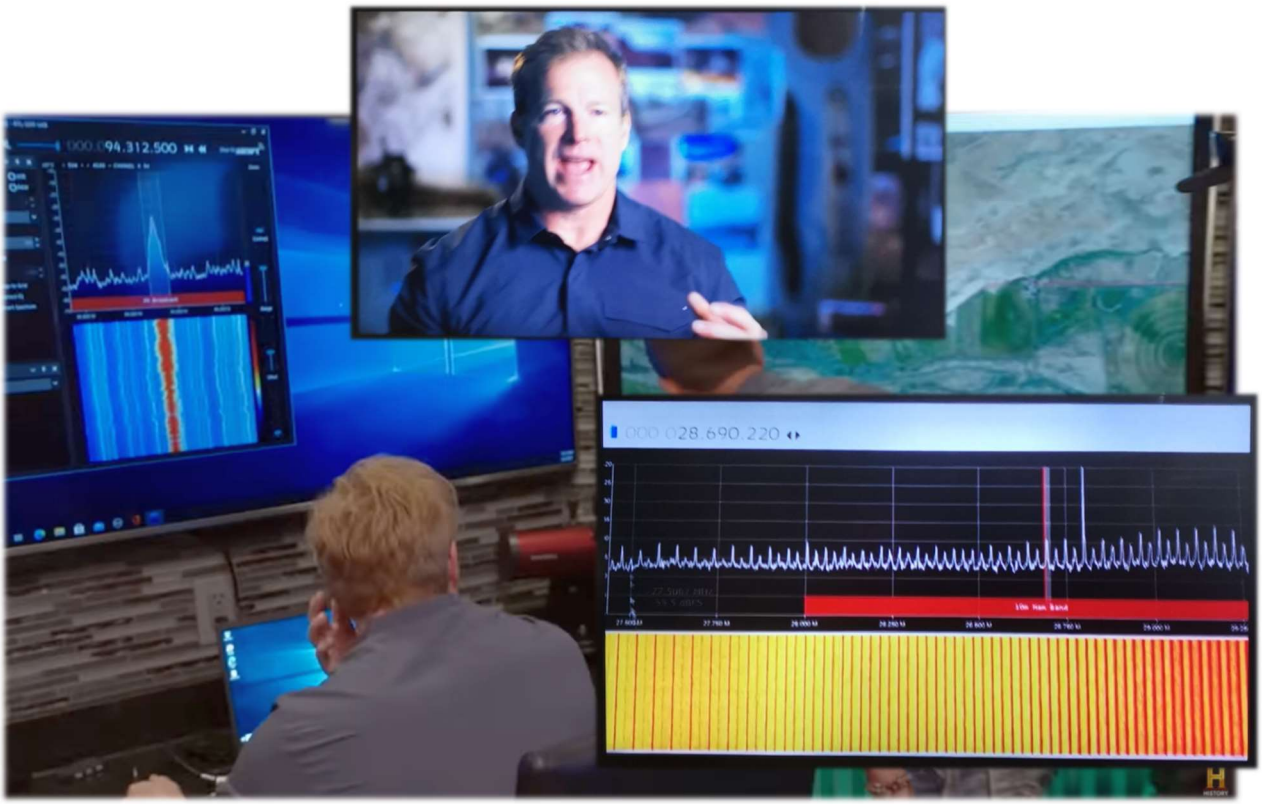


I tried to ask the Copilot generative AI in Microsoft Edge for Windows 11 about SDRsharp... here's what it answered in Italian and English, really an honor for me, hehe…

---------- VIDEO ----------

The Secret of Skinwalker Ranch: (Season 1/3)

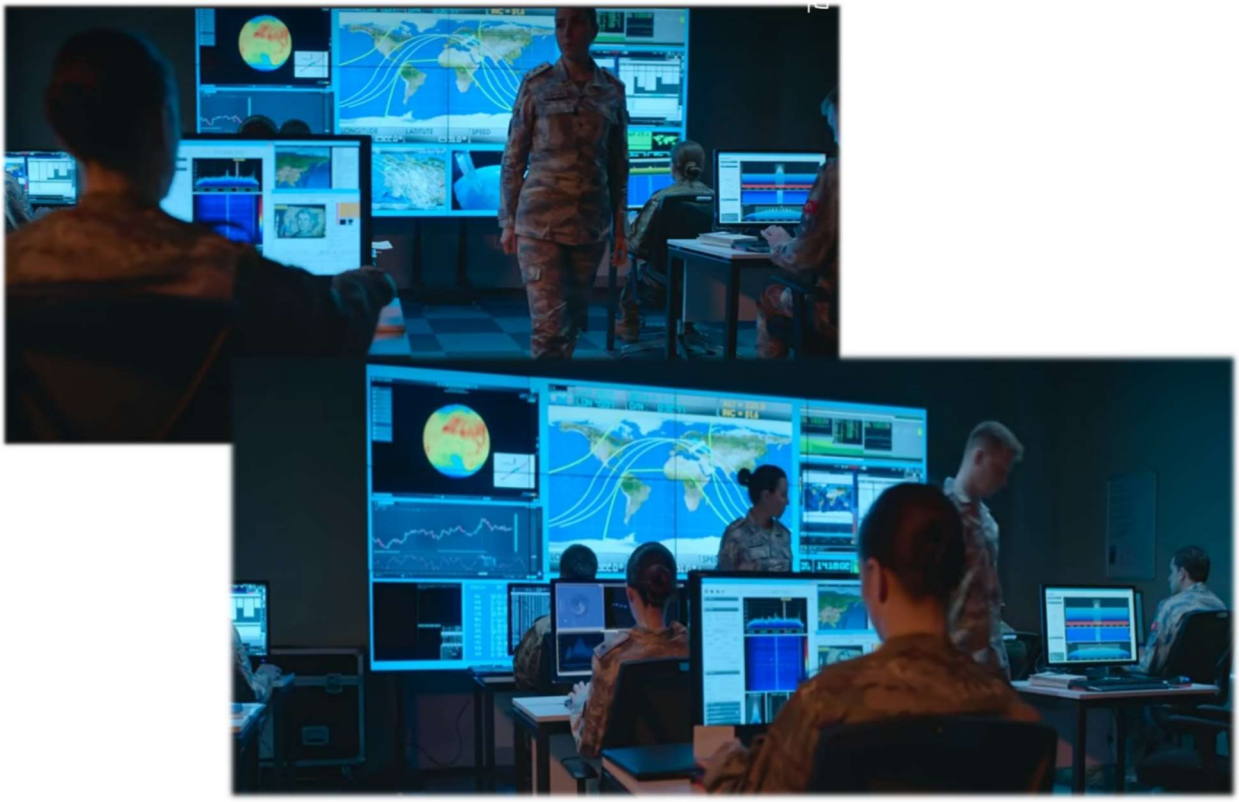
https://www.youtube.com/watch?v=OBzHOJGxhJg



Or these other screenshots taken from...

Netflix TV series Yakamoz S-245

https://twitter.com/SV2HWM/status/1517879132864106497



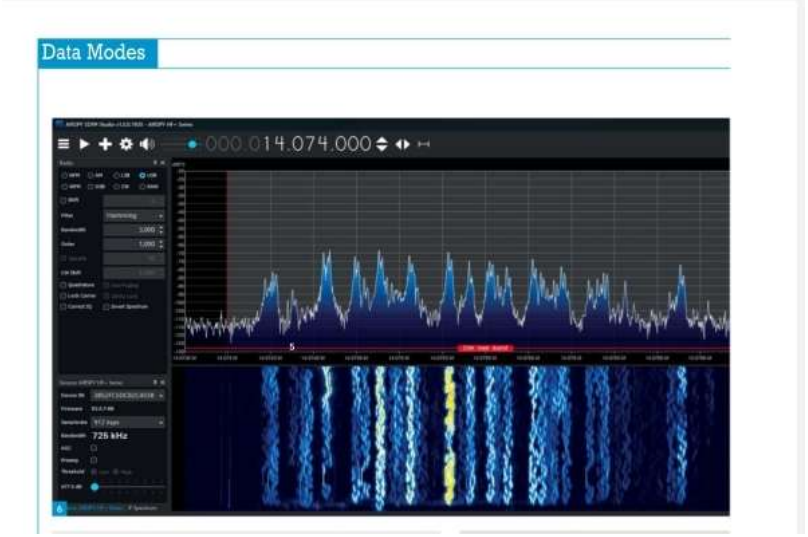
Screen taken from the blog of **"Kyushu Institute of Technology Satellite Development Project"**

Practical Wireless – june 2024:

https://www.radioenthusiast.co.uk/

SDRSharpr1920

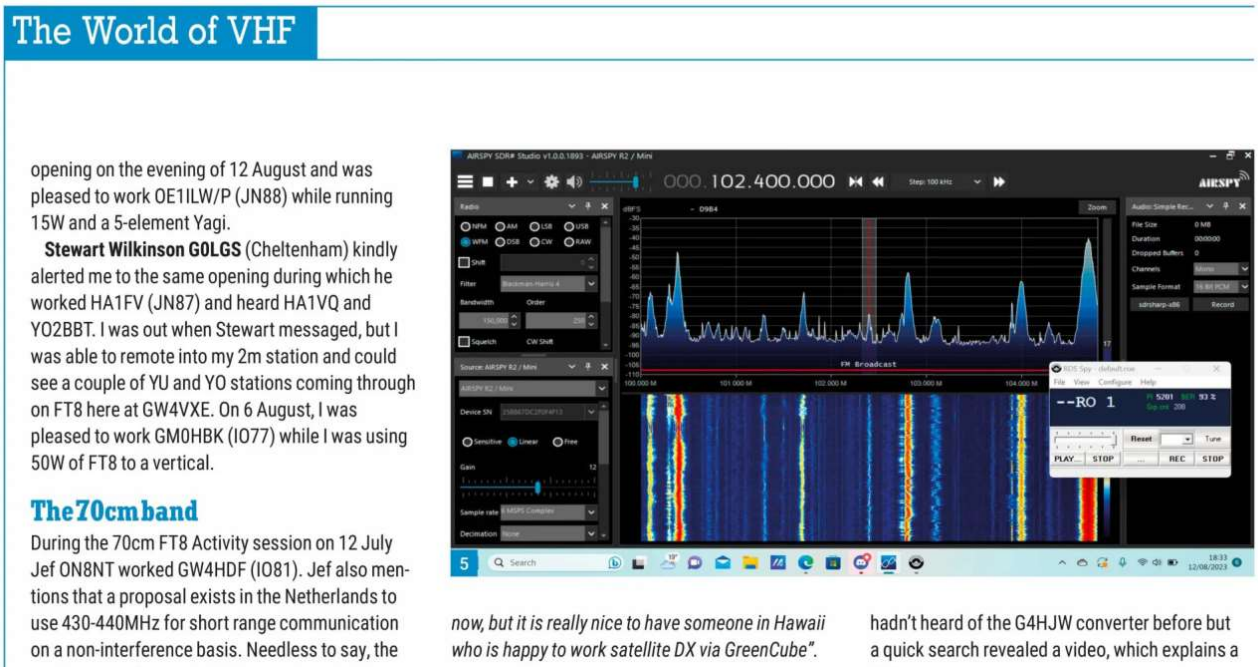
SDR Sharp continues to be one of the most popular SDR applications and is packed with useful and powerful features. The author, **Youssef Touil**, recently rewrote the User Interface (UI) to use his bespoke code instead of the previous Telerik UI software. By cutting out the overhead of the Telerik components, SDR Sharp has benefitted from a significant performance boost. As an added benefit, some of the user-developed plugins that had stopped working are now working again. At the time of writing, release 1920 is only available via the Airspy group on Groups.io. However, I'm sure it will be on general release very soon. In **Fig. 6**, I've shown a screenshot of SDR Sharp r1920, providing a detailed view of FT8 activity on 20m.



Data Modes

Practical Wireless – October 2023: https://www.radioenthusiast.co.uk/

Page 16 FMW/RDS reception tests in E-sporadic and comparison with software "RDS Spy"



The World of VHF

opening on the evening of 12 August and was pleased to work OE1ILW/P (JN88) while running 15W and a 5-element Yagi.

Stewart Wilkinson G0LGS (Cheltenham) kindly alerted me to the same opening during which he worked HA1FV (JN87) and heard HA1VQ and YO2BBT. I was out when Stewart messaged, but I was able to remote into my 2m station and could see a couple of YU and YO stations coming through on FT8 here at GW4VXE. On 6 August, I was pleased to work GM0HBK (IO77) while I was using 50W of FT8 to a vertical.

The 70cm band

During the 70cm FT8 Activity session on 12 July Jef ON8NT worked GW4HDF (IO81). Jef also mentions that a proposal exists in the Netherlands to use 430-440MHz for short range communication on a non-interference basis. Needless to say, the

now, but it is really nice to have someone in Hawaii who is happy to work satellite DX via GreenCube".

hadn't heard of the G4HJW converter before but a quick search revealed a video, which explains a

Practical Wireless – July 2023: https://www.radioenthusiast.co.uk/

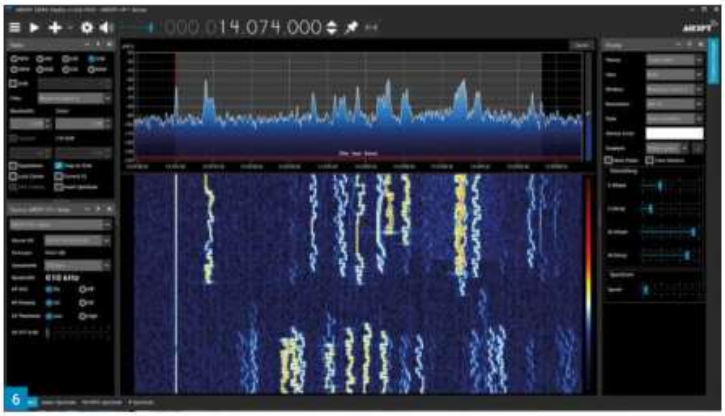
Data Modes

the display's appearance. However, you should beware of straying too far from the default settings, or you could be suppressing some signals.

The window type is not particularly critical for general listening as you will notice little practical difference between the popular Hamming and Blackman variants. The only time window choice becomes important is when measuring closely spaced signals on a quiet band. In this case, it's worth experimenting with different windows to minimise any smearing between FFT bins.

Waterfall Displays

The standard spectrum display is excellent for live signals, but the waterfall display provides a different view of the same data. The key difference is that the waterfall shows how the



The Spectrum Monitor – august 2022: https://www.thespectrummonitor.com/august2022.aspx

FEDERAL WAVELENGTHS

By Chris Parris cparris@thefedfiles.com

SDR Use in Federal Monitoring

One of the hottest topics in the radio hobby these days is the Software Defined Radio, or SDR. Over the last ten years or so, the availability of inexpensive receivers that can be controlled by using a computer has led to numerous software packages and new SDR hardware that continues to fuel innovations in communications monitoring.

Let's discuss what a software defined radio is. A software-defined radio (SDR) is a radio communications device where the normal electronic components such as mixers, filters, amplifiers have been replaced with a chip-based, wide-band radio frequency amplifier, receiver, tuner and digital processing. All these components are then controlled by software on a personal computer. SDRs can be just receivers or transmitter/receiver combinations. This design allows for a variety of different transmitting and receiving protocols based on the software used to control the SDR.

SDR technology has been used for some time in military, cellular and commercial radio systems. Only in the last ten years or so has the previously expensive SDR technology began to become readily available to the amateur and home experimenter. The newer Uniden SDS line of scanners is built around an SDR receiver, so expect to see more of this receiver technology in future scanners and receivers.

The first hobbyist SDR devices that I saw were sold as digital TV tuner "sticks" that plugged into your computers USB port. They were controlled with software on your PC that tuned the SDR receiver to local over-the-air television signals. These tuners could be controlled by the software to tune in FM broadcast stations and analog or digital TV transmissions in any of the worlds TV channels or formats.

Experimenters quickly realized that these USB stick tuners were able to be controlled by custom written software to tune in nearly any frequency the user desired. Additional computer software, such as DSD Plus, allows you to decode various digital communications modes, such as APCO P-25, DMR, NXDN, etc. at very low cost. So, the true performance of the SDR unit comes not only from the receiver hardware, but from the computer and software that is controlling that hardware, since the software is controlling and defining the modes and frequencies that the hardware is operating on.

In the earliest days of experimental digital SDR devices, there was an element of tinkering one had to be prepared for. Software was mostly command line based, lots of variables had to be set to get the USB stick to talk to the

The Airspy R2 connected to a small tablet computer. (Courtesy of the author)

software, and the best results required some experimentation by the user. Nowadays, most SDR devices are pretty much plug-and-play. You can download free versions of the basic operating programs for the popular SDR tuners, connect your device to an antenna and a PC or tablet and go from there. The operating software for popular SDR devices has been written for various computer platforms, including Windows, Mac, Linux and even Android and iOS devices.

While there are many different brands and types of SDR devices available these days, my experience has been primarily with the Airspy R2 and SDR Play units. Both are reasonably priced at under the $200 mark and both offer easy setup and operations with little preparations or computer knowledge. For the purposes of this month's column, I will concentrate on the Airspy unit and my initial experiences with it. Full disclosure – Airspy US is a TSM advertiser, but I received no compensation for any mentions in this column.

Unboxing the Airspy, you will see that you only need to make two connections to get started – a USB connection to the computer device and an antenna. The device gets its power through the USB connection, so no external power is needed. The audio from the receiver is processed through the computer that controls the SDR, so you will hear your communications through your PC speakers. Using the basic setup, you can receive many different analog modes from 24 MHz to 1800 MHz. Using an external converter (or using

The SWLing Post – october 2022:

https://swling.com/blog/2022/10/mario-shares-a-short-review-of-the-Airspy-hf-discovery/

Mario shares a short review of the Airspy HF+ Discovery SDR

5 Replies

Many thanks to *SWLing Post* contributor, Mario Filippi (N2HUN), who shares the following guest post:

Author's Airspy HF+ Discovery (small black box to the left of the laptop)

A Short Review of the Airspy HF+ Discovery SDR

by Mario Filippi (N2HUN)

I recently purchased an AirSpy HF+ Discovery. As a SWL for over 60 years who's owned many shortwave radios by manufacturers such as Drake, Yaesu, Icom, Zenith, Kenwood, Panasonic, Sony, Radio Shack, Grundig, CountyComm, MFJ, Sears, AOR and have used a number of different SDRs such as the RTL-SDR.com, HackRF, NooElec and many other rudimentary inexpensive first generation SDR dongles, I feel the AirSpy was an excellent choice. It cost $169 plus

FUNKAmateur – february 2018:

https://www.funkamateur.de/

RadioUser – March 2015

https://www.radioenthusiast.co.uk/

Other important reviews we can read here:

https://Airspy.com/reviews/

My book/guide is also starting to be cited, here are some examples...

---------- MAGAZINES ----------

monthly magazine

"RADIOKIT ELECTRONICS"

April 2021

n.4 2021 Aprile

€ 5,50

radiokit elettronica

TECNICA E COSTRUZIONI - RADIANTISMO - STRUMENTAZIONE - HOBBY

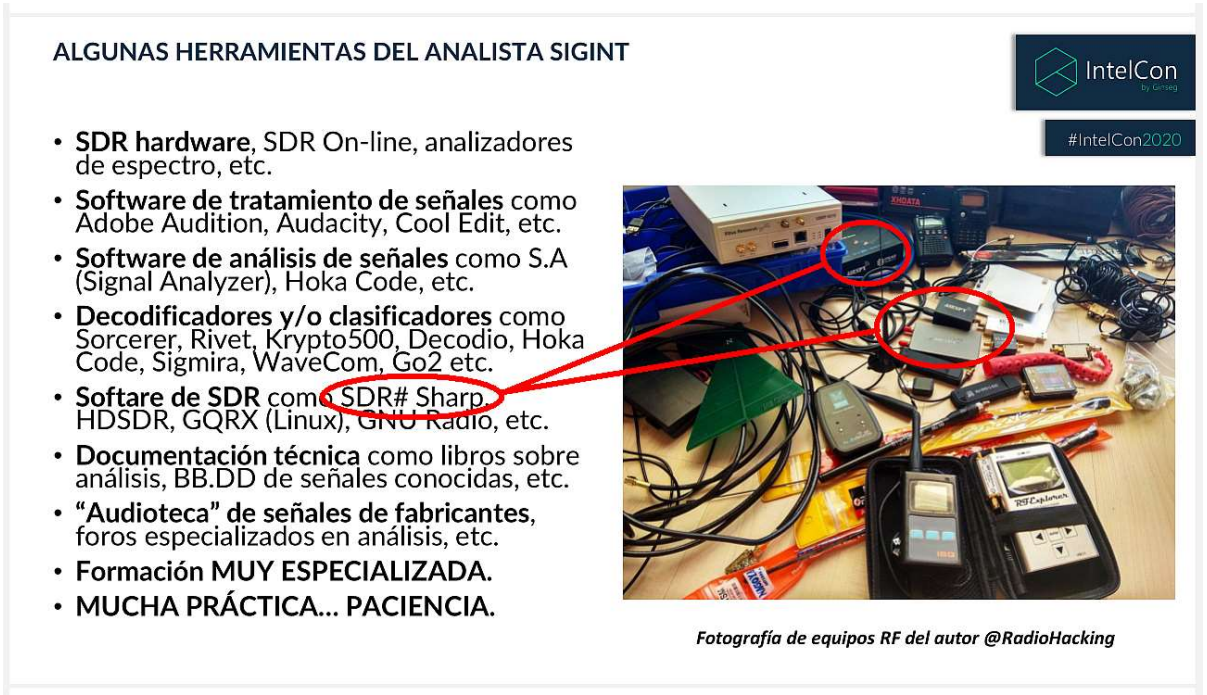
SDR SHARP

"SDRsharp, per far vedere i colori a chi ascolta in bianco e nero..." è il titolo che Paolo Romani, IZ1MLL, ha scelto per la pubblicazione su AIRSPY della nuova guida operativa v2.1 aggiornata a febbraio 2021 del favoloso SDRSharp (o SDR#), il software freeware più completo, performante, integrato, aggiornato e personalizzabile (con plug-in per ogni necessità) per i dongle RTL-SDR e ovviamente tutti i device AIRSPY. La release 1785, rilasciata ufficialmente il 5 febbraio 2021 nell'ottica della continua ricerca di miglioramento e perfezionamento, ha fatto un grande salto verso il più recente .NET5 di Microsoft. Questa piattaforma di sviluppo multisistema, open source, è capace di supportare l'esecuzione side-by-side senza la necessità di dover installare il runtime. Una guida come questa non nasce per caso. Il contenuto delle oltre quaranta pagine è il frutto di anni di ascolto, dedizione, passione e moltissimo impegno personale alla ricerca delle migliori configurazioni e ottimizzazioni possibili. Il testo è ricco di suggerimenti operativi introvabili altrove. L'augurio di Paolo: "Buona lettura e buoni ascolti con il Software Defined Radio a tutti quelli che credono in esso, poichè quando accenderemo il nostro nuovo SDR saremo in grado di comprendere facilmente che questo mondo ha davvero tante facce ma un unico cuore" è quanto di più condivisibile possa esserci. Buona lettura a tutti. Maggiori informazioni su ***https://airspy.com***

Rke 4/2021 5

---------- DIGITAL PUBLICATIONS ----------

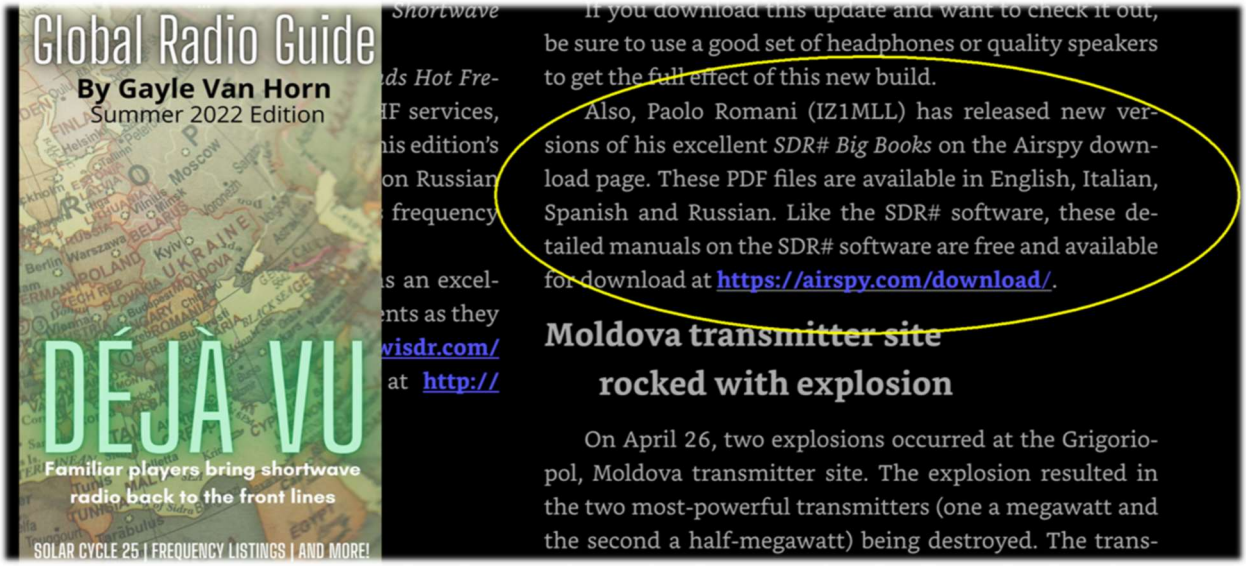
#IntelCon2020 – Matrioshka SIGINT: David Marugán @RadioHacking – Julio 2020



Fotografía de equipos RF del autor @RadioHacking

Gayle Van Horn's (W4GVH) 18th Edition of her bestselling Global Radio Guide (Summer 2022)

https://www.teakpublishing.com/books



---------- **WEB** ----------

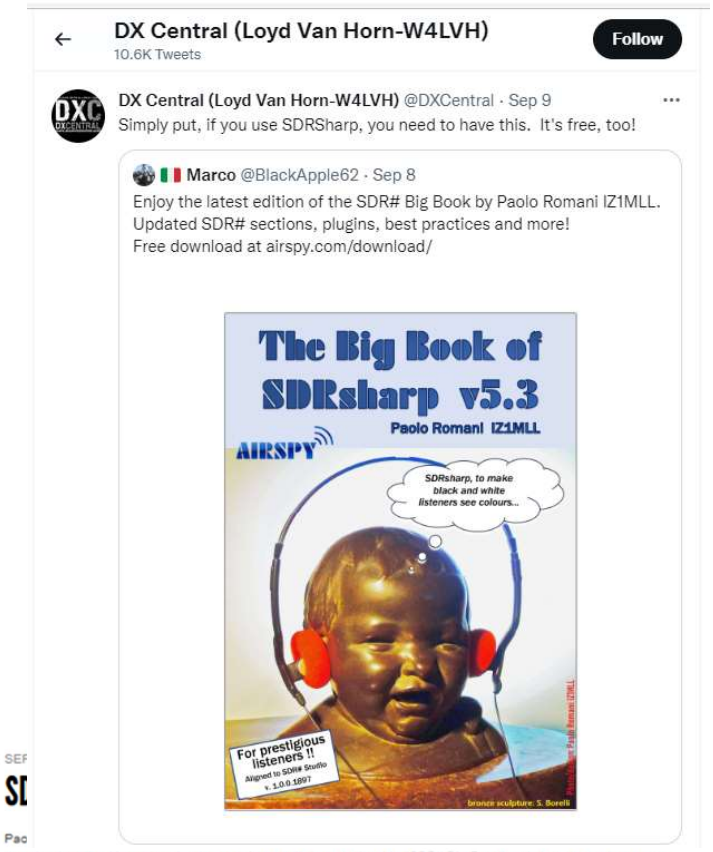
AIR RadioRama:

http://air-radiorama.blogspot.com/2023/01/ll-gran-libro-di-sdrsharp-di-paolo.html

Twitter:

https://twitter.com/BlackApple62

https://twitter.com/DXCentral



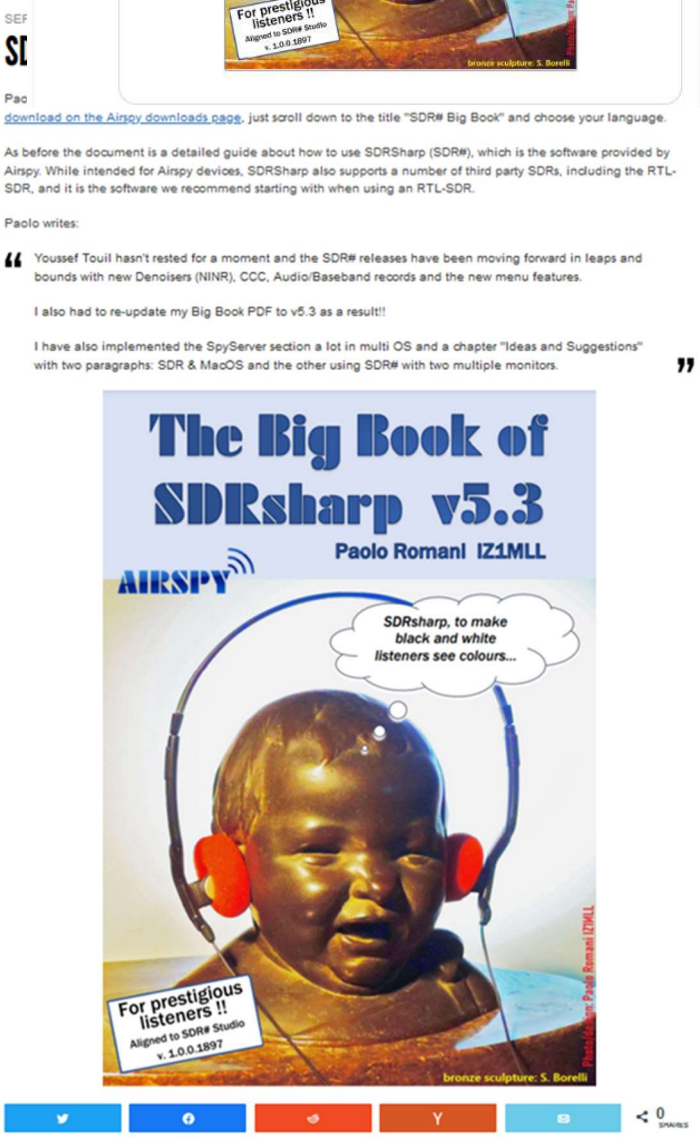
from the RTL-SDR website:

https://www.rtl-sdr.com/sdrsharp-big-guide-book-updated-to-v5-3/

https://www.rtl-sdr.com/sdrsharp-guide-v4-2-released/

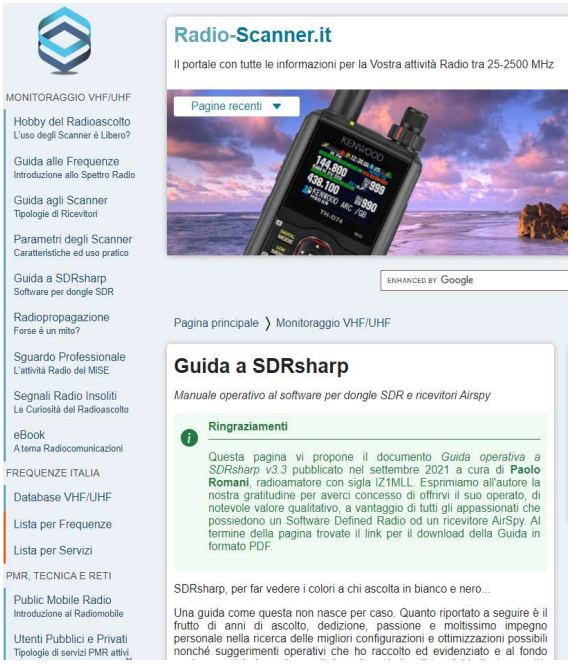
https://www.rtl-sdr.com/sdrsharp-guide-v3-0-released/

https://www.rtl-sdr.com/new-sdr-user-guide-available/



Twitter:

https://www.radio-scanner.it/guida-sdrsharp-radio.html



https://www.radiomasterlist.com/en/ebook.html

t17lab.com

BLOG NỘI DUNG VNSATCOM [BOOKMARK]

Phát hành Tài liệu SDRSharp phiên bản 4.2

19/05/2022 | SDR

Paolo Romani (IZ1MLL) gần đây đã phát hành phiên bản 4.2 của tài liệu PDF Hướng dẫn SDRSharp của mình. Cuốn sách có sẵn để tải xuống trên trang Airspy, chỉ cần cuộn xuống tiêu đề "SDR # Big Book in English".

Như trước tài liệu là hướng dẫn chi tiết cách sử dụng SDRSharp, là phần mềm được cung cấp bởi Airspy. Mặc dù dành cho các thiết bị Airspy, SDRSharp cũng hỗ trợ một số SDR của bên thứ ba, bao gồm RTL-SDR và đây là phần mềm chúng tôi khuyên bạn nên bắt đầu khi sử dụng RTL-SDR.

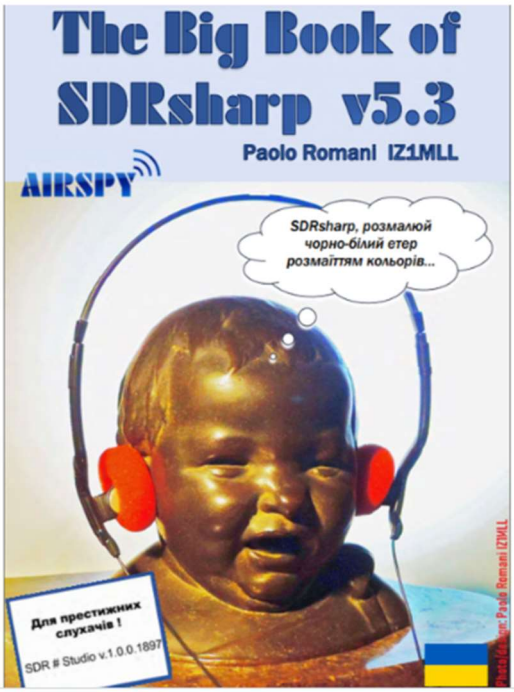
Paolo viết:
PDF v4.2 SDRsharp mới của tôi đã ra mắt. Hướng dẫn hiện dài 139 trang và bao gồm tất cả các cài đặt, tùy chỉnh giao diện người dùng, plugin bao gồm và bên thứ ba, cũng như sử dụng một số bộ giải mã và phần mềm bên ngoài, hiện được tích hợp Spyserver với Raspberry Pi 3/4, v.v.

Tải tài liệu hướng dẫn sử dụng SDR#
https://node1.t17lab.com/index.php/s/XuBl4bxQZUdwIPj

From the Static (Larry Van Horn N5FPW) @MilcomMP · 27 ott

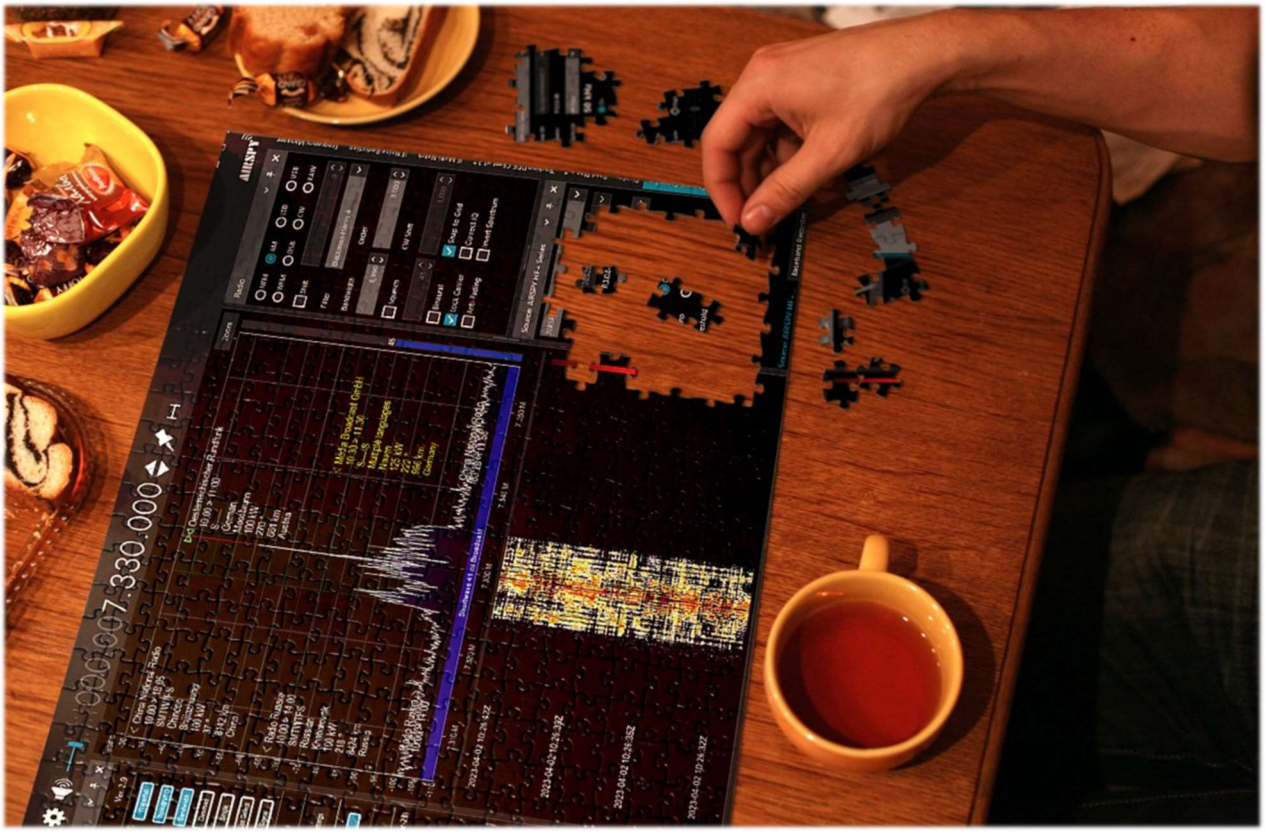
If you are looking for guidance on using SDR# look no further than the Airspy website download page and the SDR# Big Book. Available in English, Italian, Russian, Spanish, and now Ukrainian. Excellent reference and did I mention the price - Free!! Go get your copy right now.

> **prog** @lambdaprog · 27 ott
>
> The Ukrainian translation of Paolo's SDR# Big Book is now available for download: airspy.com/download/



2 10

---------- Curious images from https://twitter.com/lambdaprog ----------



from IZ1MLL

from Volodymyr

from Ja-radio

FAQ

A lot has changed with v1920 and the compatibility of plugins has been checked, some of which had problems. The new SDR# focuses on usability, correctness, high performance and low power consumption as the first step in building an embedded platform. Many DSP blocks have been updated with the latest algorithmic refinements, such as FFT display, filtering, noise reduction, channel cancellation, and Multi-Notch. Airspy's user mode drivers have also been improved to provide greater reliability and resistance to USB problems.
The current Community Installer has been updated with a list of plugins guaranteed to load correctly, while some others, which are no longer compatible, have been removed:
IF Processor (completely rewritten in the current Multi-Notch)
NetRemote (never worked properly and causes UI crash due to improper use of sockets)
Audio UDP Streamer (same multi-threading problems as NetRemote, as well as referencing DLLs that no longer exist).
Perhaps there are other plugins that need some fixes, so developer-authors are encouraged to check their own work and if necessary modify the necessary. It is also recommended to start with a new configuration file of SDR#.

Q: *"My IQ recordings don't fit ...where am I going wrong?"*

A: *I invite you to read the initial chapter "Baseband Recorder".*

Q: *"With Airspy R2 and the plugin "Audio Recorder" my recording works well, but the playback of the recorded audio is very low volume compared to the direct sound from the speakers. Audio mixer, Windows settings and configuration and everything is set to 100%, but the playback is barely noticeable".*

A*: Try using the freeware software "Audacity." Select the track and then click on Effects / Volume and Compression / Normalize / File / Export.*

Q: *"I have always detected strange things on my laptop with Airspy. On some afternoons/evenings, a clump of signals forms at a specific point in the spectrum. It is not a specific frequency, and by changing the VFO the clump always stays there: they are AM signals, broadcasting that probably saturate some stage...".*

A: *Evidently there is too much signal in HF that "pierces" the tuner and they enter IF (and in fact are fixed always at those precise points). A modest high-pass filter should remove them.*

Q*: "My Airspy (R2 but also the Discovery) have many internal products. In the 20-meter band I have leaks right on 14.230 kHz with the Discovery. Let's not talk about the R2 in the air bands where I have harmonic lines, aliases that appear as I vary the tuning...."*

A: *One has to take that a 24-1700 MHz tuner, without filters, already does things at the limits of what is possible... Also, if you see aliasing (which usually moves in the opposite direction of tuning) it means that there are stages in saturation. Too much input signal or setting gains too high. It certainly*

helps a lot to have a notch suppressor filter for the FM band and especially not to use active antennas in city environments or near radio links. The Discovery suffers less precisely because it has appropriate band/high/low passes at the top of the whole receiver.

***Q:** "The new releases v1911, 1912 and 1913 crash immediately as soon as I change the frequency or move the Spectrum. I use an RTL-SDR Blog V3."*

***A**: It is recommended to replace the rtlsdr.dll file with the new one taken from:* github.com/rtlsdrblog/rtl-sdr-blog/releases/latest/download/release.zip

***Q:** "Where are the DX stations? Can the SDR help us?"*

***A**: Use common sense. Check if there is anything that can be heard by taking advantage of the Micro Tuner. A carrier a few millihertz away is a good signal. If there is nothing or if the carrier is too weak, all you can hear after suppression is a "reconstructed background" noise. Exactly the same sound when trying to recover threshold signals around the noise floor with inappropriate NR settings. As with any instrument there is a learning curve and one's skills can make a big difference. Perhaps the next version of CCC will include an "Auto Tune" option for those who have not seen the Micro Tuner episode...*

***Q**: "Can the GPU graphics card create noise and interference?"*

***A**: Yes, this type of card can emit a significant amount of RFI. As a remedy try using ferrites but if it is possible it is better to use a computer without a GPU card!!!*

***Q**: "...with my Airspy Mini I always have bad audio (ripped and croaking) in both AM and FMN even in file recordings"*

***A**: Possible that this depends on the hardware used with little RAM and/or inadequate processor. In the configuration panel under "Sample rate", lower the sampling from 6 to 3 MSPS.*

***Q:** "At the audio level, what are the best drivers?"*

***A:** The following is a link to an article describing the differences between the various Windows audio options. It assumes that MME is an older driver, dating back to Windows 3.1 while WDM and WASAPI are newer technologies and "should" provide better performance than MME:*

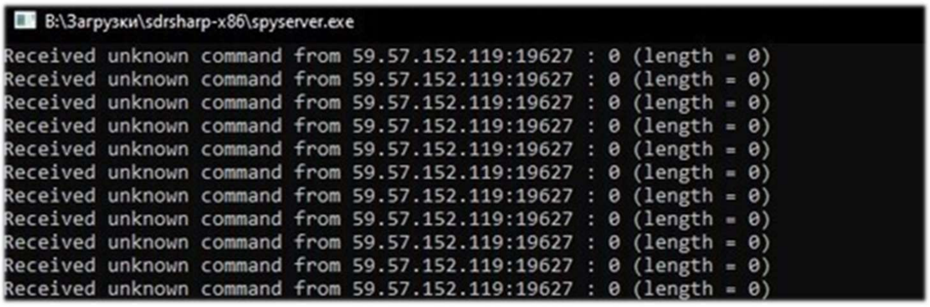
https://www.sweetwater.com/sweetcare/articles/roland-difference-between-asio-wdm-mme-drivers/

***Q:** "I have an RTL-SDR dongle connected to a USB port on my Raspberry RPi 4. When I run **./spyserver** I get this error":*

```
SPY Server v2.0.1700
Copyright (C) 2016-2018 Youssef Touil - https://Airspy.com
Reading the configuration file: spyserver.config
Listening for connections on 192.168.1.103:5555
usb_claim_interface error -6
```

A: *Shut down the other program that has it open.*

Q: *"Has anyone else encountered strange activity on Spy Server like these?"*



```
B:\Загрузки\sdrsharp-x86\spyserver.exe
Received unknown command from 59.57.152.119:19627 : 0 (length = 0)
Received unknown command from 59.57.152.119:19627 : 0 (length = 0)
Received unknown command from 59.57.152.119:19627 : 0 (length = 0)
Received unknown command from 59.57.152.119:19627 : 0 (length = 0)
Received unknown command from 59.57.152.119:19627 : 0 (length = 0)
Received unknown command from 59.57.152.119:19627 : 0 (length = 0)
Received unknown command from 59.57.152.119:19627 : 0 (length = 0)
Received unknown command from 59.57.152.119:19627 : 0 (length = 0)
Received unknown command from 59.57.152.119:19627 : 0 (length = 0)
Received unknown command from 59.57.152.119:19627 : 0 (length = 0)
Received unknown command from 59.57.152.119:19627 : 0 (length = 0)
```

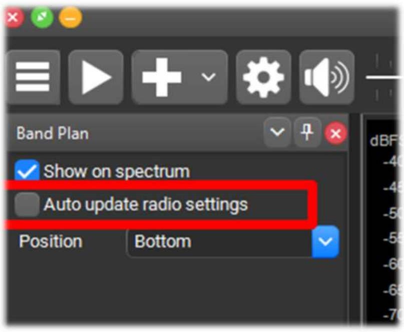
A: *Definitely an IP scanner in action... Suggestion: block IP inbound via your Firewall. Having a server on the internet makes it a target. It happens all the time and it is possible to see someone pinging, sniffing or whatever... If you don't constantly monitor youriInternet connection, and most don't, you won't notice anything, at least until you get hacked or suffer some kind of attack. Most gateways provided by the service provider offer a modest firewall and minimal monitoring and filtering capability. It would therefore be better to put a gateway in bridge mode and purchase a serious software firewall, especially if you are running any kind of servers, spyservers, FTP, remote desktops, or anything else. A good hardware firewall is also a good choice.*

Q: *"I recently purchased an Airspy Mini, but the displayed frequency (example: 429.8386 MHz) differs slightly from the real frequency (at 429.8375 MHz) with a difference of 1.1 kHz.*
How can I correct this problem? I saw that the calibration tool is no longer included and I tried the old version but it no longer starts."

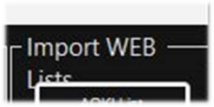
A *You have to enable debug mode in the SDRSharp.config file at line <add key="Airspy.debug" value="1" /> and then adjust the PPM calibration.*

Q: *"My Airspy SDR# does not stay in WFM (or FM either) when receiving NOAA sats. It jumps to USB mode... How can I solve this?"*

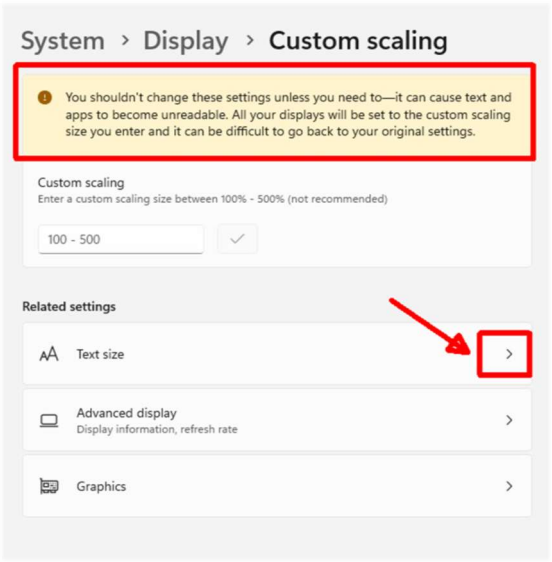
A: *Unflag the item "Auto update radio settings" from the default "Band Plan" plugin in SDR# menu, which associate frequency with a specific emission mode....*

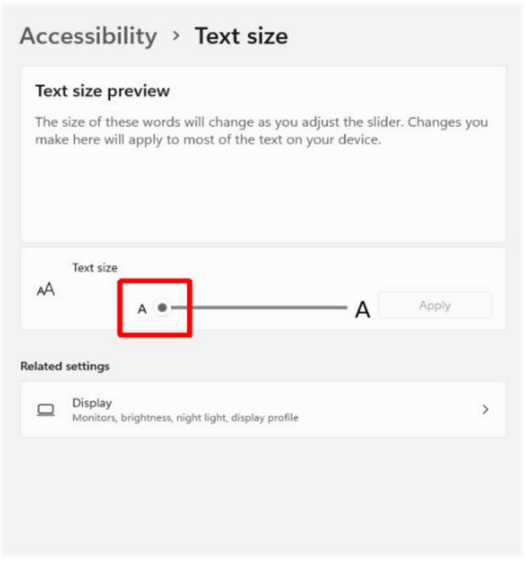


***Q**: "In some plugins appear misaligned graphics and fonts, like the following, whereas at first everything worked best..."*



***A**: The video resolution has been changed. To restore the original one, right-click on the Desktop: System / Display / Scale & layout / Scale 100% (recommended).*
Same thing if you change the size of the text…



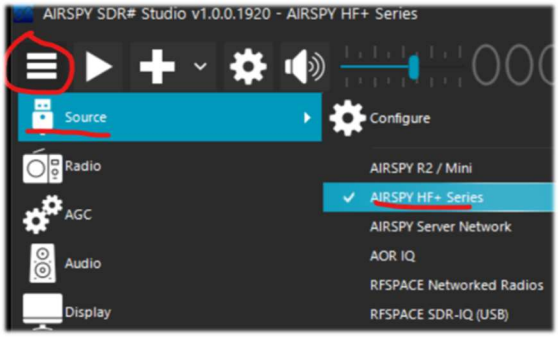


***Q**: "In some versions of SDR# after v1900 once the software is closed there remains an active process of the program visible as "SDR# Studio (32 bit)" by activating the key combination "ctrl+shift+ESC".*

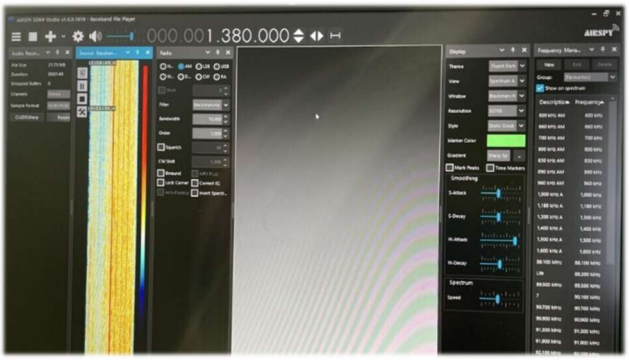
***A**: Verify that the file "SDRSharp.DigitalIfProcessor.dll" is not present in the plugins folder. This used to make IF Processor available but from v1900 onwards this service is rendered internally. Therefore, it is recommended not to install it anymore or to delete if it is present.*

***Q**: "I downloaded and installed the latest test package v1920, but after launching the software it did not recognize my Airspy HF+ Discovery. I suspect I am missing something very basic, hence my embarrassment....*

***A**: The choice of your device has changed in the latest versions of SDR# and you may have defaulted to another device. Verify here that you have selected the correct one.*

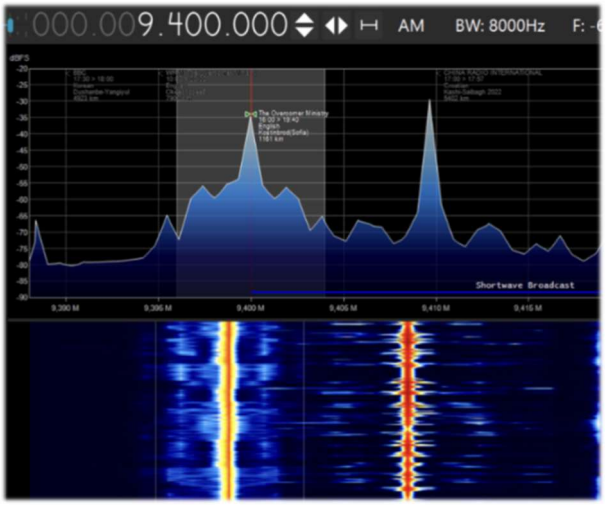


Q: *"The spectrum analyzer seems to be in draft mode or something like that. This happened after the cat jumped on the keyboard.... Is there a simple solution to this problem?"*



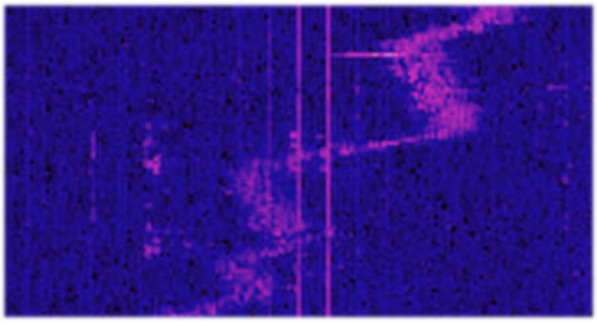
A: *Overwrite the layout file from the zip.*

Q: *"I noticed that in v1920 with SpyServer the signals on the RF window and the waterfall are not aligned. If I click on the center of the signal on the waterfall or the RF the frequency shown is the same, but the peak signal and the waterfall signal are shifted. The phenomenon is always reproducible and does not require any special steps"-*



A: *The detected artifacts can be traced back to several factors, such as reduced FFT and other compressions e.g. example, to reduce network utilization by the SpyServer. The use of decimation (Bandwidth drop-down menu) to magnify signals using the SpyServers is a better use of resources than using the slider Zoom. Or use the Full I/Q option whenever possible to avoid what is observed...*

D: *"I often get signals like these on 27/28 MHz. What could they be?"*



A: *They could be attributable to spurious emissions variables from semi-industrial heater/welder. Some are common and very well documented in some radio listening groups.*

SDRsharp history

Just to keep chronological memory of the "latest" software versions....

version	date	Change log
...		
1716	15sep19	**Last revision with No Skin**
...		
1761	04oct20	Added real sampling capability for single ADC radios. This brings significant CPU savings compared to the full bandwidth IQ conversion. To enable this feature for the R2/Mini the config key "Airspy.useRealSampling" must be set to "1". The baseband recording is not yet available for real sampling, but the IF should be still available for third party plugins.
1763	06oct20	Added full support for recording and playing Real spectrum files. Plugin authors are invited to contact me for more details.
1764	07oct20	Added Vasili's File Player and RTL R820T enhanced plugin.
1765	09oct20	Fixed the audio recording in the Wave plugin; Added more acceleration to the DSP.
1766	18oct20	Added AM DX Co-Channel Canceller plugin. Use in combination with the Zoom FFT filter.
1767	19oct20	Enabled the Boost SNR feature for all IF signals in the DNR plugin; Added marker colors for the Dark and Clear themes in the Co-Channel plugin; Many DSP code enhancements.
1768	19oct20	Improved the rejection in the Co-Channel Canceller; Added more controls: - Channel Bandwidth for the co-channel, IF Offset to shift the IF and filter out the interference.
1769	20oct20	Improved the the Co-Channel Canceller algorithm: Better tracking, Better phase noise, Better rejection
1770	24oct20	Many enhancements for the AM Co-Channel Canceller: Added more controls: Integration and Sensitivity, Better phase and amplitude tracking, Added some visual feedback in the spectrum Window to ease the tuning.
1771	28oct20	Added a new Co-Channel Canceller for FM. Same usage as the AM version; Allow wider bandwidth selection with dynamic decimation; Many DSP code enhancements.
1772	30oct20	Added a "Sensitivity" setting to the FM Co-Channel Canceller. This allows better fine tuning of the co-channel rejection. Many enhancements for the AM Co-Channel plugin. Added a new noise threshold algorithm that works with the dynamic decimation. The Wide FM mode is also supported.
1773	05nov20	Added Anti-Fading processing for the AM Co-Channel Canceller. Use this with Zero Offset. Changed the stepped increments to continuous for the different settings when applicable (NR, NB, CCC, Zoom, etc.)
1774	06nov20	Initialize the maximum VFO bandwidth for the SpyServer client from the config.
1775	06nov20	Polish: Enable the keyboard control of the Telerik sliders.
1776	07nov20	Added a status marker for the different DSP and plugin sections.
1777	10nov20	**With collassable panels**. Added a new high performance resampler for digital outputs.
1778	13nov20	New Visual Studio Interface with support to all the existing plugins.
1779	14nov20	Fixed the device initialization synchronization when the control panel is not active.
1780	14nov20	Added support for saving/loading the UI layout; The profiles can be saved/loaded live.
1781	16nov20	Smoother and faster handling of Airspy devices; Faster startup; The spectrum is now kept in shape when resizing; Same for the peak hold.
1782	17nov20	Added a stepped navigation bar.
1783	22nov20	Many audio and FFT latency optimizations; The sizes of the dock windows are now saved.
1784	23nov20	Smoother FFT streaming and lower memory usage.
1785	05feb21	**Now in Dotnet 5 Microsoft.**

1786	06feb21	Added new IMustLoadGui interface for forcing the plugin loader to bypass the lazy loading if needed. This is useful for plugins that need to be activated on startup. Examples updated in the Plugin SDK.
1787	06feb21	Added support for lazy GUI loading with active background processing.
1788	07feb21	Added a menu command to open all the setting panels available.
1789	10feb21	Faster loading of the "full plugin config", Better layout, Faster rendering, Fixed the auto-scroll theming.
1790	11feb21	Faster master loading; Faster slice loading; More slicing options; Many UI enhancements (rendering and performance). More layout enhancements; Added fall-back docking for older plugins. More layout and UI improvements.
1792	12feb21	Removed the panel borders for the plugins.
1793	13feb21	New adaptive FFT slicing/overlapping/skipping algorithm for the display; Improved refresh rate.
1795	15feb21	Optimized the adaptive FFT sequencing. Added sequence control and dynamic frame skip for the FFT display. Now the FFT display supports high sample rates at speed-and-resolution-constant resource usage.
1796	16feb21	Set the Garbage Collection to low latency mode; Added dynamic buffering depending on the data usage; A few minor UI enhancements.
1797	17feb21	Fixed many numerical rounding issues due to the way LLVM handles int64 and doubles; Code cleanup.
1798	17feb21	Set the step bar to fixed size. That was really annoying; Added new properties in the control interface: LockCarrier, AntiFading, VisualPeak, VisualFloor, ThemeName, Added extended logging to SNR Logger plugin, Clode cleanup. Getting ready for rev 1800.
1799	18feb21	More resampler optimizations. Significant gains in CPU usage.
1800	18feb21	Added two more properties in the control interface: ThemeForeColor, ThemeBackColor
1801	19feb21	Added automatic Plugin discovery and loading. Now you can just place the extra plugins in the "Plugins" directory and they will be loaded automatically. It is also use separate directories or some custom file tree. To disable the loading of a specific directory or a dll, rename it so it starts with an underscore "_". The plugin directory can be set in the config file so you can share it between multiple installations. You can use the config key "core.pluginsDirectory". Added automatic IF shift adjustment for the slices when using IF shifted sources. Many minor UI refinements.
1802	20feb21	Added more APIs: ThemePanelColor Property, RegisterKeyboardShortcut.
1803	20feb21	Added fail-save boundaries for range APIs.
1804	23feb21	Added more support code for RTL-SDR with the Community Package.
1805	24feb21	Updated Telerik library to version R1 2021 SP2; More consistent behavior of the PanelBackColor property.
1806	24feb21	Enhanced the initial control panel resizing mechanism for the plugins; Updated the build system for easier Telerik upgrades; Updated Microsoft.Windows.Compatibility" to version 5.0.2.
1807	26feb21	More loading speed optimizations; Fixed the initial position of the spectrum splitter with the main window maximized.
1808	02mar21	Removed the old .net Framework compatibility assemblies from linked executable. No impact on the API; More UI polishing: Main window size, startup location and startup size; More UI polishing: Plugin panels.
1809	04mar21	Replaced the web map with Telerik RadMap in the SpyServer source; Added full support for mouse wheel scrolling in the TrackBars (sliders); A few other UI enhancements. Initialize the bandwidth display for the HF+ source; Added binding redirects for better support of different .net assembly versions; Minor UI enhancements.
1810	06mar21	Added the necessary dependencies for Calico and many other plugins in the main package. These are not necessary for SDR# to run, but will ease the deployment of the plugins. Re-added the Windows Compatibility Package for the older plugins.

1811	29mar21	More DSP optimizations; Many fixes for RTL dongles (mainly workraounds for old libs); Revert to libusb 1.0.20.11004 for backward compatibility; Revert to portaudio 2016 for backward compatibility; Code cleanup. The ThemeForeColor property now reflects the color of a label within a plugin panel; Many performance optimizations for the Sharp Kernels library (shark.dll).
1812	03aug21	Added a new API for enumerating the loaded plugin instances. Added Linrad spectrum dot mode. Updated Telerik toolkit to version 2021.2.614.50; Added Gray and Dark Office 2019 Themes; Rewrote the spectrum rendering code to take advantage of more poweful CPUs and give smoother experience; Allow Airspy front-end controller to tune using sub-harmonic mixing (up to 4.29 GHz); Optimized the layout of the trackbars; More GC tweaks.
1813	16aug21	Switched to Server Garbage Collection for faster Telerik loading; Reordered RTL sources in the menu.
1814	17aug21	Updated the Table Layout controls and UI animations; Replaced the RadColorBox control with the OS default; Improved the layout loading.
1815	17aug21	Rearranged the loading of the plugins.
1816	18aug21	Fixed a regression in the dock visibility.
1817	18aug21	Improved the default waterfall gradient for better handling of HDR signals; Improved the resizing of the zoom/offset/range sliders.
1818	19aug21	Added low-latency "best effort" mode for audio playback; Added a gradient selector and a few built-in styles; Added more Airspy specific buffering; Many UI improvements. optimizations; Configured the audio latency dynamically; Improved the loading of docked plugins.
1819	20aug21	Improved the sensitivity of the FM Co-Channel Canceller; Improved the Co-Channel initialization code.
1820	21aug21	Added more sanity checks in the AM Co-Channel Canceller; Added theming fallback.
1822	21aug21	A few layout optimizations; More IQ buffering for slow sources; New theme loading mechanism with automatic theming for legacy plugins.
1823	01oct21	Upgraded to Telerik UI for WinForms R3 2021 (version 2021.3.914); Improved the UI loading.
1824	04oct21	Many GUI optimizations; Added progressive loading.
1825	05oct21	Many UI and GC optimizations; Added a status message in the splash screen.
1826	05oct21	Much faster UI loading.
1827	05oct21	Fix the Zoom FFT plugin initialization. Updated the quantization of the spyserver and moved its Windows tool chain to clang.
1828	06oct21	Fixed the initialization of the Airspy Network Browser.
1829	08oct21	Rounded corners around the status text in the Splash Screen - Windows 11 Style.
1830	08oct21	Moved more C# functions to the Sharp Kernels (shark) library.
1831	26nov21	Upgraded Telerik UI for WinForms R3 2021.
1832	24dec21	Improved the spectrum responsiveness when streaming the FFT data from a SpyServer; Improved the resolution of the frequency display for frequencies below 2 MHz. **Upgraded to dotnet 6 with single file build and R2R.** ***A START.BAT file in the installation package temporarily enables the Tiered PGO (Profile Guided Optimizations) before the program starts...*** `set DOTNET_TieredPGO=1` `start sdrsharp.exe`
1833	31dec21	Multi-threaded GFX for smoother display; Many other optimizations for lower resource usage in the lower hardware configurations.
1834	01jan22	The Band Plan plugin now supports the multi-threaded UI; Fixed the text update of the main window; The Frequency Manager plugin now supports the multi-threaded UI; Better property UI updating code.
1835	04jan22	**New display for the Band Plan** and the built-in Frequency Manager to avoid over-crowding the spectrum view. The xml databases are not loaded from the current

		directory of the process, which eases the use of profile-specific entries; Many FFT optimizations; Smoother rendering and more responsive UI even with limited resources; The produced XML files are now indented; More FFT polishing.
1836	05jan22	More robust code for the waterfall update.
1837	05jan22	More graphics optimizations; Smoother frame timing; More graphics optimizations; Smoother frame timing; Fixed a sequential resizing crash that needed to be atomic.
1838	06jan22	Fixed the frequency manager loading; Sharper edges for the bookmarks.
1839	07jan22	Offloaded the main thread from all the real-time UI processing.
1840	08jan22	Rendering API cleanup; Fixed the SpyServer FFT updating.
1841	08jan22	Fixed the FFT display configuration.
1842	08jan22	Added a new hardware accelerated API for the plugin rendering. This can be used like the standard .net Graphics API.
1843	13jan22	Better FFT scheduling to save CPU time while still getting optimal rendering; More drawing APIs; Using the system's threadpool for handling the FFT; Better FFT timing for smoother rendering; Compensate for CPU clock irregulatities in the FFT stream; Added config settings to bypass automatic database update in the Band Plan plugin.
1844	18jan22	Added a new FFT engine with better performance; Added a new FFT API for plugin developers; Lower CPU usage overall; Lower memory usage; Faster and more accurate rendering.
1845	18jan22	Fixed the MPX visualization; Adjusted the latency of the display pipeline; Moved more function to the native kernel library (shark.dll); Added native memory allocation; More performance optimizations to use the new infrastructure.
1846	18jan22	Added more gfx caching for faster rendering.
1847/9	18jan22	One more rendering optimization to accomodate for slow plugins; Added more steps in the rendering pipline of the spectrum analyzer. This allows instant responsiveness while the data is being rendered.
1850	19jan22	Added dynamic latency adjustment to minimize the lag between the visual and the audio paths.
1851	20jan22	More polishing: Lower CPU usage for the same processing quality.
1852	20jan22	Update the visual feedback for the filter band.
1853	20jan22	New Telerik release 2022 R1. **Starting from SDR# release 1853, the DSP will be using a reworked version of the PFFFT (yes, that's not a joke) FFT library.** This surprisingly fast library was modified to fit within the object model of the DSP and will allow faster FFT speeds in the spectrum displays and some filters. Some frequency domain plugins like the Noise Reduction, IF Filter, etc. can also benefit from this improvement. The legacy FFT routines are still available for the old plugins, but the new ones are encapsulated in a simple to use C# class called DFT. Another area of improvement is the deterministic memory management for the buffers. This comes as a side effect of the global rework of the DSP, and will allow a more accurate on-demand adjustment of the used memory. The changes are transparent for the plugins, unless something stupid is being done. The other side effect is the lower memory usage on average. The display components have also been revamped to use a pipelined approach. This includes the sequencing the IQ (or Real) data, planning the FFTs, executing them, timing the display and compensating for the CPU fluctuations. A lot of operations are now hardware accelerated, but will not show as a direct GPU usage. Instead, the dwm.exe (Desktop Window Manager) process will show some extra GPU usage, but it's not that big. The overall electric power usage is lower with these changes, which may be a most welcome improvement for portables. And of course, a lot of polishing has been done and still ongoing.
1854	26jan22	Added support to clear native memory in the UnsafeBuffer class; More FFT polish; Better stream synchronization code.
1855	26mar22	Added assembly resolving for the plugins compiled using a newer version of the .NET SDK.Scaled down the FFT display for the SpyServer client; Many improvements in the

		FFT display components; Added more dependencies for the plugins: System.Data.DataSetExtensions; Better FFT sequencing and timing; Increased the Zoom FFT resolution; Better stop/tear-down sequence for font-ends; Upgraded Telerik to version 2022 R1 SP1; Added forward compatibility for plugins written in more recent versions of the .NET SDK.
1856	28apr22	Replaced the old Noise Reduction processor with a new algorithm: **Natural Intelligence Noise Reduction (NINR).** This results in less artifacts, deeper noise cancelling, and lower CPU usage.
1858	28apr22	Adjusted the NINR spectra smoothing.
1859	01jun22	Added a "Slope" setting to the NINR; Better smoothing algorithm for the NINR to save CPU; New NINR presets.
1860	03jun22	Fixed the initialization of the frequency shift; Reordered the default plugins so that the Noise Blankers process their respective streams before the Noise Reduction plugins; Updated Telerik UI to version R2 2022 (2022.2.510; Clear the spectrum components when resized smaller than the minimum usable surface.
1861	03jul22	GFX code cleanup; **New frequency domain FM demodulation** with improved linearity. As a side effect, the RDS also decodes faster and better; Moe RDS tweaks to match the new demod.
1862	04jul22	Many improvement in the RDS decoder; Added bandwidth margins related to the demodulation sample rate.
1863 1864	04jul22	New filtering processors; New fast PLL for transient RDS signals; Fixed some clicking in FM when changing the filters or the squelch; More code optimizations and cleanup.
1865	06jul22	More RDS tweaks; Tweaked the RDS decoder some more.
1866	08jul22	Improved the AM/DSB audio; Updated the NINR NR defaults to match; Adusted the Audio HPF limits.
1867	08jul22	Adjusted the frequency response for AM/DSB.
1868	08jul22	Adjusted the AM/DSB HPF corner frequency to 30Hz; **Updated Telerik to version R2 2022 SP1.**
1869	15jul22	Fixed the initialization sequence of the AM Co-Channel Canceller.
1870	15jul22	**Added Binaural detection to the DSB mode**; Adjusted the AM audio filter to match DSB; Added deemphasis for LSB/USB when "Lock Carrier" is enabled; "Anti-Fading" results in "Enhanced Mono" output with DSB.
1871	15jul22	Added a check box to switch the Binaural mode on and off; Updated the audio filters to match both modes.
1872	17jul22	Extended the Binaural mode to AM. Now the L and R channel depend on the actual phase information of the transmitted carrier; Many filtering improvements.
1873	19jul22	Many rendering optimizations.
1874	20jul22	Fixed the binaural initialization code.
1875	24jul22	Improved the NINR smoothing algorithm; Updated the quality factor of the audio HPF.
1876 1877	25jul22	Sharper tone detection for the NINR along multiple performance enhancements; Adjusted the HPF for AM/DSB audio.
1878	26jul22	Added hardware acceleration to the NINR algorithm while keeping the same behavior.
1879	29jul22	Handle poorly modulated AM transmitters in the NINR; Lock Carrier is now processed before the IF plugins.
1880	30jul22	Save the waterfall gradient by value rather than by index; Update the gradient indicator in real time after modification; Better handling of the low frequencies in the NINR; Adjust the Q factor of the audio HPF; Added two NINR profile contributions.
1881	31jul22	Improved the Carrier Locker and the FM Detector.
1882	02aug22	New compiler directives.
1883	03aug22	Work around a libusb limitation for device hot selection; Affected Airspy R0, R2, Mini, HF+ Dual, HF+ Discovery.
1884	03aug22	Added quadrature audio output for LSB/USB modes. Some brains are capable of processing it.

1885	04aug22	Added new NINR profiles; added more FFT sizes for better de-noising; **Refactored the Baseband File Player.**
1886	07aug22	**Added a new Micro Tuner panel in Zoom FFT. This panel can be used in conjunction with the AM Co-Channel canceller to get a very deep rejection of the selected station;** Major rework of the Spectrum Analyzer graphic component to allow micro-tuning; The old references should still work with the new extended API; Added new processing hook points for the Micro Tuner plugin; **Added a new AM Co-Channel algorithm with a native implementation.**
1887	08aug22	Major improvement of the Co-Channel Canceller with a much simplified control. No F1 driver license required.
1888	11aug22 12aug22	**New FFT processor for IF/AF filtering, NR, AM CCC, Anti-Fading, and many other key features**; Improved the behavior of the Lock Carrier PLL; Fixed the frequency display in the Spectrum Analyzer; Changed power ratios to amplitude ratios for the output "volume" slider; Updated the NINR profiles to match the new FFT engine; Tweaked the Lock Carrier parameters.
1889	17aug22	Added a new processing hook in the DSP chain to allow the NR to be placed after the Carrier Locker. Improved the FFT processor; Tweaked the NINR profiles. Increased the refresh rate of the IF and AF panels.
1890	20aug22	The "Configure Source" button now shows the Server selection map for the SpyServer client; **Renamed the built-in Audio and Baseband recorders to allow third party equivalents to be loaded;** Updated the quality factor of the audio IIR HPF to prevent ringing near DC; Refactoring and code cleanup; Added a long term stability factor for the Carrier Locker.
1891	22aug22	New scaling for the NINR; Improved the base FFT processor; Added more DSP tools in shark.dll; New scaling in the NINR GUI. Adjusted the NINR profiles.
1892	25aug22 26aug22	**Added a new "Super PLL" to replace the Carrier Locker;** Default to 10sec resilience time for the "Super PLL"; Changed the Slope setting of the NINR to power dB scale; Adjusted the Q factor of the HPF; Force the app's culture too en-us; Set the text rendering compatibility for old plugins; Fail safe creation of the "Audio" directory when the user selects a read-only directory; Tweaked the technical constants for the Anti-Fading, Super PLL and NINR; Extended the plugin API to allow the direct docking of spectrum panels without needing a configuration panel; Split the Zoom FFT plugin into separate plugins with built-in configuration; **Following the popular demand, we arranged the main menu to avoid clogging the UI with plugins. This is an intermediate solution until a fullly fledged plugin manager is implemented;** Added a new API to lock the center frequency to be used by the recording plugins; Reset the Carrier Locker for each session; The Zoom Bar can now be set sticky or displayed on demand to keep more space for the spectrum.
1893	27aug22	Added a compatibility method for older plugins; Added a new NINR profile by RNEI's Rose.
1894	29aug22 30aug22	More UI polish: Main menu and Zoom FFT plugins; Many NINR fidelity and CPU usage improvements; **Added shorter names for the panels with full name and category for the plugin menu; Increased the size of all the buttons and menus; New iconography;** Autosize the columns of the default frequency manager;
1895	02sep22	Let the OS impose the locale.
1896	03sep22	Frequency Manager: Prevent Windows from setting odd size values. Revision 1896
1897	03sep22 06sep22	Updated the SpyServer code to use the latest shark library. Sharper "+" icons. Simplified the NINR controls by replacing the Attack/Decay settings with a single "Time Smoothing" slider. The old smoothing setting was renamed "Frequency Smoothing". Adapted the NINR UI to the new settings. **Replaced the "Enabled" checkbox with a nice toggle button in the DSP plugins**. More UI polish.
1898	08sep22	Better memory alignment for modern CPUs; More accurate side band resizing.

1899	14sep22	Faster loading of the built-in plugins (no reflection); **Optimized** the **frequency step buttons in the main tool bar.**
1900	16sep22 17sep22 21sep22 25sep22	**Upgraded Telerik toolkit to version 2022 R3 (2022.3.913).** **Added a new optional "Auto Tune" control** for the AM Co-Channel Canceller for quick operation without the Micro Tuner. A few DSP optmizations in the AGC. **Enable up/down keys for the frequency step buttons**. A few UI tweaks. Cleaned up the Multi-Notch plugin; Optimized the GC behavior (lower Memory Usage + Fewer interruptions); New compiler optimizations for the DSP; Solution cleanup. (TEST with New Speech Enhancement plugin using Krisp Noise AI Canceller; Krisp cleanup). **Updated Telerik UI toolkit to version 2022 R3 SP1.**
1901	20oct22	Updated build configuration for the upcoming .NET 7; Improved the feature power estimation of the NINR; Updated the NINR profiles to suit the updated noise reduction engine.
1902	09nov22	**Upgraded to .NET 7 and Telerik 2022 R3.**
1903	11nov22	New digit glyphs for the Frequency dial; Updated the deployment config.
1904	17nov22	Fix Calicocat exception handling; **New zoomable digit glyphs for the frequency dial.** The zoom factor can be set using the config "key core.frequencyDialZoom"; More UI polish.
1905	19nov22	UI tweaks: Resized some tool windows.
1906	18dec22 19dec22	CPP Code cleanup; Optimized the AGC to handle AM QSB; Optimized the NINR for detection quality and CPU usage; Updated the NINR profiles to reflect the new core features; Optimized the Anti-fading; Increased the depth of NINR action in the default profiles. **Encapsulated the Step Size menu as a drop-down item of a smaller button to save UI estate. New AGC; Improved FFT processor.**
1907	24jan23	Fixed an update problem when changing the CW shift tone while streaming.
1908	25jan23	**Added support for more calibration options for the Airspy HF+ series.**
1909	25jan23	Added a new "Communications" equalization profile for NFM, but also LSB and USB (when not used with Lock Carrier); Adjusted the deemphasis for NFM.
1910	27feb23	**Upgraded Telerik UI to 2023.1.117**; Tweaked the NINR engine some more; Added FFT Offset and Range config keys for the AF and MPX displays; Updated the NINR default profiles.
1911	07jun23	Merge branch 'master' of https://github.com/touil/sdrsharp; **Upgraded Telerik UI to version 2023 R2** (This version fixes many UI glitches like window resizing and selectors disappearing); **Many RDS decoding improvements**; Code cleanup; Work around some Telerik quirks with the Visual Studio 2012 theme; Moved the native callbacks code to the latest C# syntax; Renamed the "Speech" profile of the NINR to "DX"; Moved the IQ Source selection to the main menu; Replaced NewtonsoftJson with System.Text.Json in the SpyServer client; Improved the performance of the native calls; Added dynamic resizing for the entire UI; Added a new "Weak" profile for the NINR; Added a special AF monitoring DSP hook; Reordered the AF processing so that the de-emphasis happens at the very end of the chain; Improved the memory usage of the NINR engine; Many DSP low level optimizations and code cleanup; Work around Telerik RadSpinEditor so Filter Bandwidth can be editable; Marked the assemblies with DisableRuntimeMarshalling where applicable; Refactored the PI invokes to use dotnet 7/8 code generation instead of runtime calls; Exported more APIs from Sharp Kernels library (shark.dll); New smoothing algos for the NINR.
1912	08jun23	More RDS improvements.
1913	08jun23	Normalized the processing gain through all the FM demodulators.
1914	06jul23	Better handling of theming for native Winforms Checkboxes; Many improvements in the RDS decoder; Smoother FFT speed slider. The old configs may show a slow FFT. Adjust to suit.
1915	08jul23	Optimized the IQ Balancer in shark library. **New RTL-SDR IO interface.**

1916	09jul23	Fixed the AM Co-Channel Canceller: Handle zero buffers. Tweaked the AM CCC defaults.
1917	10jul23	Accelerated the sample conversion for the Baseband File Player source.
1918	16jul23	Many UI improvements; Lower Memory and CPU usage. Better RDS decoding in noisy scenarios. Fixed "Invert Spectrum" for the Baseband File Player.
1919	17jul23 28jul23	**Added an experimental FM MPX Fuzzer to dig noisy RDS with multipath and co-channel interference;** Optimized the internal IQ streaming; Faster closing/loading of the Layout files; Support Blocking and Non Blocking IQ sources; The built-in audio recorder now records the monitoring final stage instead of the pre-deemphasis stage; Improved the MPX Fuzzer; Added dynamic loading of the ATT steps for the HF+ devices; **Upgraded Telerik UI to version R2 2023**; New improved smoothing algorithm for the Noise Reduction; Cleaned up the NINR code and added more smoothing algorithms to choose from; More optimizations and code cleanup for the NINR; Tighter CPU optimizations for the NINR; Normalized the internal audio and mpx paths for 0 dBFS; Added a 3 dB hysteresis to the AM demod; Added dynamic scaling of NFM signals according to their occupied bandwidth; New simplified AGC algorithm.
1920	04aug23 25apr24	Added a learning function for the AGC. This helps setting the AGC for optimal level stability (as opposed to pumping); **Added preliminary support for AOR radios**; Many UI speed optimizations; Added a new psychoacoustic function to the NINR; Added a new Low Power / Fast NR engine; Improved the NINR-LP (Low Power) engine; Many low level optimizations for the DSP library; Fixed Unmute fnuction when the volume is at the minimum; Code cleanup; Many DSP optimizations; UI code refactoring; Added a new FFT processor and referenced it in the NINR and IF Notch plugin; Better handling of phase distortions in the NINR; Code cleanup; Fixed the bandwidth display fo HF+ devices; Optimized the CPU usage of the NINR along with a new time smoothing algorithm; Updated the NINR profiles; Introduced a "Sticky Locking" to switch between the "dumb" regular PLL and the enhanced PLL with smart tracking of fading signals; Added generic support for antenna, preamp and att selection for AOR radios; **Upgraded to .NET 8.0**: Updated the native FFT processor; Updated the .NET wrappers for the FFT processor; Updated Telerik UI to version 2003 R3 SP1; Added configurable FPS setting for the Spectrum Analyzer component. The new config key is refreshFPS; Updated the AM Squelch for a wider range; Added buttons with flat icons to the Baseband File Player; Many UI optimizations; Code cleanup to suit .net 8; Added AOR per-device reconfigurable UI; Many performance optimizations for the spectrum display; Optimized the FFT rendering; Optimized the FFT smoothing; More DSP optimizations targeting SSE4.2; Arranged the config keys and defaults; Optimized the FFT buffering; Optimized the FrequencyEdit timer; New FFT sequencing algorithm; Fine tuned the FFT pixel fitting algorithm; Improved the CPU usage of the FFT sequencer; Use native memory allocator for the UnsafeBuffer; Fix theme saving/loading; Black background color for the buttons in the Baseband File player; Improved the processing of the AM Co-Channel Canceller and the AM Anti-Fading; Many DSP tweaks; **Refactored the Airspy HF+ controller**; Refactored the Signal Diagnostics plugin to use a more accurate power estimation method; Massive code cleanup; Fixed a spurious lock on stop bug with HF+ receivers; Run the app at a normal priority; Improved the NINR frequency/time resolution; Added more NR profiles; Updated the SpyServer with the latest DSP from SDR#; Normalized the streamed FFT to dBFS; Configured the release build for the new SpyServer; Optimized the FFT sequencer for lower memory bandwidth, Added opportunistic synchronization for the FFT rendering; Enhanced the display timing; Force absolute paths for config and layout files; A few optimizations for the SpyServer; Version update; SpyServer code cleanup; SDR# code cleanup; New plugin loading strategy; Added a gfx tear-down workaround in the spectrum display components for dotnet 8; Updated the build configuration; Added border snapping for the floating plugin windows; The closed floating windows are now removed from the docking database; Optimized the

		loading of plugin UIs; Fixed AOR logging; Upgraded Telerik UI to R1 2024; Fixed VFO update for dotnet 8; Added new Windows 11 themes; **Moved most of the main UI to Winforms; Moved RadButton's to Winforms Button's; Converted the remaining menus to Winforms**; **Moved the FrontEnd UIs and the plugins to native WinForms controls; Removed the remaining Telerik behavior (except the docking manager); More Native UI customizations;** Adjusted the layout initialization code for the native controls; Fixed the init of the main window position; Optimized the drawing of the custom controls; Removed the Telerik RadDock and replaced with DockPanelSuite; More UI performance tweaks; Implemented the extended plugin tool window mechanism with the new docking manager; The newly opened plugins start floating and centered around the main window; More UI tweaks; Fixed the loading of the layout; More UI tweaks; Updated the built-in Frequency Manager to use the native controls instead of Telerik; More UI polishing; Added Bias-Tee support for the HF+ series; Extended the rang of the MicroTuner to +/- 30 Hz; **Improved the Band Plan display**; Added more buffering control settings in the config file; Allow audio latency setting up to 500ms**; Improved the RDS decoder some more;** UI improvements; Added theming to altrnating DataGridView rows; Implemented Immersive Dark Mode for WDM; Extended the theming services to use the system's dark mode styles; Improved the support for system theming; Added Bias-Tee for RTL-SDR Blog dongles; More theming support; Added a custom trackbar component with support for theming; Cleanup; Better layout for the R2/Mini controller; Initialize the immersive frame for dock panels; Optimized the layout of the HF+ controller; Added frequency tracking to the built-in frequency manager; Optimized the FFT streaming to reduce the memory usage some more; Fixed the RTLTCP controller; Added custom arrow painting for the comboboxes with theming support; Updated the AOR source to use the new themed combobox; A few ehnacement for the themed combobox; Improved the theming of the numeric up/down and the comboboxes; Implemented a flicker-less combobox; Fixed a mysterious null exception inside user32.dll; Workaround for Windows Common Controls's editable Combobox; More UI tweaks; Increased the resolution of the Multi-Notch; A few UI improvements; Code cleanup; Improved the FFT streaming; Code cleanup; Improved the memory management of the FFT stream; Ported the control panels to the themed UI: - HackRF - RTL USB - RTL TCP - RFSPACE SDR IP - RFSPACE SDR IQ; More GUI optimizations; Optimized the initialization of the IQ sources; Added the theming support for the File Player dialogs; Added a map control for the SpyServer Client; Saved the location and zoom level of the SpyServer map; Fixed RTL-SDR UI; Improved the docking system; Add fallback code for C# 11.0; Added a layout helper for old plugins; Set the default FFT visualization panels; Fixed the auto-scrolling of the built-in frequency manager; Updated the theming service; Included Calico CAT plugin by Tag Loomis to the main package with support for dotnet 8; Better theming code; Added lazy loading for Calico CAT; Fixed the Sharp Slicer source; Many UI optimizations; Added sticky hint to the volume/mute button; Fixed the frequency update code; Added fallback path for the built-in Frequency Manager; Enhanced the device enumeration for RTL-SDR; Added a Clear button for the multi-notch; Force the control of the visibility for the extended plugin panels by their parent plugins; A few layout optimizations; Improved the behavior and memory usage of the custom slider; Merge pull request from the PureWinforms branch. **Getting rid of Telerik UI. Once for all.**

Conclusion

A journey of miles begins meter by meter and here we have traveled a very long way together indeed...
If this book has helped you to better appreciate SDRsharp I consider it an excellent achievement.
I have personally tested everything in the book, now it is your turn to do the same!

I end this journey of ours together with a quote that is worth all of them.

"In radio we have an appropriate instrument for uniting the peoples of the world, for making their voices, needs and aspirations heard.
The significance of this modern medium is thus fully revealed: a wide avenue of communication for the improvement of our mutual relations is at our disposal; we have only to follow its course in a spirit of tolerance and sympathetic understanding, ready to use the achievements of science and human ingenuity for the common good.
I am firmly convinced of the possibility of achieving this ideal..."

(Guglielmo Marconi, March 1937)

Glossary

ADS-B - Automatic Dependent Surveillance - Broadcast
AF – Alternate Frequencies (RDS)
AGC – Automatic Gain Control (for optimal listening levels)
AIS – Automatic Identification System
ALE - Automatic Link Establishment / HF standard for initiating/supporting digital comms
AM – Amplitude Modulation
AOS - Acquisition of Signal (or Satellite)
APRS - Automatic Packet Reporting System (radioamateur data transmission system)
BALUN - BALanced-UNbalanced, device for adapting an unbalanced/balanced line
BW – BandWidth
Carrier - modulated radio wave carrying an information
CAT - Computer Aided Tuning (to control rtx via computer)
CPU – Central Processing Unit
CTCSS - Continuous Tone-Coded Squelch System (analogic)
CW – Continuous Wave
DAB/DAB+ - Digital Audio Broadcasting
dB - decibel
dBFS - Decibels Full Scale
DCS - Digital Coded Squelch (digital)
DGPS – Differential Global Positioning System
DMR – Digital Mobile Radio, is one of the main open standards for radio communications
DPI - Dots Per Inch, graphic screen resolution
dPMR – digital Private Mobile radio, other open radio communication standard
DRM – Digital Radio Mondiale – digital radio in HF
DSB – Double Side Band
DSD+ – Digital Speech Decoder, software for decoding multistandard digital audio signals
DSP – Digital Signal Processing
DTMF - Dual-tone multi-frequency
DX – Long-distance radio connection
EON – Enchanced Other Networks (RDS)
FFT – Fast Fourier Transform
FIC - Fast Information Channel (DAB)
FM - Frequency Modulation
FM-DX - search for distant FM radio stations under particular propagation conditions
FSK – Frequency shift keying
FT8 - Franke-Taylor design, 8-FSK modulation
GMDSS - Global Maritime Distress Safety System (world maritime security system)
GMT - Greenwich Mean Time (related to summer time, so different from UTC)
GNSS - Global Navigation Satellite System, geo-radiolocation system
GPS - Global Positioning System, USA satellite positioning and navigation system
HDR – High Dynamic Range
HF – High Frequency (3-30 MHz, decametric 100-10 m)
HUB - hardware connecting various devices to the computer
ICAO - International Civil Aviation Organization
IF – Intermediate Frequency
KSPS – kilosample per second (10^3 * sps)
LDOC - Long Distance Operational Control
LF – Low Frequency (30 / 300 kHz, kilometric 10-1 km)
LNA – Low Noise Amplifier
LOS - Loss of Signal (or Satellite)

LSB – Lower Side Band
mA – milliAmpere (submultiple Ampere)
MDS - MultiDimensional Scaling
MF – Medium Frequency (300 kHz / 3 MHz, ettometric 1 km-100m)
MPX – Multiplexing
MSC - Main Service Channel (DAB)
MSPS - Megasample per second (10^6 * sps)
MUX - short for "Multiplex", technique for transmitting digital radio/TV signals
MW – Medium wave
MWARA - Major World Air Route Areas
NDB - Non-Directional beacons
NFM o **FMN**– Narrow Frequency Modulation
PI – Programme Identification (RDS)
PLL – Phase-Locked Loop
PPM – Parts per Million
PS or **PSN** – Programme Service Name (RDS)
PTY – Program Type (RDS)
QRSS - Very slow speed Morse code
QSB - amateur radio Q code indicating fading (variation in signal strength over time)
QSO - amateur radio Q code to indicate a communication or connection
QTH – amateur radio Q code indicating own geographical position
RAW – unprocessed data
RDARA - Regional and Domestic Air Route Area
RDS – Radio Data System
RF – Radio Frequency
RT – Radio Text (RDS)
RTTY – Radioteletype
SAM – Synchronous AM
SAR – Search And Rescue
SMA – SubMiniature type A (coaxial connector)
SSB - Single Side Band (transmissions)
SSTV – Slow Scan TV
TA – Travel Announcements (RDS)
TCP - Transmission Control Protocol
TCXO – Temperature Compensated Crystal Oscillator
TII - Transmitter Ident Information (DAB)
TMC – Traffic Message Channel (RDS)
TP – Traffic Programme (RDS)
UHF – Ultra High Frequency (300 MHz / 3 GHz, decimetric 1m-100mm)
USB – Upper Side Band
UTC – Universal Time Coordinated
VFO – Variable Frequency Oscillator
VHF –Very High Frequency (30 / 300 MHz, metric 10-1 m)
VIS - Vertical Interval Signaling (SSTV)
VLF – Very Low Frequency (3 / 30 kHz, miriametric 100-10 km)
VOLMET - vol météo (Weather Information for Aircraft in Flight)
Volt – unit of measurement of electric potential
Watt – unit of measurement of power
WEFAX – Weatherfax
WFM o **FMW** – Wide Frequency Modulation


AIRSPY
Spyverter
AIRSPY
AIRSPY
Design: Paolo Romani IZ1MLL