

Hydrogen line observations: From frugal to advanced

Part 3: Receivers and Software (Backends)

Wolfgang Herrmann and the Astropeler "Hydrogen Team"

1. Introduction

Observing the emission of neutral hydrogen from our galaxy has become a very achievable target for amateurs. Low cost components have become readily available over the recent years which substantially facilitate the observation possibilities. There are quite a few options to approach this goal, varying in technical and financial effort. Achievable results will be different, but as will be demonstrated some basic observations are possible with a very frugal approach.

This series of articles intends to give an overview of various options and to do a comparison between these. Each component of a receiving chain will be dealt with one by one: The antenna, the low noise amplifier, filter and the receiver itself (the "backend" in radio astronomy parlance). Once the individual components are covered, a comparison between various setups using these components is done. Actual observations are performed in order to allow an apples to apples comparison to the extent possible.

This third part of the series will deal with the backend options and corresponding software.

2. Reminder: Basic setup of a hydrogen line receiving system

As a reminder from the previous article, the basic setup for hydrogen line observation consists of an antenna, a low noise amplifier (LNA), a filter, some more amplifier(s) and the receiver (backend) as shown in Fig. 1.

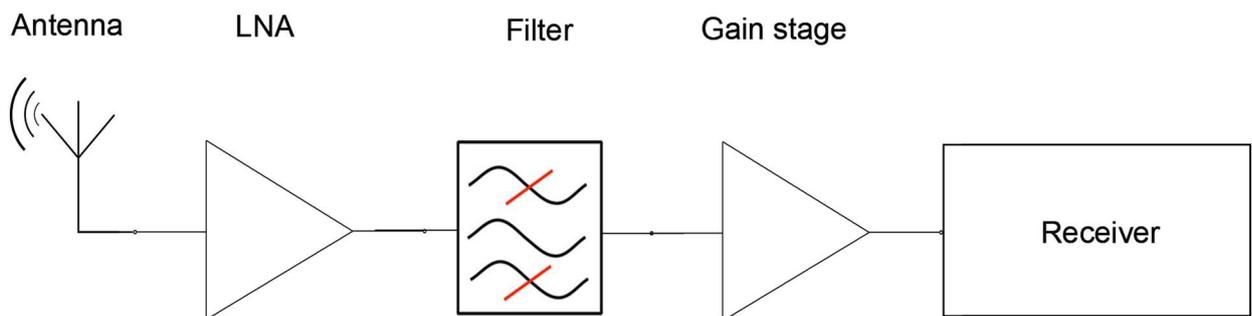


Figure 1: Generic setup

In the two previous articles we have dealt with various antennas, amplifiers and filters. This article will now continue with the remaining part of receiving chain, the receiver and the corresponding software, collectively the "backend".

3. Approach to backends

In the case of the detection of interstellar hydrogen one needs to record a spectrum at around 1420.405 MHz. Traditionally the approach has been to use a receiver with a limited bandwidth which is scanned stepwise through the total bandwidth of interest and thus recording the spectrum piece by piece. However, this technique is no longer the optimum way of doing it. The best approach today is to do a Fast Fourier Transform (FFT) over the full bandwidth of the incoming signal. The big advantage of this is that the full spectrum can be recorded during the full observation time. When scanning stepwise only a part of the spectrum is observed at any given point in time. Therefore, the FFT technique will give a much better signal to noise ratio for a given duration of an observation.

This technique has become available for amateurs with software defined radios (SDR). Here the main part of the analysis and in particular the FFT is done by a computer.

Therefore, essentially two components are required:

- A device which converts the analogue RF signal into digital format to allow processing of the signal. The typical flow is that the device provides complex samples (called In-phase and Quadrature-phase or I/Q) samples
- A piece of software which takes the incoming I/Q samples and does an FFT on this stream and then provides the resulting spectrum in a usable format.

4. Backend hardware

We first deal with various hardware which can be used. It has to be noted that the hardware presented here are just some of the many options, and the scene is changing rapidly as other devices come onto the market.

4.1. RTL-SDR

This device is widely used and very cheap. It is a small USB stick (see fig. 2 and 4) which is originally intended for digital TV reception. It has been detected that these devices can be used as software defined radios if they are equipped with a RTL283U chip [1].



Figure 2: RTL-SDR

There are several variants which differ in the tuner chip used. Today the most common ones are based on a R820T tuner or a later version, the R820T2. Both cover the range from 24 MHz to 1766 MHz.

A few years back the Elonics E4000 tuner used to be very common in these devices. However, this chip is no longer manufactured and devices based on this tuner are difficult to come by these days. The advantage of this tuner over the R820T is the coverage of frequencies up to 2200 MHz. For the hydrogen detection at 1420 MHz however this is not needed.

Depending on the manufacturer, the RF connection is either via an IEC-PAL antenna connector or a MCX connector as shown in fig 3.



Figure 3: Connector variants (Antenna connector left, MCX right)

Both connector variants are not ideal at 1420 MHz. We found improvements in noise performance if the connector was replaced by a SMA connector or if a cable was soldered to the board with a good connector at the other end.

There are also RTL-SDRs available which come with an SMA connector and are placed in a metal housing, see fig 4.



Figure 4: RTL-SDR in metal housing with SMA connector

4.1.1. Clock accuracy and improvement option

The clock of the RTL-SDR is pretty inaccurate and therefore the received spectrum will be somewhat shifted. This results in inaccuracy for the measured velocity of hydrogen clouds. Typically, we found the error to be about 30 ppm. One can, however, apply a correction if the requirements are not too stringent.

For best results one can do a hardware modification to replace the on-board crystal with an external 28.8 MHz signal.

To do this, the crystal has to be removed. A coax cable is then connected to one of the crystal pin positions on the board as shown in fig. 5. It should be fastened by some means (I used hot glue). The outer conductor of the coax can be connected to some convenient place with ground connection. In the example shown in fig. 5 this is done via the green wire.

The required signal is about 200 mV with a frequency of 28.8 MHz.

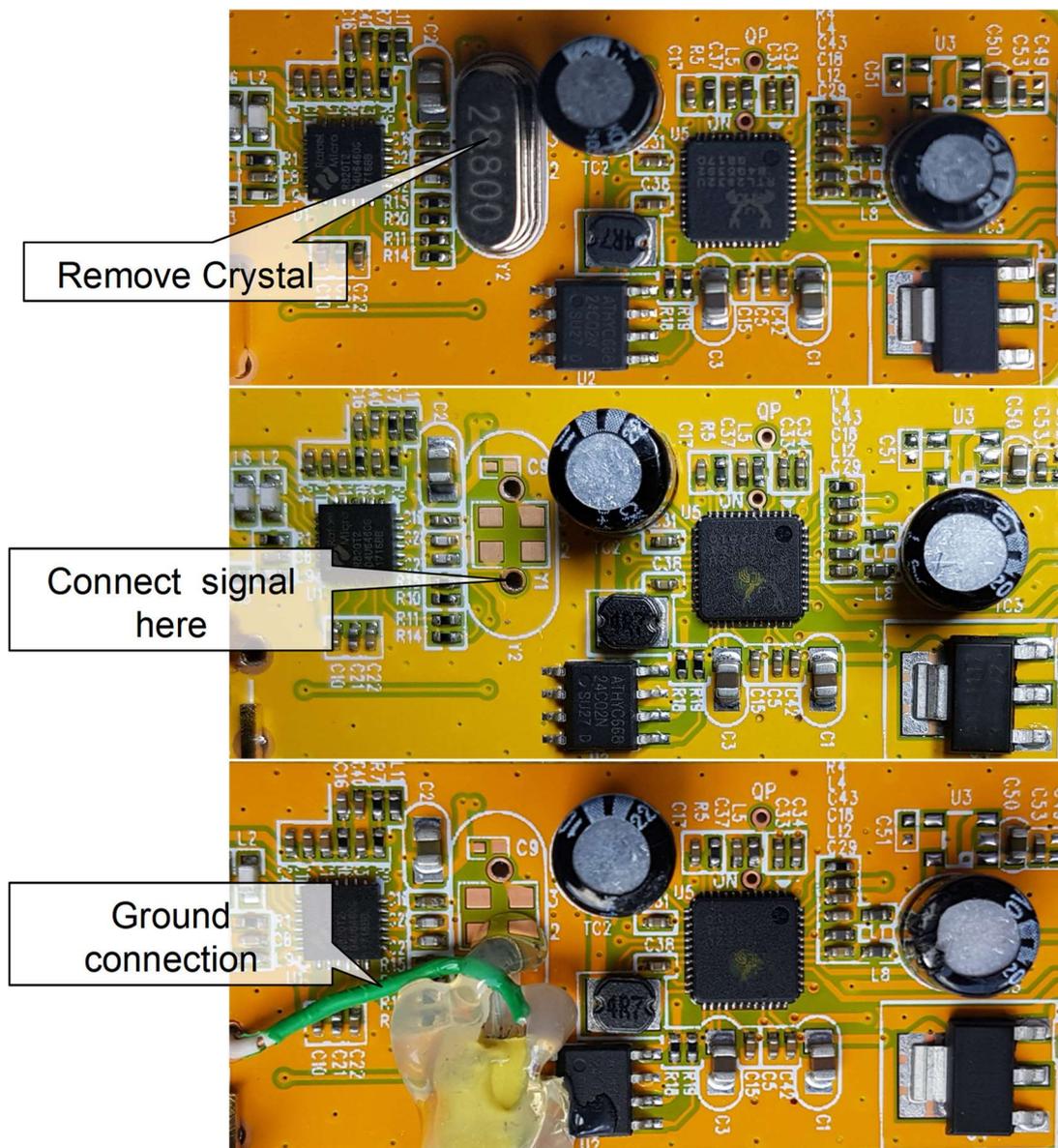


Figure 5: Modification of clock on a RTL-SDR.

4.2. Hack-RF

The HackRF is a popular software defined radio covering a frequency band of 1 MHz to 6 GHz with a maximum bandwidth of 20 MHz, see fig. 6. It can be clocked externally for enhanced frequency accuracy which is desirable for radio astronomy (RA) applications [2].



Figure 6: HackRF, picture from: <https://github.com/fd0/hackrf-one-pictures>

4.3. Chinese Clone of the Hack-RF

A clone of the HackRF is sold by Chinese suppliers via eBay. As the design is open source (at least I believe so) this should be legal.

The version we have acquired is a board covered by some acrylic sheets (removed for the fig. 7 below), however these sheets are not an enclosure as in the original version.



Figure 7: HackRF clone, cover removed

The first experiences with this unit were very disappointing. There were lots of strong spurious lines in the spectrum, and the frequency was off even with an accurate external clock. It came nowhere near to the original. In this state it would have been useless for any radio astronomy work. However, it turned out that many of the issues could be cured by an upgrade to the latest firmware.

4.4. ADALM-Pluto

A fairly recent addition to the choice of software defined radios is the ADALM Pluto by Analog Devices, shown in fig 8. It is based on the AD9363 transceiver chip. It supports the range from 325 to 3800 MHz and can be "hacked" to support 70 to 6000 MHz.



Figure 8: Analog Devices ADALM-Pluto

The ADALM-Pluto does not have a connector for an external clock. It is, however, possible to provide an external clock by doing a small hardware modification. A description how to do this can be found at [3]. We have not tried this modification yet.

4.5. Ettus Research N200

Ettus Research is offering a great variety of SDRs. We have (partially) tested the USRP N200, which is a network device communicating over Gigabit Ethernet, see fig. 9. It is a modular device where a motherboard can be equipped with various different RF daughterboards.

All measurements reported here are with the SBX daughterboard (400 to 4400 MHz). This board, however, failed during the tests so that only measurements of the spectral response could be done.

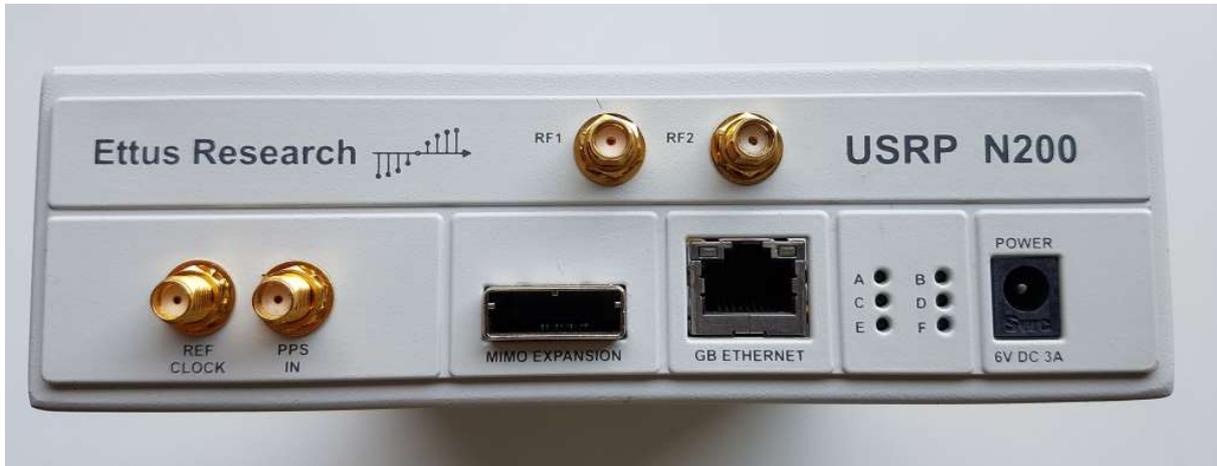


Figure 9: Ettus Research N200

4.6. Other options, not tested

As mentioned, there are very many other software defined radio devices on the market which may also serve well for the purpose. We did not have these devices for testing so we cannot state anything about the performance based on own experience. Such devices are:

- Lime SDR, and variants thereof such as Lime SDR Mini
- Blade RF
- Ettus Research line of products other than the tested N 200
- AirSpy (several versions)
- SDRPlay (several versions)

It should be noted that the Lime SDR is the basis for the RASDR project initiated by SARA.

A comprehensive overview of the software defined radio market can be found at [4].

Finally, a note on the "Funcube Dongle": While this may be a good choice for other applications, this device is not well suited for hydrogen detection. The bandwidth of the device is too small to cover a typical hydrogen emission line, and therefore one would need to do scanning which greatly reduces the efficiency of detection as outlined above.

4.7. Pricing

Reference [4] is giving indicative pricing for the units tested as well as for the other options listed above.

4.8. Testing of backend hardware

4.8.1. Spectral response

One of the things which matters is the spectral response of the SDR. In the case of hydrogen line observations, we are looking for small changes in the background spectrum. These are obviously easier to detect if the spectral response of the SDR is relatively flat and not distorted by any spurious lines.

Test method

The spectral flatness was tested by applying a broadband noise test signal. It was generated by a noise source which was band limited by a Minicircuits VBFZ-1400-S+ filter which has been described in the previous article. We checked the flatness of the test signal +/- 10 MHz around the frequency of the hydrogen line.

Within this bandwidth, our test signal was fairly flat (within about 0.2 dB) allowing a good assessment of the spectral response of the devices, see fig 10.

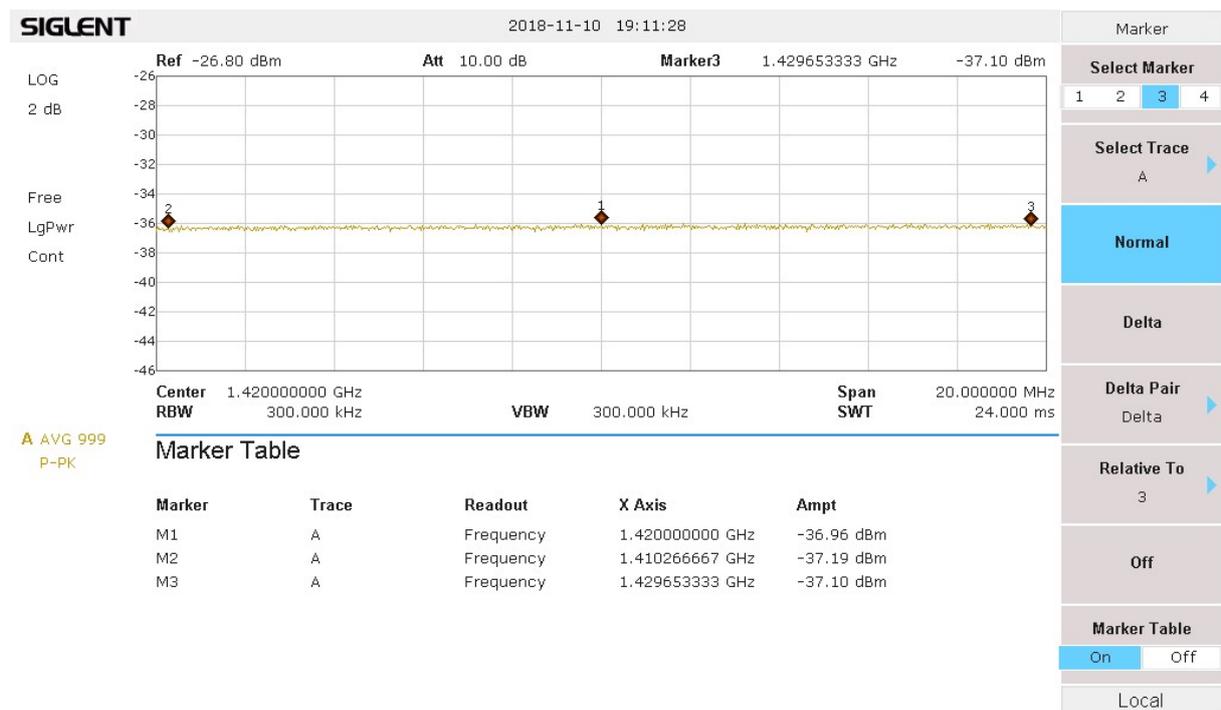


Figure 10: Test signal for spectral response measurements

This signal was recorded by the SDR under test and the spectrum was then analyzed with the same software as it is used for actual observations and which is described in section 5 of this article. Please note that the plots have a linear vertical scale (not dB!).

Measurement with 2 MHz sample rate

The graph below (fig. 11) shows the response of the various SDR using a sample rate of 2 MHz. The vertical scale is linear as one would typically use in radio astronomy.

The horizontal scale is frequency in MHz (top scale) and converted to the corresponding velocity for a hydrogen rest frequency of 1420.405 MHz (bottom scale). Please note that we used the standard convention of radio astronomy (blue shifted left and red shifted right) which means higher frequencies are to the left and lower frequencies are to the right.

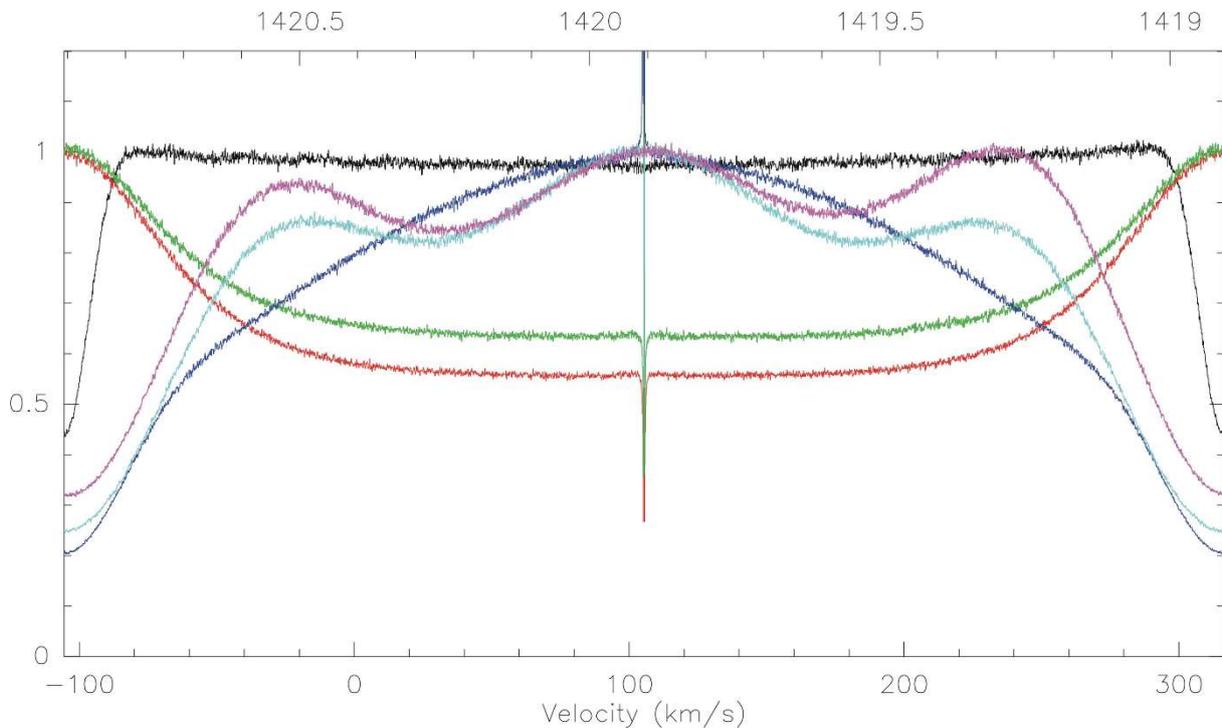


Figure 11: Spectral response of different SDRs, 2 MHz sample rate

The spectrum has not been centred at 0 km/s on purpose. At the centre frequency many SDRs have a DC offset spike which would then overlap with any hydrogen line which is not desirable. All spectra have been re-scaled (or normalized) for the purpose of comparison so the maximum intensity corresponds to unity.

- RTL-SDR with R820 / R820T2 tuner, violet line
These devices all show a "wavy" response which will make it more difficult to see low intensity hydrogen lines
- RTL-SDR with E4000 tuner, light blue line
Also a "wavy" response, a bit more symmetrical
- HackRF (original), orange line
There is a fairly flat region but only in the centre 60% of the spectrum
- HackRF (clone), green line
Quite similar to the original
- USRP, blue line
Fairly uneven response
- ADAM Pluto, black line
Very even response across the spectrum with a bit of ripple

Measurement with 5 MHz sample rate

The same measurement as before however with 5 MHz sample rate is shown in fig. 12. Since the RTL-SDR do not support higher sample rates than 2.4 MHz, only the other SDRs have been tested. Note that is plot shows a wider frequency and velocity range, corresponding to the higher sample rate and wider receiver bandwidth.

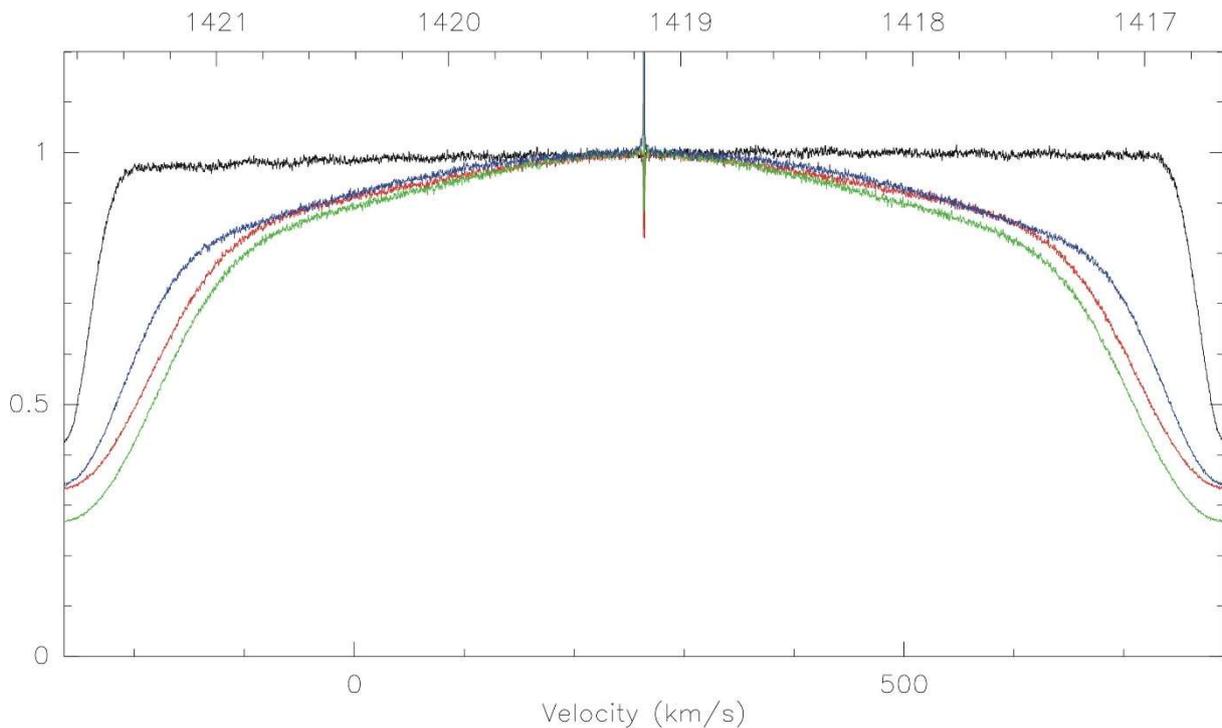


Figure 12: Spectral response of different SDRs, 5 MHz sample rate

- HackRF (original), orange line
The response is curved. It is interesting to note that the shape is very different from what is observed at 2 MHz sample rate
- HackRF (clone), green line
Quite similar to the original
- USRP, blue line
Also similar to the HackRF
- ADAM Pluto, black line
Very even response across the spectrum with a bit of ripple

Measurement with 10 MHz sample rate

Again, the same measurement, now with 10 MHz sample rate. The results are shown in fig. 13.

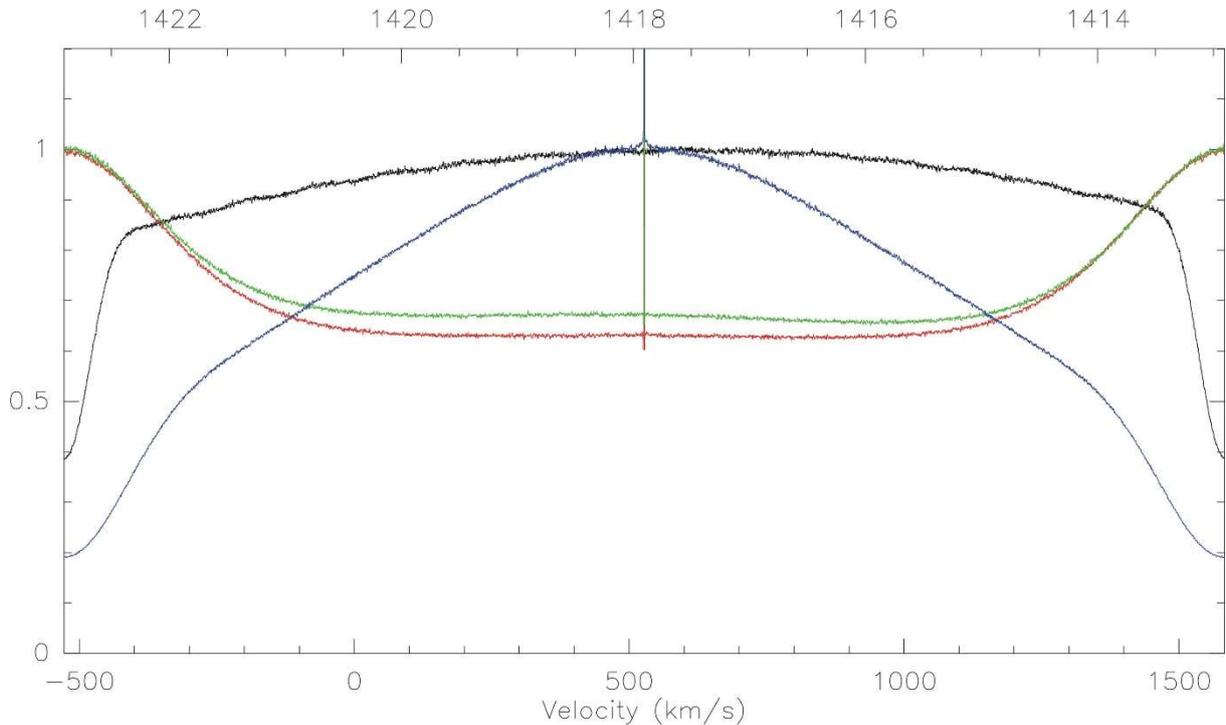


Figure 13: Spectral response of different SDRs, 10 MHz sample rate

- HackRF (original), orange line
The curve is now again similar to what is measured with 2 MHz sample rate
- HackRF (clone), green line
Quite similar to the original
- USRP, blue line
Significant roll-off to both sides similar to what one gets at 2 MHz sample rate
- ADAM Pluto, black line
Flattest response, but not as flat as at lower sample rates

4.8.2. Allan time

In order to detect weak signals longer integration times are frequently employed. The noise decreases with the square root of the integration time as determined by the radiometer equation:

$$\sigma_T \approx \frac{T_{\text{sys}}}{\sqrt{\Delta\nu_{\text{RF}}\tau}} \quad (1)$$

where σ_T is the rms fluctuation of the observed signal, T_{sys} the system temperature, $\Delta\nu_{\text{RF}}$ the observing bandwidth and τ the integration time.

However, improving the signal to noise ratio (SNR) by increasing the integration time may not go on indefinitely. Due to imperfections and fluctuations of the receiving chain, at some point in time longer integration may no longer improve the SNR. In fact, it may even decrease in some cases.

The time at which further integration is no longer improving the rms noise as determined by the radiometer equation is called Allan time.

This can be tested by determining the root mean square (rms) noise for various integration times. If the rms noise is plotted against the integration time in a dual logarithmic plot, the result should be a straight line with a slope of -0.5.

Test method

A band limited noise signal of about -120dBm/Hz was combined via a splitter with a carrier signal of -95 dBm. This combined signal was fed into the SDR under test. Spectra were taken with 770 Hz resolution. The integration time started at 10 seconds and was increased up to 20,000 seconds. For each of these measurements the rms noise and the signal to noise ratio was determined. The rms noise and integration time were plotted in a dual logarithmic graph.

Results

The good news is that all SDRs tested did not show any significant deviation from the theoretical (straight line) behaviour. Therefore, one can conclude that all devices are usable at least up to integration times of 20,000 seconds (about 5.5 hrs) which should be sufficient for most practical applications.

Therefore, only two of the results are shown below in fig. 14 and 15 for the Noolec RTL and the HackRF clone respectively. The other devices show equivalent behaviour.

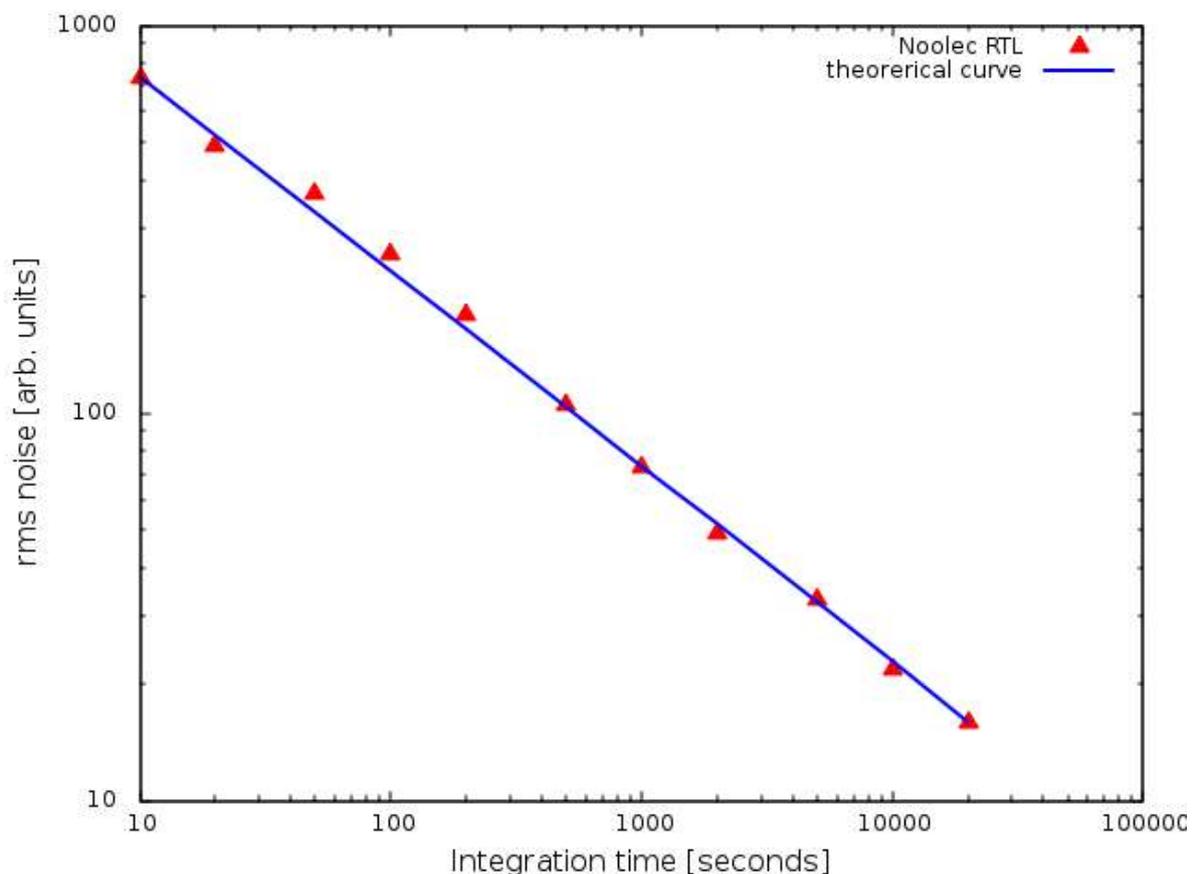


Figure 14: Allan plot for the Noolec RTL

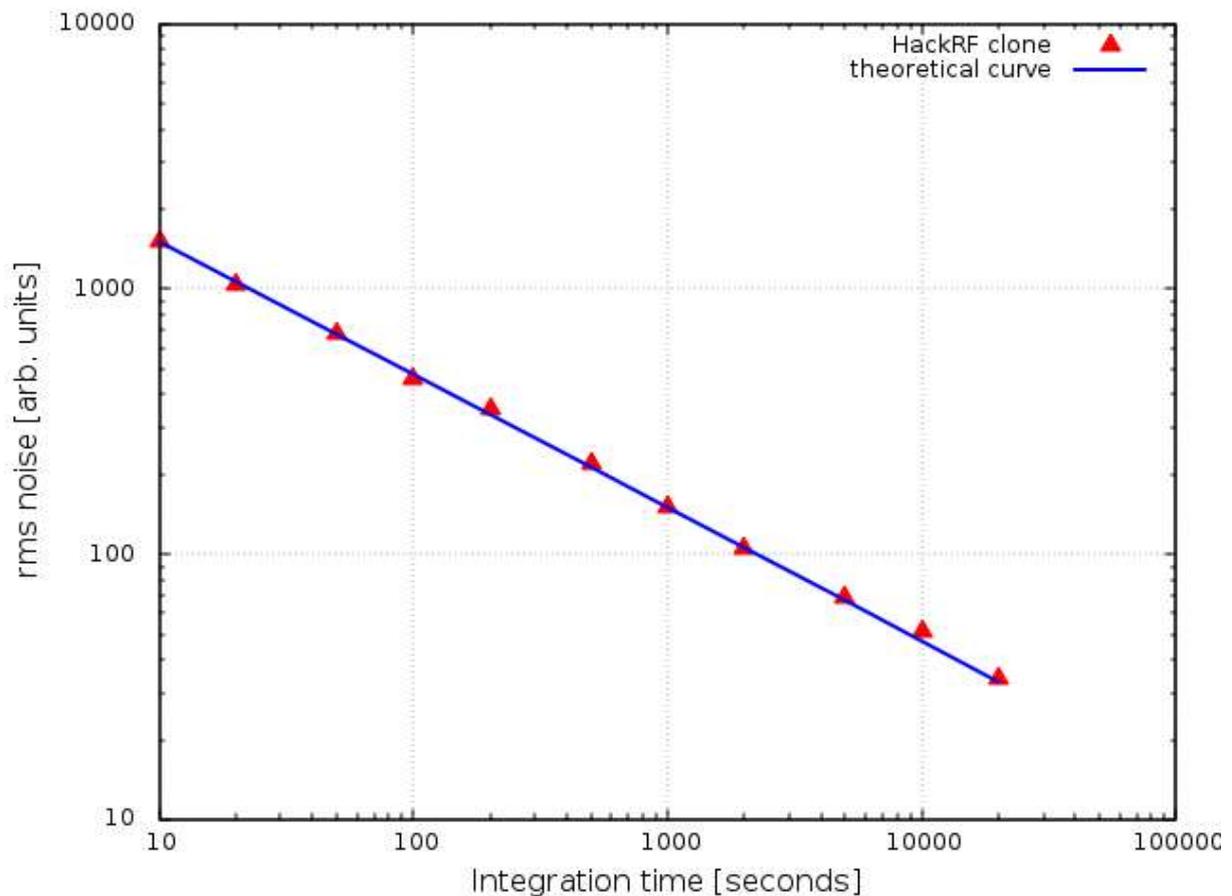


Figure 15: Allan plot for the HackRF clone

4.8.3. Gain stability

For pure spectral measurements the gain stability is of less concern unless one needs to do calibrated intensity measurements. However, if a telescope is also intended to be used for the detection of continuum sources, gain stability is of paramount importance. Due to this we have also looked at the gain stability.

Test method

A stable noise signal band limited to about 20 MHz around 1420 MHz was connected to the SDRs under test. The spectral density was about -145 dBm/Hz. The SDR was allowed to reach thermal equilibrium in idle state by connecting its USB port and letting it sit for about one hour before starting the test. After that, spectra were recorded every second for about 10,000 seconds. Sampling rates used were 2048 KHz for the RTL-SDR and 6 MHz for the other devices. The gain was set to maximum for the respective device under test.

The spectra were recorded with a resolution of 512 channels. The innermost 300 channels were added to form a total power value. The reason for restricting to 300 channels was to avoid the areas of strongly varying spectral response in some of the devices.

The test result is presented as the variation of the total power over time.

RTL-SDR

Two RTL-SDRs were tested: One with a metal enclosure (the one shown in fig. 4) and one without any housing (pure board). Both devices show a very significant "run in" time where the recorded power drops by almost 2 dB. This is most likely a thermal effect where the sampling increases the chip temperature. The run-in time of the device in the enclosure (green line in fig. 16) is significantly longer than for the other device (blue line in fig. 16). This can be explained by its substantial higher thermal mass.

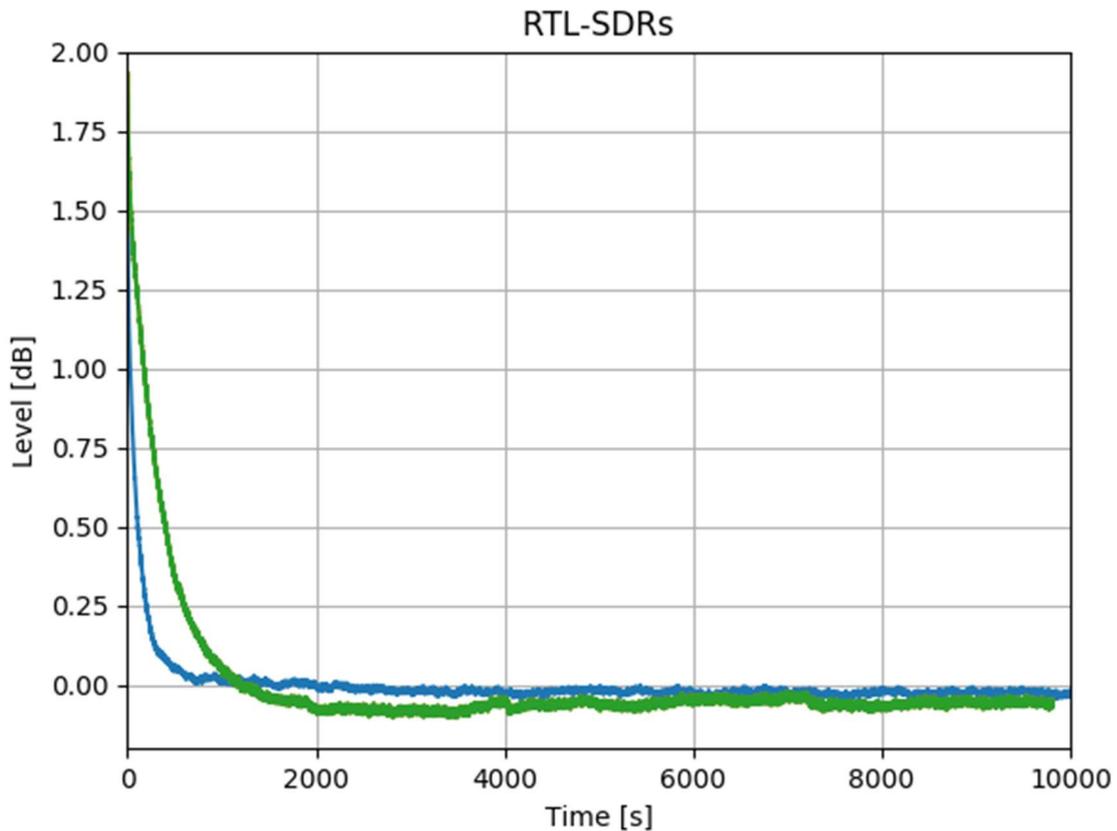


Figure 16: Total power variation, RTL-SDR

HackRF and HackRF clone

Both devices also show a significant run in gain variation - although much less than the RTL-SDRs. The HackRF clone seems to have a bit of a drift over time as shown in fig. 17.

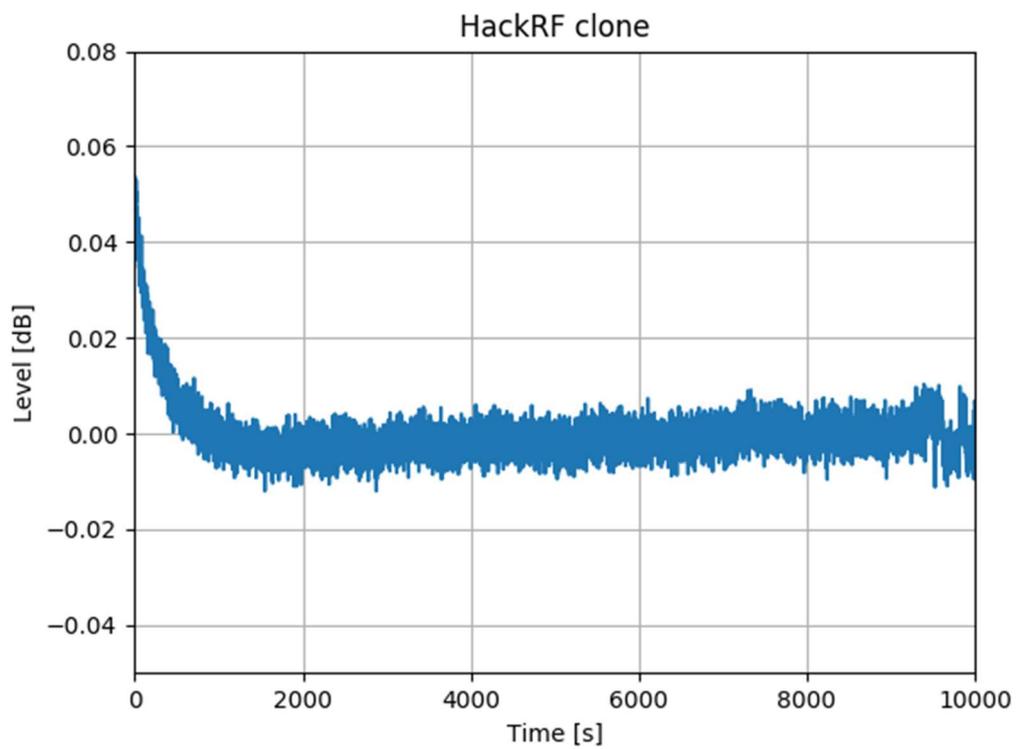
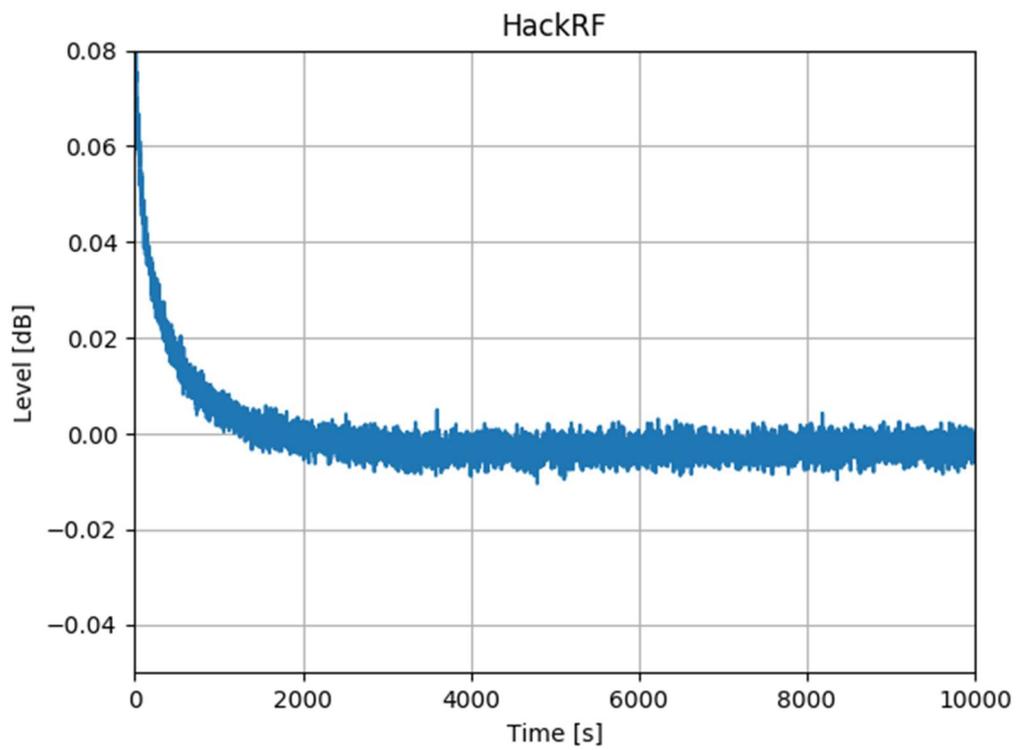


Figure 17: Total power variation, HackRF and HackRF clone

ADALM Pluto

There is no observable thermal run in with the ADALM Pluto and the measured power remains constant over the test time of 10,000 seconds as shown in fig 18.

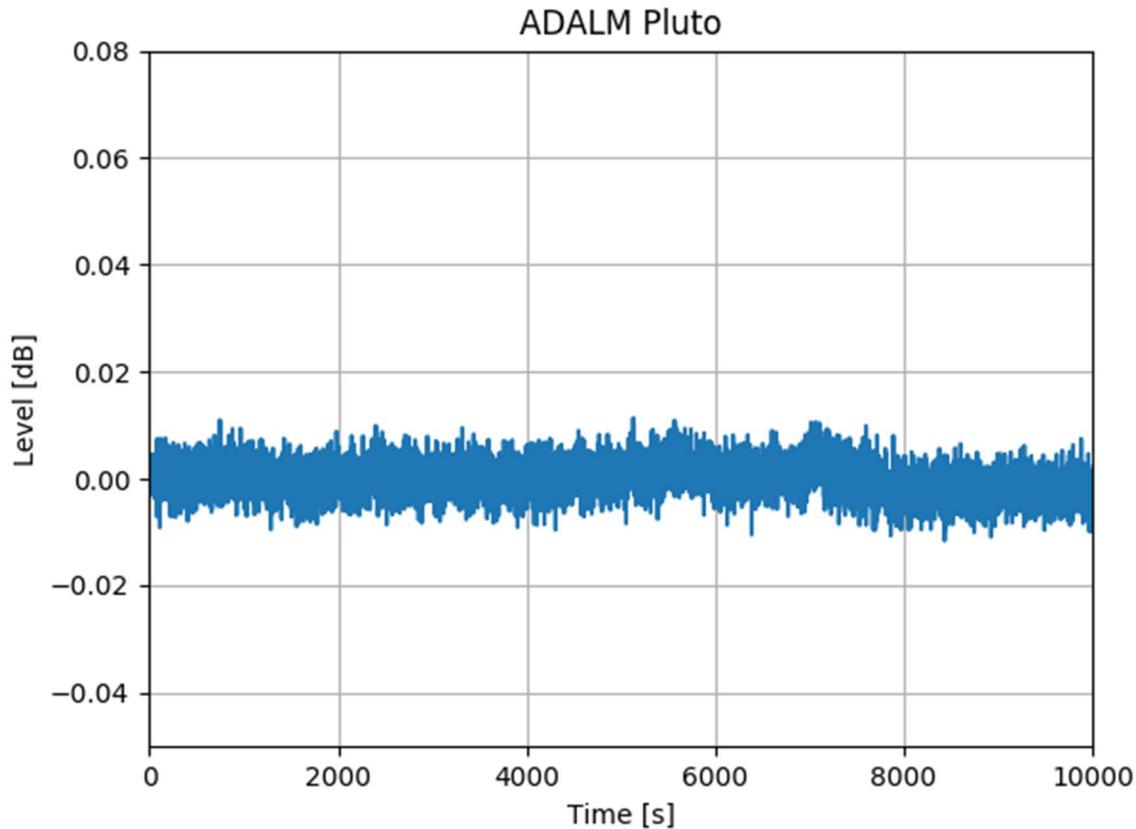


Figure 18: Total power variation, ADALM Pluto

4.8.4. Spurious lines

SDRs can show spurious lines, especially when operated at their sensitivity limit. Typically, these lines are quite narrow and therefore can be told apart from a hydrogen line which is substantially broader. Still they can be a nuisance. We have therefore investigated this by looking at the spectrum at a low level input. Of course, there may be different behaviour from unit to unit. The results are therefore only indicative.

Test method

A low power noise signal (approx. -150 dBm/Hz) was connected to the SDR and the SDR was operated at maximum gain. Spectra were averaged over 5 min and the resulting spectra was recorded. The spectra were post processed by fitting a polynomial to remove the baseline variation. The resulting spectrum is shown.

RTL-SDR

The RTL-SDR shows a number of spurious lines including one directly at the rest frequency of hydrogen, albeit not too strong. This is shown below in fig. 19.

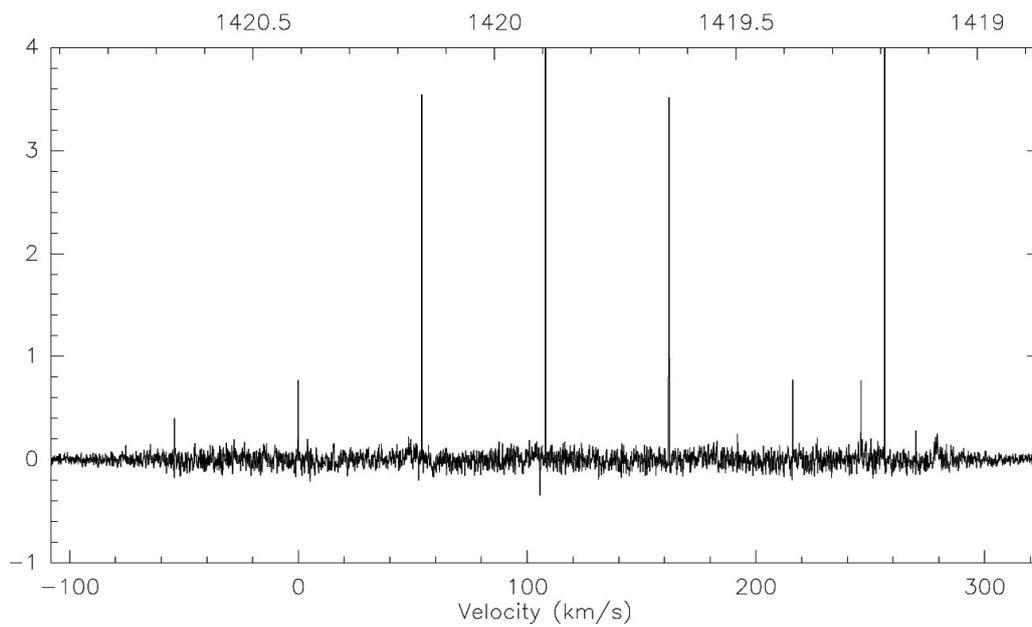


Figure 19: Spurious lines RTL-SDR

RTL-SDR in metal housing

The RTL-SDR which is provided in a metal housing is much better in respect to spurious lines (see fig. 20). This is a bit surprising as the spurious lines in the previous example are not coming from the outside.

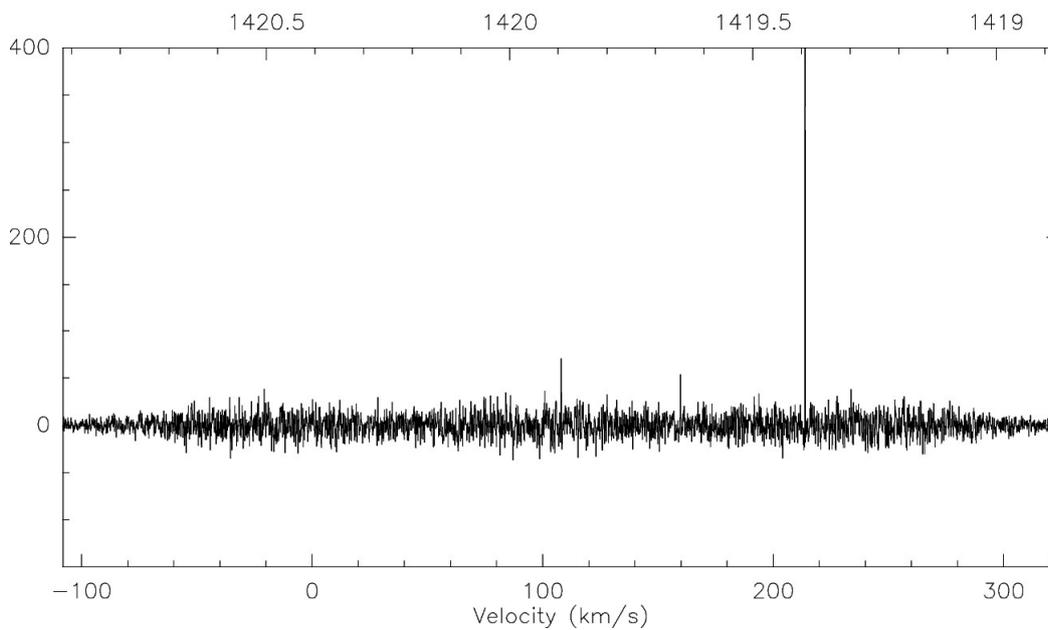


Figure 20: Spurious lines RTL-SDR in metal housing

HackRF

The HackRF is relatively clean with just one spurious line, see fig. 21. The strong signal at the center frequency is typical for SDRs with a zero IF and is unavoidable for this kind of devices. The occurrence of this strong line is also the reason why we choose the center frequency to be different from the rest frequency of the hydrogen.

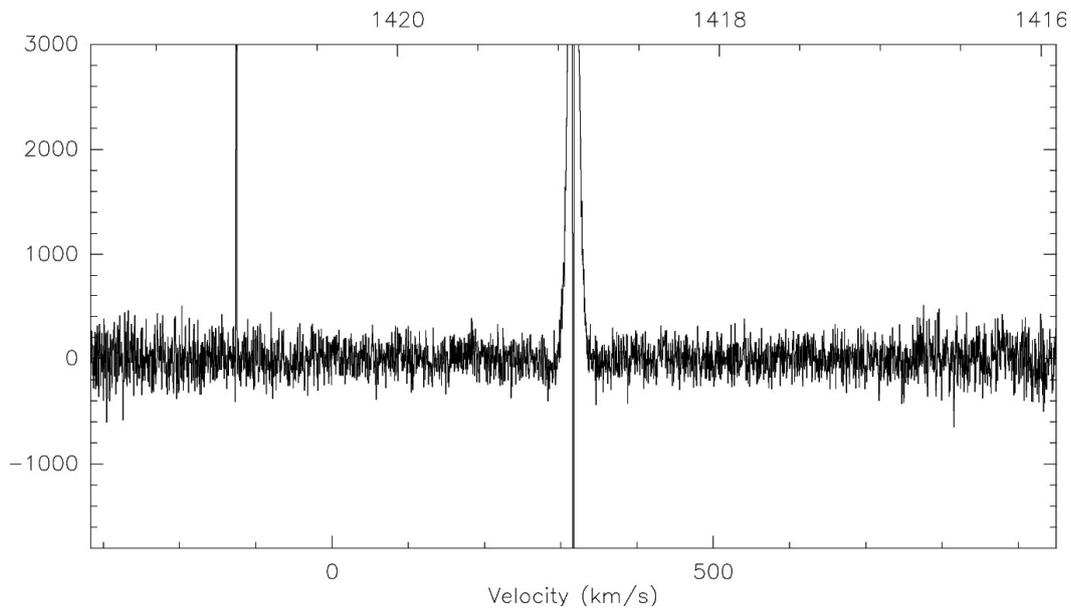


Figure 21: Spurious lines HackRF

HackRF clone

The clone of the HackRF performs not quite as good as the original, see fig. 22. The gain seems to be less as well with the same setting as indicated by the lower noise floor.

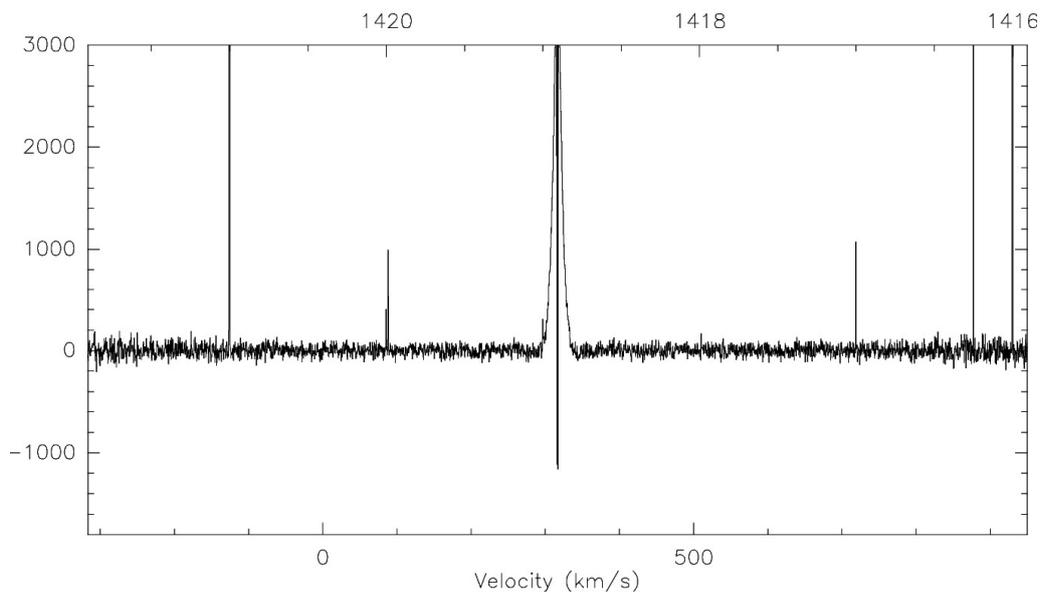


Figure 22: Spurious lines HackRF

ADALM Pluto

With the exception of the center line the ADALM Pluto shows no spurious lines. However, the response has a “wobble” which is not removed by our baseline correction, see fig. 23. This effect can already be seen in the spectral response plots above (Fig. 11, 12 and 13).

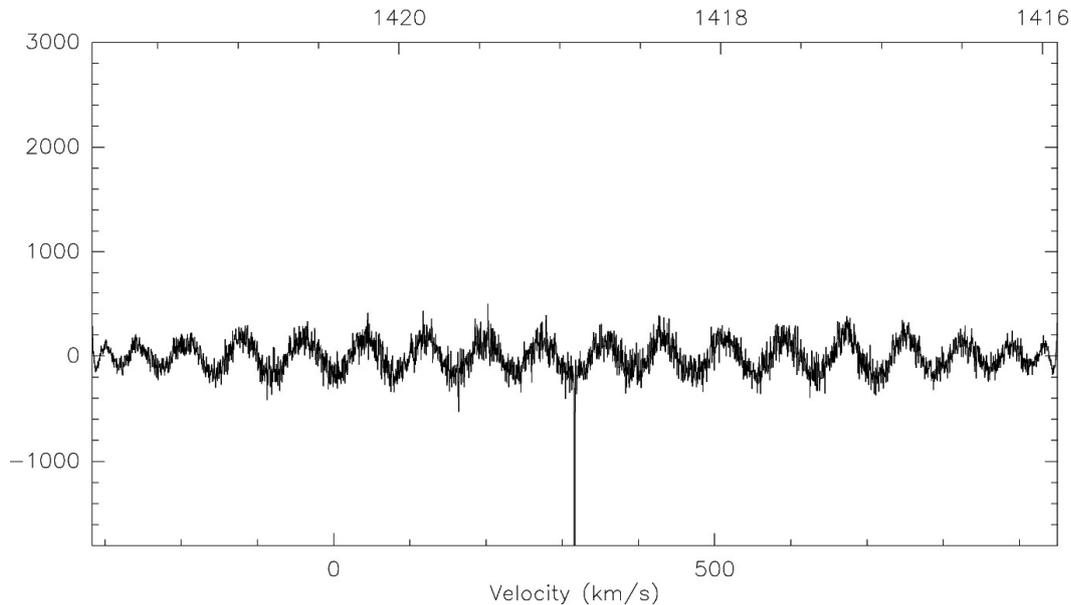


Figure 23: Spurious lines HackRF

Still this baseline variation can be removed by fitting a sinewave function in addition to the polynomial used, but it is further complication.

We have tested two ADALM Plutos to see if this just one device showing this effect. However, both are identical in their behavior.

Conclusions

Obviously among the devices tested there is no ideal candidate. The ADALM Pluto has many things going for it, but the “wobble” in the spectral response is a drawback. Also, the lack of a clock input requires some hardware work if high spectral precision is the goal. However, it seems to be the only option to do calibrated intensity measurements as it has no noticeable thermal run in.

The HackRF has a reasonable spectral response but the thermal run in behavior makes it unsuitable for quantitative measurements unless specific procedures are used to keep it in thermal equilibrium. This means that one would have to have the sampling going on all the time.

Finally, the RTL-SDR is the clear winner if one is shooting for a low budget solution.

5. Backend software

There are quite a few options for software which can be used with the hardware described above. Within the scope of this project, we did not test the various packages. Therefore, only the software tool-chain which we commonly use with our telescopes is described here. The various tests shown above were done using this particular software.

However, we try to give an overview what else is available. There will be oversights and omissions but hopefully we have covered some of the major ones.

5.1. Tool-chain used at Astropileer Stockert

5.1.1. SoapySDR and soapy_power

Our standard tool-chain for SDRs is based on SoapySDR [5] in which controls the SDR and collects the raw data. This software has the benefit that it supports a large number of SDRs. Due to this, a common set of software can be used to work with different SDRs.

The basic concept of SoapySDR is to provide an abstraction layer which isolates the application from the specifics of the various SDRs. These specifics are implemented in “Device Modules” which drive the SDR. The application itself, however, will be isolated and have a common interface to access the data. The site referenced in [5] describes the concept in more detail.

The objective for hydrogen observations is to get a spectrum. Therefore, a Fast Fourier Transform (FFT) process is required. Fortunately, there is a good implementation using the SoapySDR API: soapy_power [6]. By using this in combination with SoapySDR it is very easy to get a spectrum from almost any SDR. It is interesting to note that the default centre frequency of soapy_power is the rest frequency of the hydrogen line. This indicates that the author of this application had radio astronomy in mind when he developed the application.

Finally, the spectral data provided by soapy_power is collected by a site-specific application which also reads data from the telescope control and writes FITS (Flexible Image Transport System) files which is a common format in astronomy. These files include a header which contains all relevant information from the observation.

The OS used for this installation is Linux. However, SoapySDR is available for Windows as well. The Windows variant has not been tested so far.

5.1.2. CLASS

Post processing of the data such as baseline correction and generation of plots is done using CLASS. This is part of the GILDAS package developed by the team at Institut de Radioastronomie Millimétrique (IRAM) in Grenoble, France [7]. This package is a standard among astronomers doing spectral observations in the radio regime.

GILDAS is available for both Linux and Windows.

5.1.3. rtl_power_fftw

Before moving to SoapySDR we have used a software which supports the RTL-SDR only. This software is called rtl_power_fftw [8]. It is an enhanced version of the rtl_power software which is part of the standard rtl-sdr software. The improvement over the older software is that it uses a more efficient FFTW library.

5.2. Other options

As noted, the other options listed below are mentioned for information only. No tests have been done with these packages.

5.2.1. GNU radio based software

GNU radio also offers the advantage of isolating the specifics of an application from the SDR hardware. We are aware of two applications based on GNU radio which support radio astronomy observations.

One is developed by Marcus Leech [9] and provides both spectral and continuum observation capabilities. The other is developed by Glen Langston [10].

5.2.2. CFRAD

For Windows users there is a software program developed by Michiel Klaasen called CFRAD [11] which supports the RTL-SDR. The basic principle is that first the SDR is set and controlled by SDRSharp and then CFRAD takes over to collect the samples and processes them.

According to some forum discussion it seems that this program does not support Windows 10, so earlier Windows OS version have to be used currently.

There are also two more beta versions of the software supporting the Airspy and the HackRF SDRs.

References:

- [1] <https://www.rtl-sdr.com/about-rtl-sdr/>
- [2] <https://greatscottgadgets.com/hackrf/>
- [3] <https://ez.analog.com/university-program/f/q-a/77922/will-it-be-possible-to-feed-in-a-reference-clock-to-the-adalm-pluto>
- [4] <https://www.rtl-sdr.com/roundup-software-defined-radios/>
- [5] <https://github.com/pothosware/SoapySDR/wiki>
- [6] https://github.com/xmikos/soapy_power
- [7] <https://www.iram.fr/IRAMFR/GILDAS/>
- [8] <https://github.com/AD-Vega/rtl-power-fftw>
- [9] https://github.com/ccera-astro/spectro_radiometer
- [10] <https://github.com/glangsto/gr-nsf>
- [11] <http://parac.eu/downloads.htm>



About the Author: Dr. Wolfgang Herrmann is the president of the "Astropeiler Stockert e.V.". This is an all volunteer, non-profit amateur organization taking care of the Stockert radio telescope facility which includes a 25m dish.

He received his PhD in Physics from the University of Bonn. He has spent most of his professional career in the telecommunication industry. At retirement age, he now enjoys learning as much as possible about radio astronomy, doing observations and improving the instruments. Contact the author at messbetrieb@astropeiler.de