Quick Start Guide

Installation on 8, 8.1, 10 and 11

Airspy is a plug-and-play device and does not require any particular driver installation on Windows download and install the right driver for you.

There are exceptions where the original configuration of the PC does not allow the automatic insta (/download) should be installed manually with the following procedure:

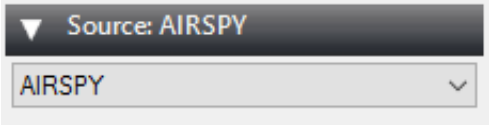
- Download and unzip the WinUSB Compatibility Driver (/download)
- Open the device manager and select Airspy
- Select “Update Driver” then “Browse My Computer” to the inf file

Installation on Windows Server or Windows X

Please don’t. They are not supported.

Using SDR#

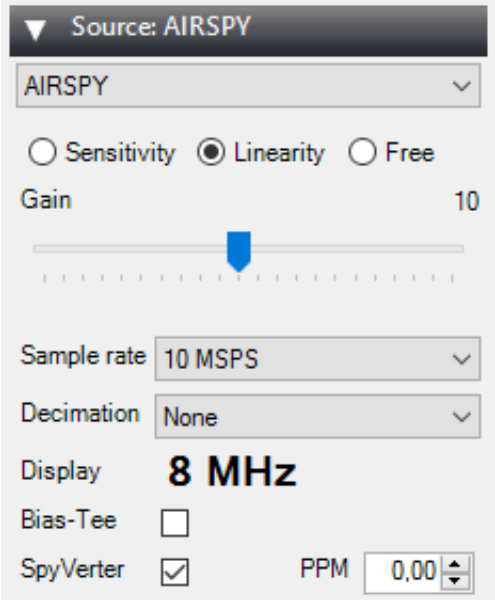
Airspy was designed by the same people who developed the SDR# software (http://sdrsharp.com) s
its powerful features.
First, go to the download page (/download) and get a copy. Then run SDRSharp.exe and select the '



Then next step is the gain configuration. As depicted in this screen shot, there are many gain mode

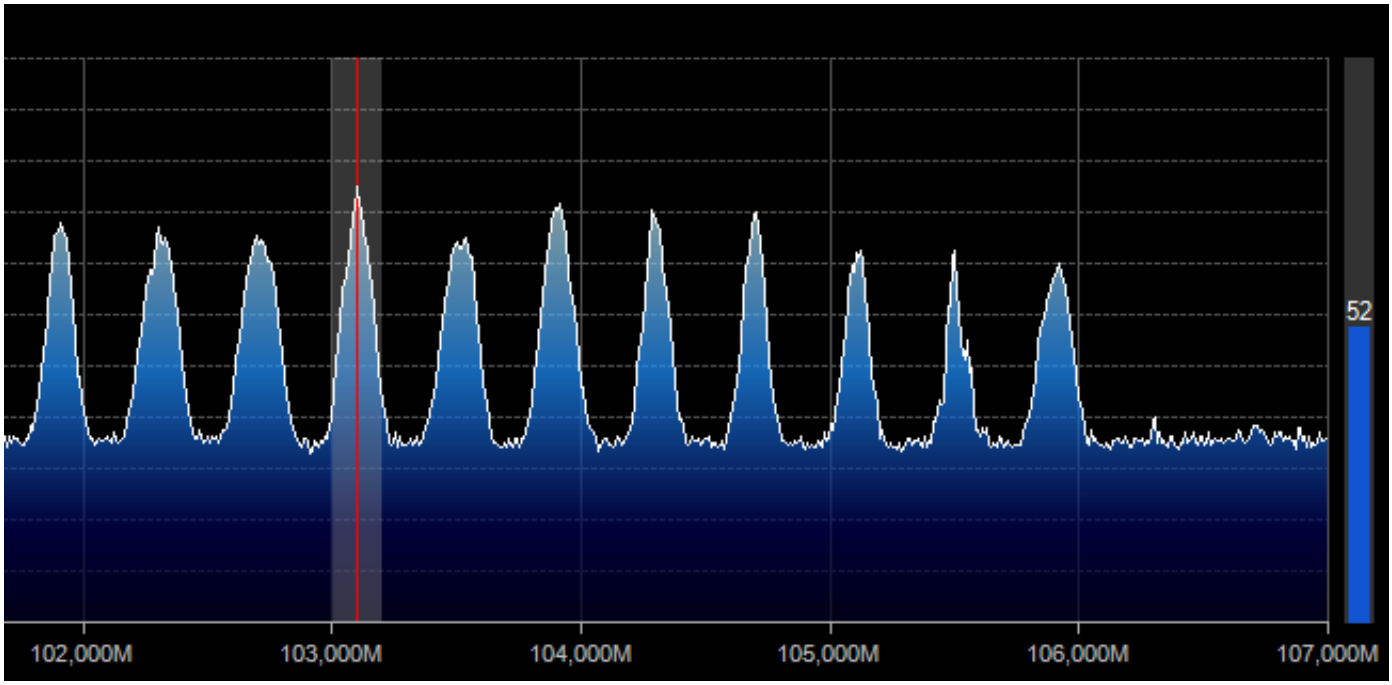
- Sensitivity
- Linearity
- Free (Custom)

The "Linearity" mode is the one you want to start with:



The following fine tuning procedure ensures you have the maximum SNR on the signal of interest v

- Start with the minimum gain
- Increase the gain until the natural noise floor rises by about 5dB
- Check that increasing the gain no longer increase SNR. If not, increase the gain by an extra notcl
- Fine tune to maximize the SNR (the blue bar graph on the right)



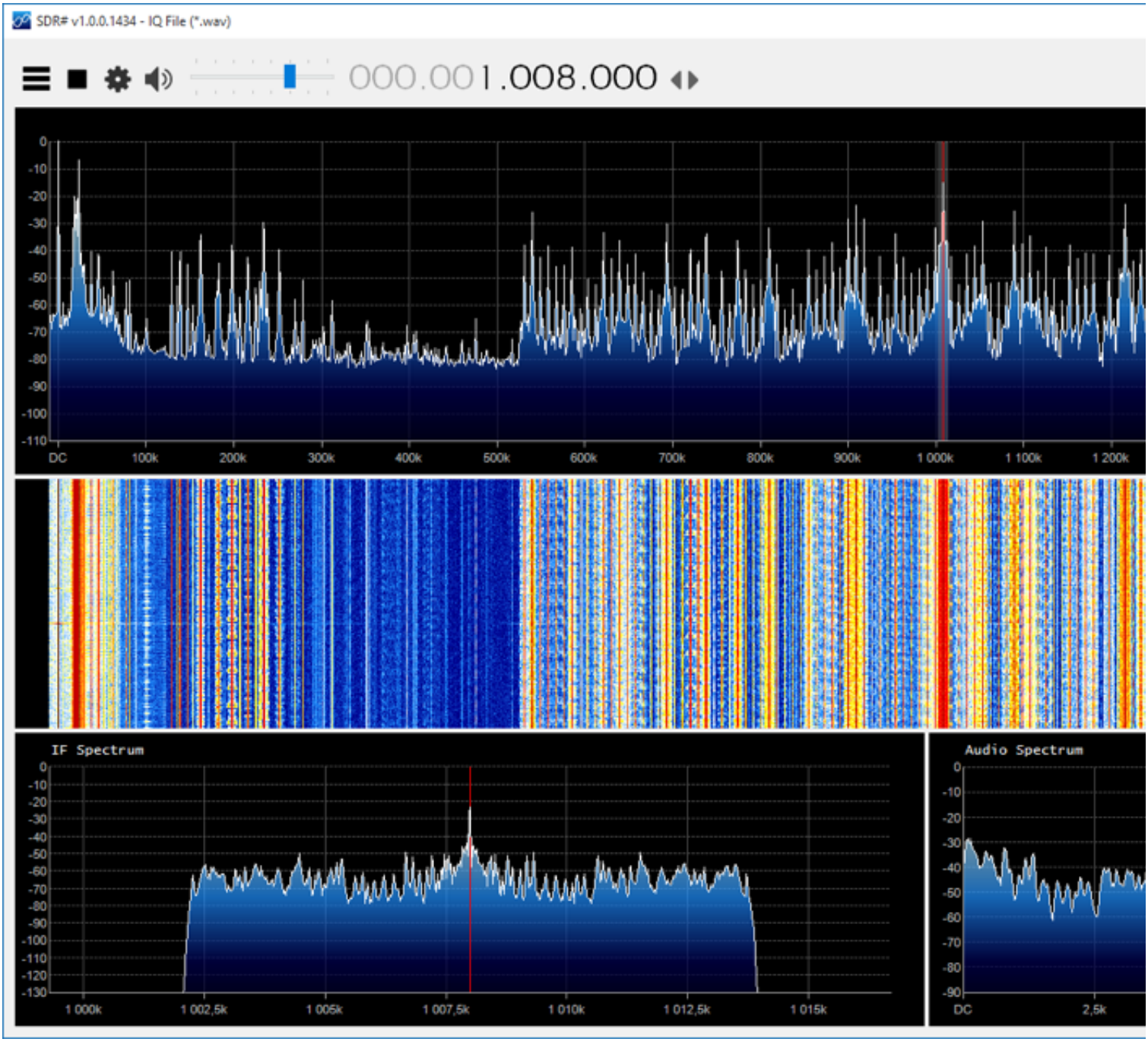
In any case, you should make sure the RF noise floor just overrides the quantization noise floor of t

Using the SpyVerter

Airspy has an option to cover the HF bands: The SpyVerter. It is a low loss, high dynamic range, rug
same kind that is used in very high end HF rigs like the Elecraft K3. It offers HF coverage starting ne
L of the Airspy alone. The default software settings allow Airspy to power the SpyVerter via the bias
connect the “IF Output” of the SpyVerter to the RF Input of the Airspy via the supplied barrel adapt

It is recommended to use the Linearity gain mode in HF.

To verify your setup, plug a HF antenna and tune the AM broadcast band. You should get a view lik



(/downloads/spyverter_vlf_lw_mw.png)

Using the SPY Server for Remote SDR Operati

Presentation

Since revision 1553, we offer a new utility for the Airspy series to convert a Raspberry Pi, an Odroid
featured SDR server.
The SPY Server allows you to connect many SDR# “clients” to the same Airspy or RTL-SDR device ov

How does it work?

When only one client is connected, full control (center frequency, gain) is granted. When there are
device and the RF gain are locked.
The default settings allow using the server in the LAN with optimal latency. The “Full IQ” option allo
SDR client (SDR#). If the network bandwidth is not sufficient to handle the traffic, uncheck the “Full
data and reduced IQ data that it necessary for decoding one VFO. The bandwidth savings allows us
connection.

Configuration

The SPY Server comes with a standard configuration file that allows you using Airspy or RTL-SDR di
the same server, you can have multiple configuration files and pass the right one as parameter to t
If the SpyVerter (/products/spyverter-r2) is used, the following parameters should be set in **spyserv**

```
# 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
```

Running on Windows

Edit the configuration file if necessary and double click on **spyserver.exe**.

Running on x86 Linux

First, make sure you are have the latest version of **libairspy** (https://github.com/airspy/airspyone_l
Extract “spyserver_linux_x86” and “spyserver.config” from the standard zip, chmod and run.

```
$ unzip sdrsharp-86.zip
$ chmod a+x spyserver_linux_x86
$ ./spyserver_linux_x86 spyserver.config
```

Running on ARM Linux (Raspberry Pi or Odroid)

First, make sure you are have the latest version of **libairspy** (https://github.com/airspy/airspyone_h
Download and extract the standard ARM package and run.

```
wget http://airspy.com/downloads/spyserver-arm32.tgz
tar xzf spyserver-arm32.tgz
./spyserver spyserver.config
```

You can also use the **automated** installation script by Rodrigo Pérez which will do everything for yo

```
wget -qO - http://airspy.com/downloads/install-spyserver-rpi.sh | sudo bash
```

Linux Troubleshooting

What could go wrong in the Linux world? Well, many things (see the GPL sections 15 and 16 (https:/
most obvious ones people reported so far:

- The program runs, but cannot acquire the device: The user mode driver is probaby not up to da
- Sometimes using **sudo** makes things just magically work.
- "`GLIBCXX_3.4.21′ not found" => Try installing GCC 5

Using the ADSB decoder in embedded ARM bo

Prerequisite

You need firmware version 1.0 rc7 or newer to use Airspy as an ADSB receiver. Get the latest updat
https://github.com/airspy/firmware/releases (https://github.com/airspy/firmware/releases)
READ the instructions before doing anything.

Raspberry Pi 3 Performance Tuning

The latest Raspbian distros come with a new default CPU governor that is not suitable for sustaine
high frequency (1200MHz), get very hot, then throttle back to a very low frequency (600 MHz). The
two extremes and the USB controller will drop whole packets, which could affect both the number
The following settings will force the "performance" governor at a much more acceptable CPU frequ

and without sacrificing the performance of the decodes.
If very high MLAT reliability is required, the following switche should be applied to the decoder:

```
sudo /home/pi/airspy_adsb -l 30005:beast -x -m 12
```

This will run all the decoding in one core using 5 signal phases instead of 10, which requires half of
Additionally, we recommend to put these lines in **/etc/rc.local** :

```
# /!\ RPi3 Only! /!\
echo performance > /sys/devices/system/cpu/cpu0/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu1/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu2/cpufreq/scaling_governor
echo performance > /sys/devices/system/cpu/cpu3/cpufreq/scaling_governor
```

Then append these lines to **/boot/config.txt** :

```
# /!\ RPi3 Only! /!\
force_turbo=1
arm_freq=800
```

Lastly, you can also disable the “FIQ Fix” that limits the number of USB interrupts. Less USB interru
the USB hardware with the risk of missing more buffers.
To disable the “FIQ Fix”, you can add the following parameter to the line in **/boot/cmdline.txt** :

```
dwc_otg.fiq_fix_enable=0
```

Standard setup (non FlightAware)

The whole process for setting up a high performance ADSB receiver has been simplified. For the Ra
following commands:

```
mkdir airspy
cd airspy
wget http://airspy.com/downloads/airspy_adsb-linux-arm.tgz
tar xzf airspy_adsb-linux-arm.tgz
sudo ./airspy_adsb -l 30005:beast -l 47806:asavr -p -m 12 -x &
```

FlightAware Integration

The easiest way to integrate Airspy into an existing FlightAware PiAware is described here (https://
dump1090-fa-configuration/44343).
If you don’t need all the details, you can just use this neat installation script by FlightAware membe
images:

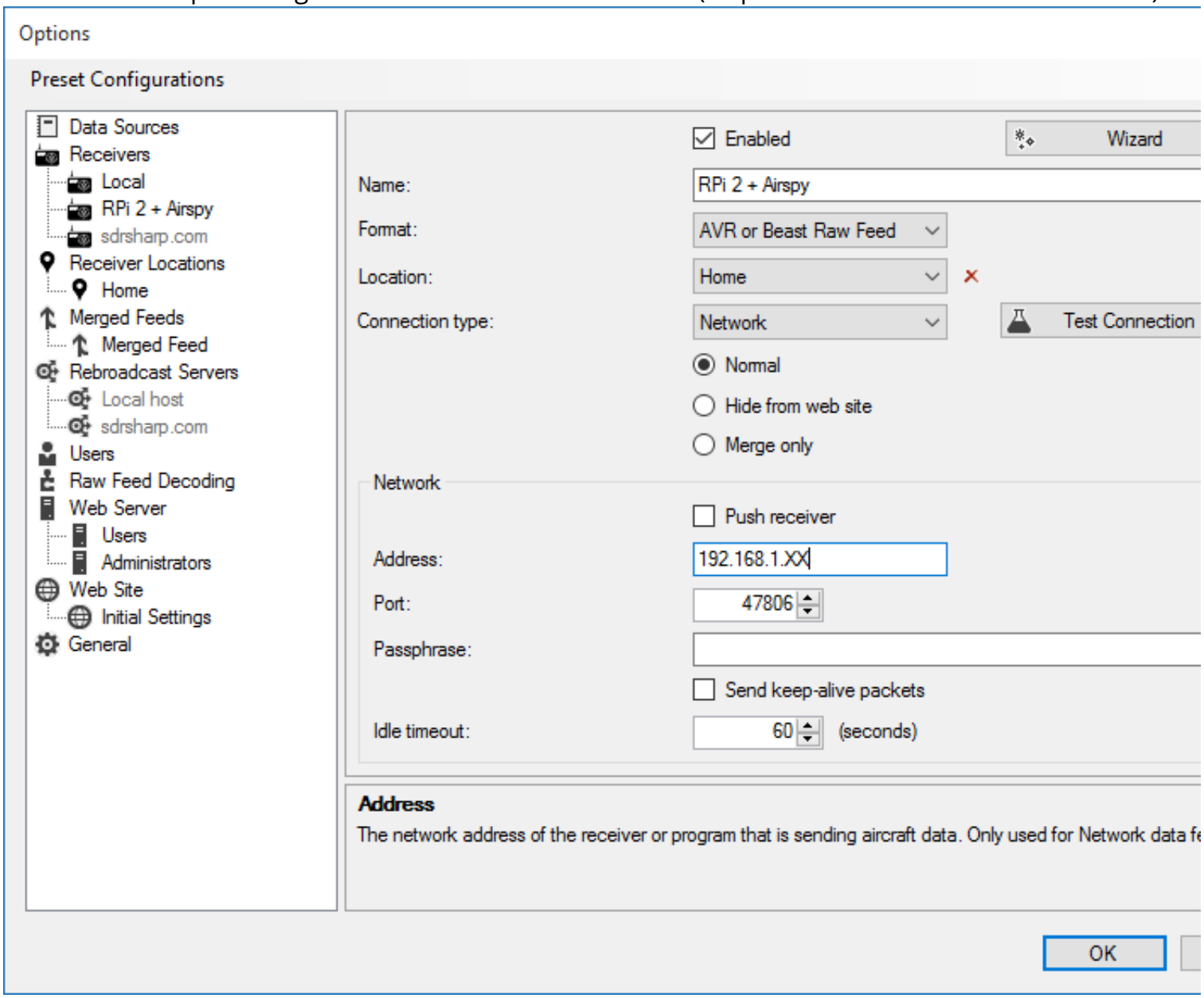
```
sudo bash -c "$(wget -O - https://raw.githubusercontent.com/wiedehopf/airspy-conf/master/install.sh)"
```

Operation

Once you have your decoder up and running, you can use any aircraft tracking software that suppo

Virtual Radar Sever

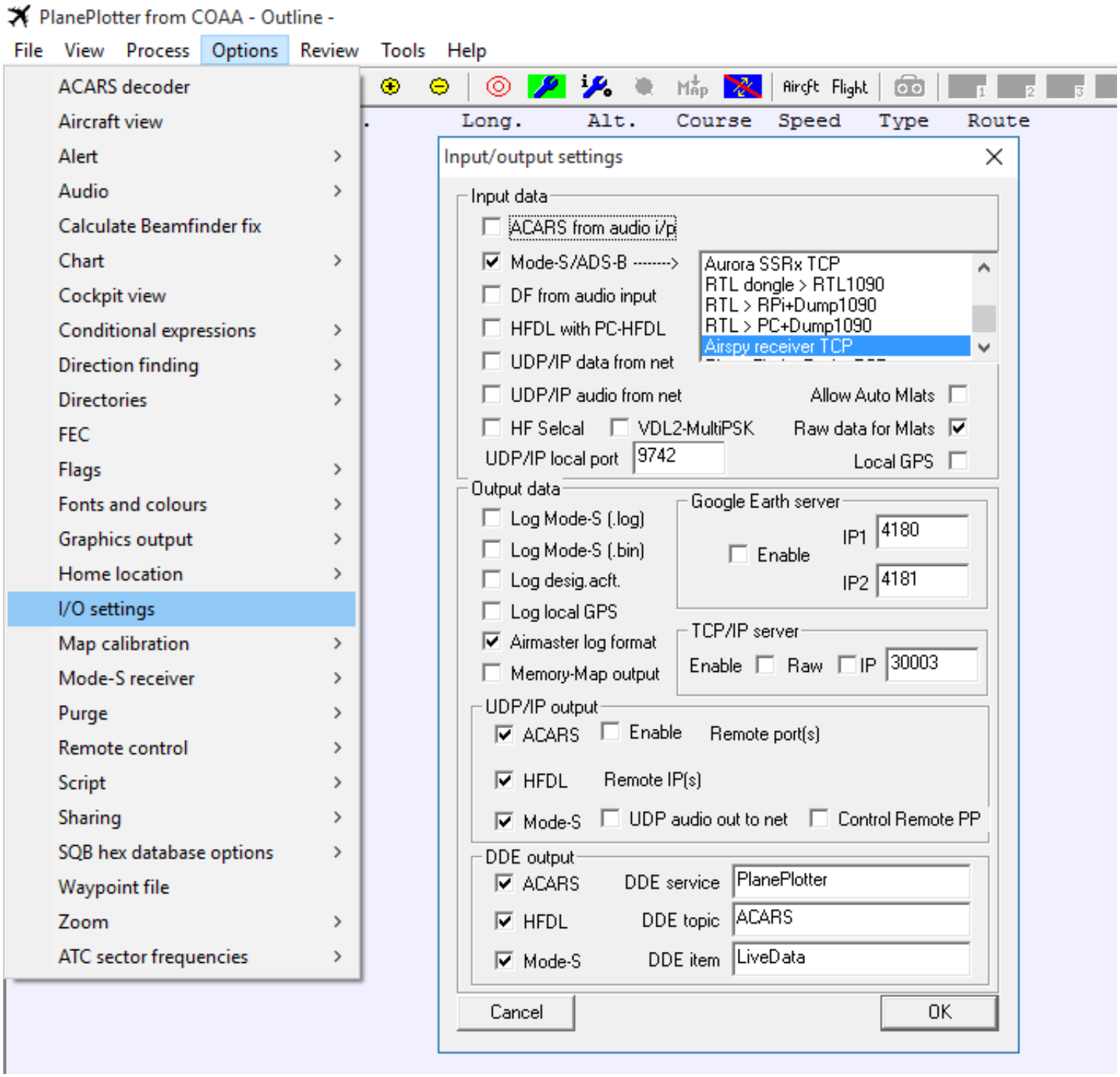
Here's an example configuration for Virtual Radar Server (http://www.virtualradarserver.co.uk/):



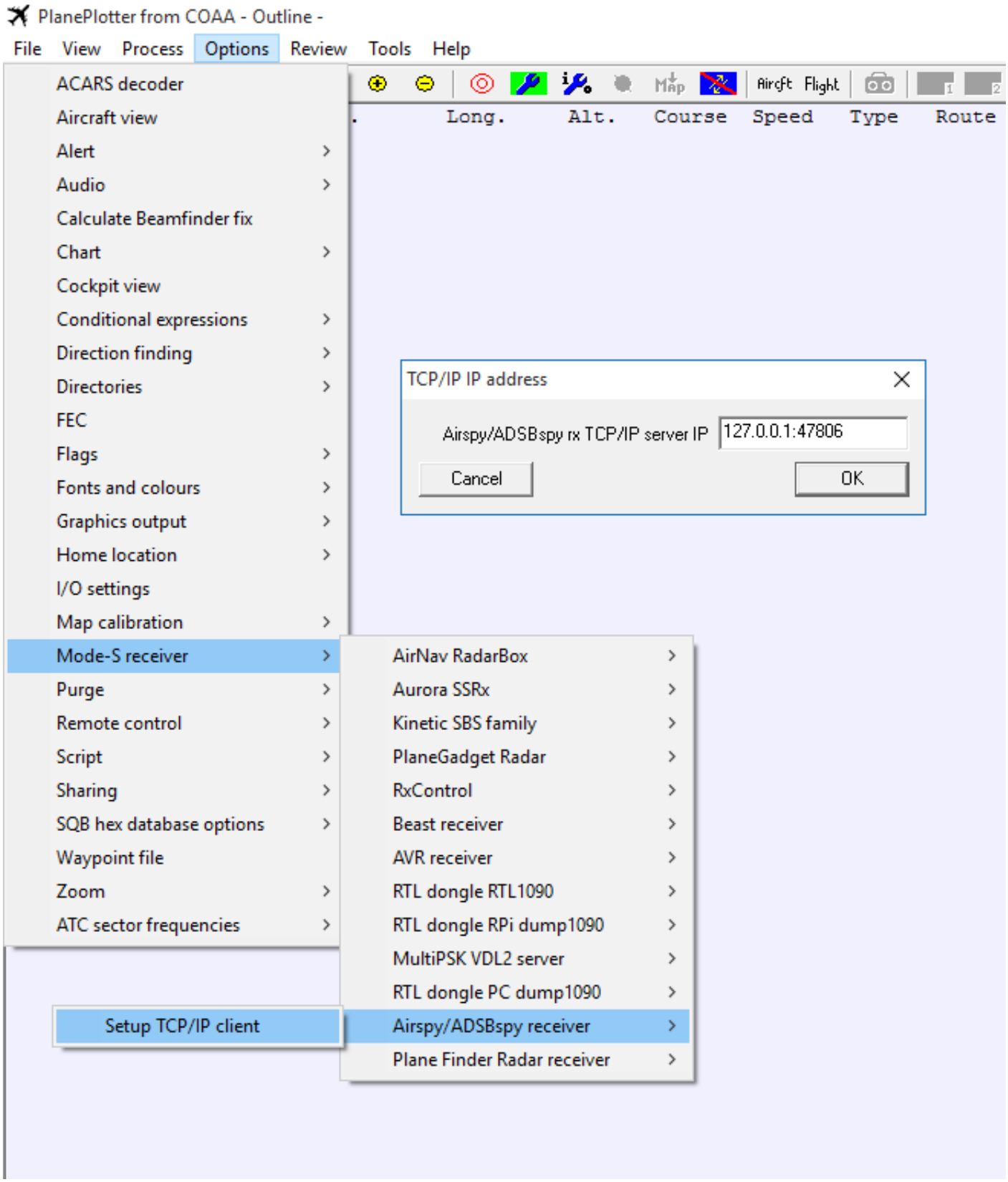
(/downloads/airspy_pi_vrs_config.png)

Plane Plotter

For Plane Plotter (http://www.coaa.co.uk/planeplotter.htm), you can use this configuration:



(/downloads/pp_config1.png)



(/downloads/pp_config2.png)

The effects of oversampling and decimation

The decimation option allows you to trade some of the instantaneous bandwidth to get more bit re
rule of thumb is 3dB every time you divide the bandwidth by two. So, decimation by two gives you
on.
To further leverage the effects of the decimation, you can also readjust the gain using the same pr
gives you more SNR after decimation. So basically, when you use a higher decimation rate, adjust f

The Bias-Tee option: 4.5v @ 50mA

In some cases, one may need a very low noise figure. For example, some satellite communication s
0.2dB. In such case, using an external preamplifier near the antenna system can improve the overa
preamplifier directly from Airspy by injecting DC at **4.5v**. Of course, such a preamp should have its
amplified RF signals. The current budget is limited to **50mA**, so that's enough to run most of the m
Another use case one can imagine is to switch between antennas using that bias signal or even pov
LF and HF, like the SpyVerter mentioned above.

Calibration

Starting from June 2016, Airspy R2 units are calibrated in the factory up to ~0.05 PPM. This calibrati
used for the firmware, so upgrading the firmware won't delete it unless you do it by yourself.
The procedure to recalibrate a unit is as follows:

- Open the calibration tool (AirspyCalibrate.exe (/download/)),
- Clear the correction,
- Reset the device (unplug/plug again),
- Inject a known and accurate signal,
- Set its exact frequency in the tool,
- Click on "Calibrate",
- Reset the device.

Note that this procedure requires firmware version 1.0-RC9 (https://github.com/airspy/firmware/re

Troubleshooting PC performance problems

We created a set of command line utilities to help troubleshooting the performance problems.

- Download the latest release of the tools package (https://github.com/airspy/host/releases)
- Open a console (cmd.exe) and run `airspy_rx -r NUL -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 y
    - Use another USB port
    - Update your USB drivers (Prefer OEM drivers to generic ones)
    - Check your anti-virus or any other CPU heavy task

- See also our Wiki: https://github.com/airspy/host/wiki/Troubleshooting (https://github.com/airsp
  configuration/solutions)

Known issues

- Slow computers won't be able to run Airspy at 10MSPS without dropping samples. That's a lot of
  modern machine as described in the minimum hardware requirements.
- In a fresh Windows 7/8/10 installation, Windows Update will grab the generic Microsoft USB driv
  operation and it is advised to update them with the latest version from your OEM. For example,
  some PCs from choppy to perfectly smooth.
- Some old generation USB 3.0 controllers are not compatible with USB 2.0 devices at high speed.
  problem. And again, make sure you run Airspy in a modern machine as described in the minimu
- In some PCs, the experimental 2.5MSPS mode is vulnerable to USB noise. This will be improved
  to get lower sample rates is to use some decimation with the 10MSPS stream.
- USB Hubs won't work at the full sampling rate. Just don't use them.
- Beware of overheating. Keep your unit in open air environment while running.

That's all folks!