

*The RTL SDR V3
(or, Eight Bits On HF)
By D. B. Gain*



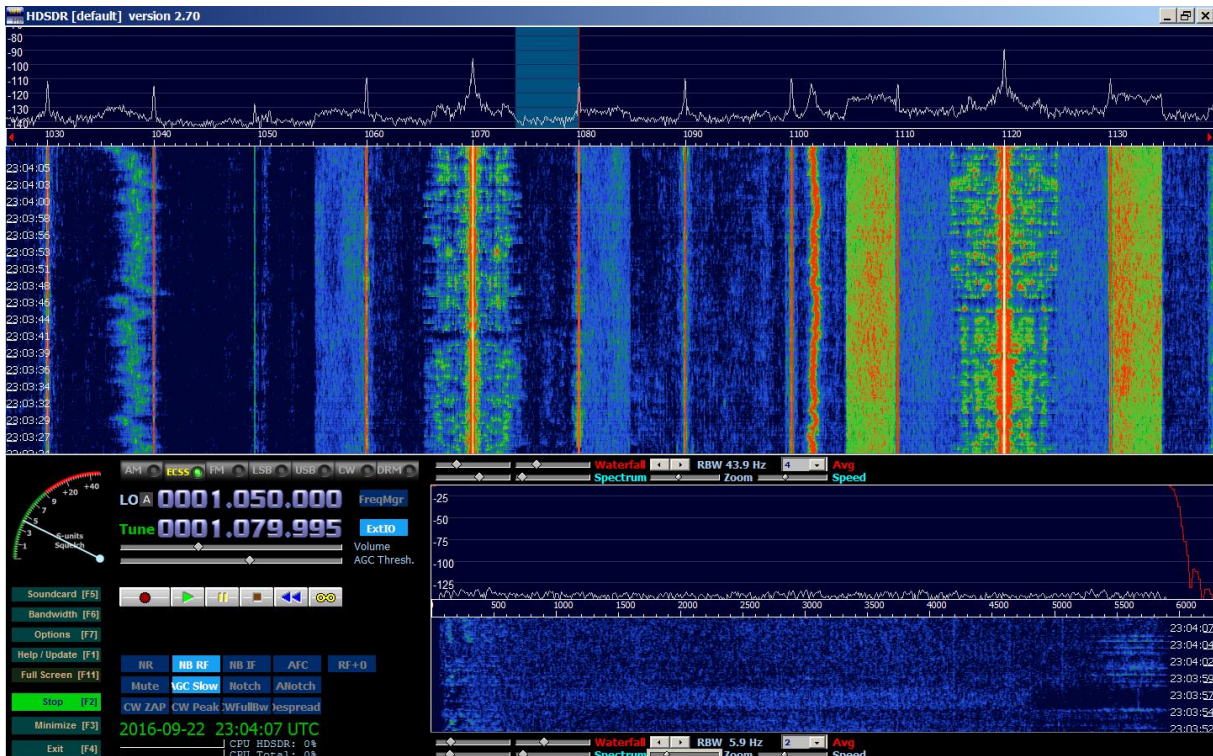
Purpose and Scope;

To see what you get using an \$20 8bit ADC (Analog to Digital Converter chip featuring an ENOB of 7) SDR (Software Defined Radio, a radio that is "defined" by the software employed to control it as to frequency ranges covered and methods of tuning and demodulation) and commonly used often free software such as SDR#, HSDR and other SDR control apps as well as decoding apps for operation in the HF environment. Yes I know this is a tall order for HF work but I'm intrigued by the minimalist (in hardware as well as expense) aspect of an 8bits on HF paradigm.

Radio Hardware;

RTL-SDR Blog on the internet recently touted a new RTL dongle, one with improved performance over run of the mill RTL dongles. The fine folks at RTL SDR Blog were kind enough to send me a sample RTL SDR V3 dongle and antenna kit, this is the dongle used throughout the review.

If you didn't know, the RTL USB dongles are originally manufactured for VHF/UHF digital TV broadcast reception via a pc and were found to be hackable for general radio pursuits such as scanning, trunked radio system tracking, APCO25 and other digital modulation scheme decoding, as well as other radio endeavours in the 25 to 2000MHz range. You plug the dongle into a handy USB port, fire up the control app (after installing a driver and/or dll for the dongle perhaps), and off you go into the vast new world of SDR radio. A world rich in Software Defined Radio and all it entails.



AMBC band displaying several IBOC HD stations

Notice I didn't mention HF in the paragraph above. That's because the RTL dongles aren't designed for HF reception at all. This is where the hack comes in, some fine folks discovered that one could control the various I/Q channel inputs via software, not only to allow coverage from 25 to 2000MHz, but by using the direct sampling input, provide for HF coverage from around 500KHz or a bit lower to 30MHz. Some dongles lacking hardware support for direct input mode will require a hardware modification to allow for direct input HF use, typically the soldering of a single wire from the antenna connector to one of the ADC input pins. This HF capability is what made me jump at the chance to test one of these RTL's as I'm an insolvent HF fanatic.

Normally one wants more than an 8bit ADC to get sufficient gain and dynamic range on HF (or anywhere else), somewhere along the lines of 14bits and up will do very well, 24bits would get you a theoretical 144dB dynamic range (dynamic range is the ability to copy weak signals undistorted in the presence of much stronger nearby and not so nearby signals, very similar to being able to hear a whisper spoken to you while standing next to a jet engine at full afterburner), but an inexpensive 24bit chip that can handle HF has yet to enter the hobbyist market, so the 8bit chips are the price leader. These higher bit ADC chips exist in the marketplace, but typically in the form of military and government surveillance receivers that come with high prices. Oh by the way, there's a 24bit or better ADC sitting in the CD/DVD player of your PC.

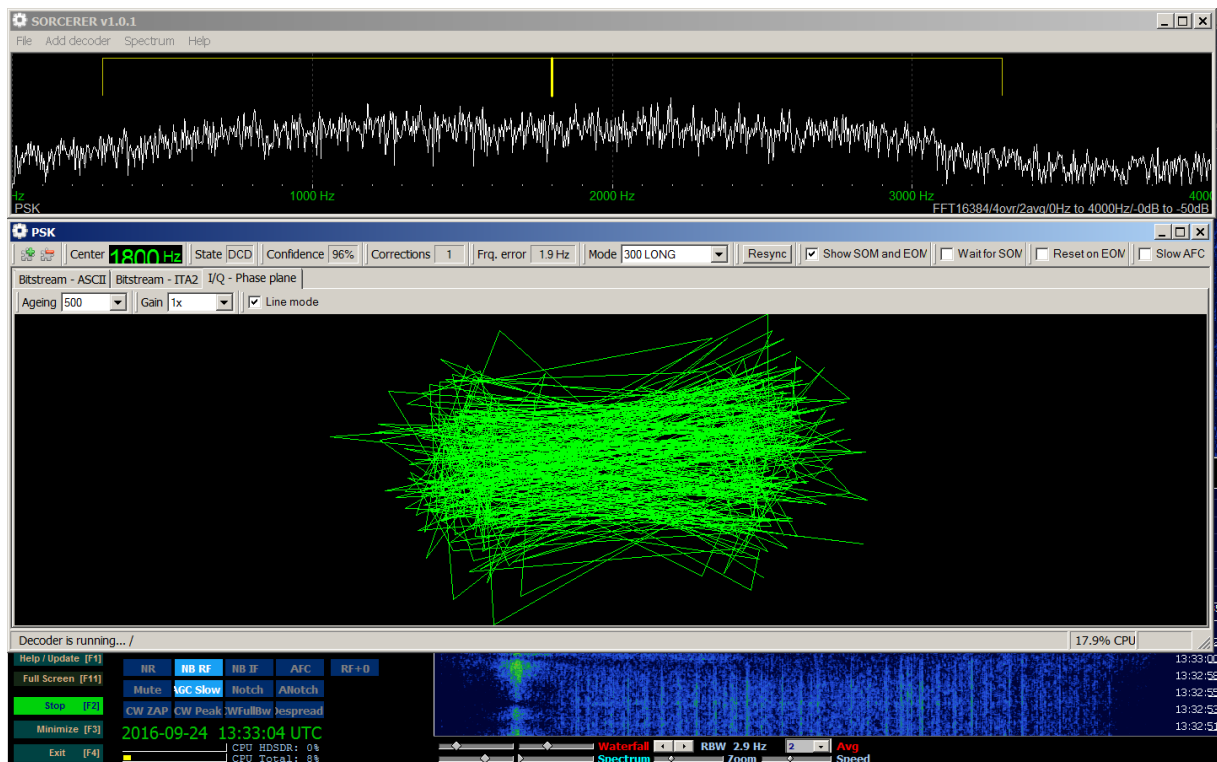
Some of the consumer grade SDR radios aimed at the HF market are of the 12 to 16bit range and these do nicely in the crazy HF environment, especially when one gets to 14bits and above.

The jump from 8bits to 12bits is on the order of 6 times the price of the 8bit V3, but the jump from 12 to 14 bits is around three times the price or more of the 12bit SDRs. The way I look at it, the jump from 8 to 12bits gets you about 15 to 20dB dynamic range increase, which is very noticeable and useful and is a game changer compared with 8bits. The performance difference between a 12bit ADC and a 14bit ADC is closer than the performance difference between an 8bit ADC and a 12bit ADC. I guess what I'm really waiting for is the time when future 32bit chips are as cheap as the 8bit ones of today, however long that may take.

The physical properties of said dongle are about a thumb's length and half thickness of a thumb, with a USB connector on one end and an antenna port on the other.

The dongle comes with a mag mount antenna base with attached coax and SMA connectors and two telescopic rod antennas, one of impressive stature. The V3 can also be purchased by itself for a bit less money.

The new RTL dongle sports a shiny aluminium case to keep signals in or out of the SDR as the case may be, said case also acting as a nice heat sink, and has a measure of noise and interference reduction in the circuit board itself compared to previous RTL dongles, especially with regard to the USB line and switching mode power supply, very welcome advances in the dongle art. Typically these dongles have a plastic case and are thusly prone to allowing interfering signals from the pc and other in home devices as well as the local RF environment to spoil the signals before they enter the pc through the dongle and SDR app. Another advance is the presence of a TCXO, or Temperature Controlled Crystal Oscillator.



Co-channel FAX on S4285 Signal

The TCXO reduces the oscillator drift that arises normally from internal component temperature variations due to operation and environmental factors by keeping the crystal that the reference oscillator is derived from at a more constant temperature, negating drift after a few minutes of warmup. I can vouch for the RTL V3 TCXO's abilities to keep you on frequency with little or no drift, at least on HF. The dongle also sports a SMA antenna connector, one will have to obtain an SMA to whatever coaxial connector one uses. I obtained a SMA to UHF adapter, lightweight so as to keep stress off the antenna connector in the RTL V3.

PC Hardware;

I used a PC with a 4 core Intel CPU, Intel mobo, 4gb ram, and 32bit Win7 OS. The onboard HIDEF sound system handles the sound routing via the stereo mixer. The particular sound system has a max sampling rate of 48KHz but that works out just fine for HF decoding apps and general HF listening.

As the max audio bandwidth I use for HF reception (and feed to decoder apps) is 10KHz or less, 48KHz is more than enough to satisfy the Nyquist criterion. Harry Nyquist said you have to sample a given signal of interest at least twice its highest frequency to reproduce the sample digitally without loss or distortion. In regards to sampling analog things into the digital world we should listen to Harry Nyquist, he knows what he's talking about. A soundcard typically takes analog signals presented to it, runs them into an ADC (analog to digital converter chip), manipulates them digitally and then sends them off to speakers or an application, via a DAC which is the reverse of a ADC, a digital to analog converter.

Antenna System Hardware;

The antenna for HF used in this review was a off center fed dipole up around 30ft, RG6 cable TV coax feedline, and a Barker and Williamson 30MHz lowpass filter. Ferrite beads were placed on the antenna end and connector end of the coax to reduce noises traveling upon the coax outer shield.

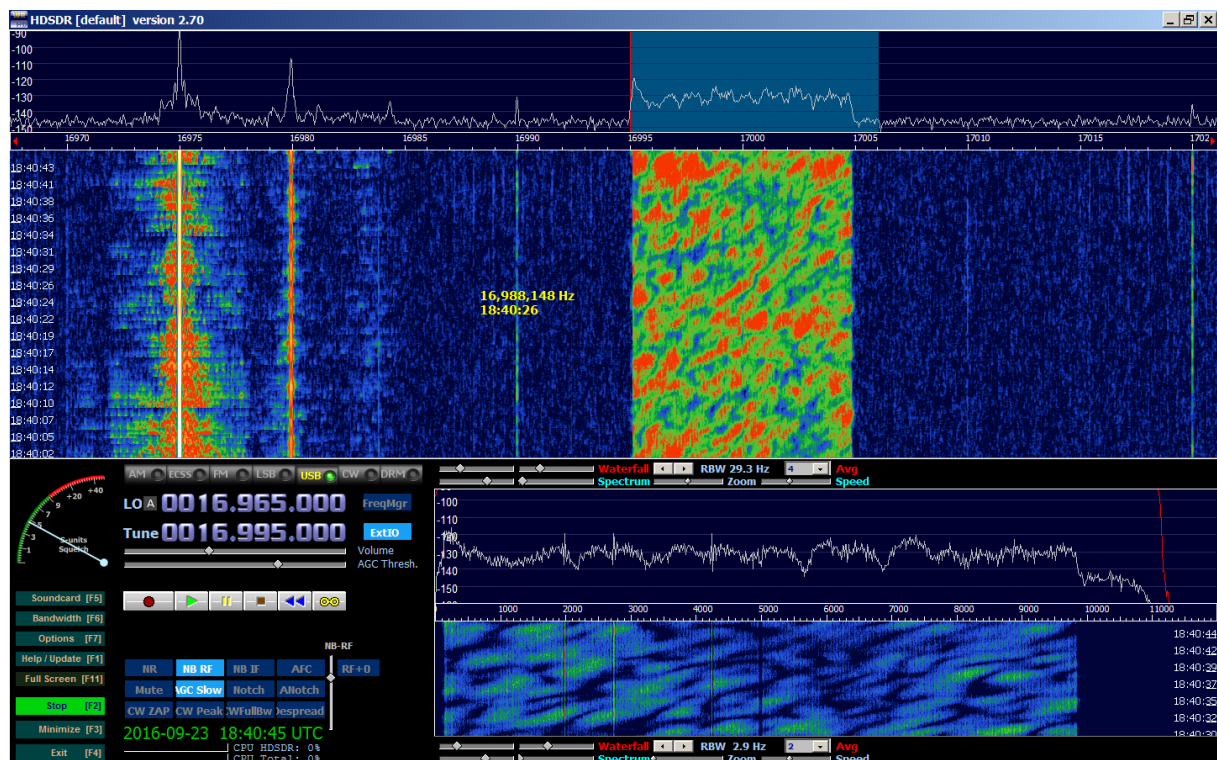
Off-center fed dipoles have an interesting effect on SWR; they often have low SWR every other MHz through the spectrum. Sometimes this low SWR happily lands on a band of interest, but no matter, even if the antenna has higher than desired SWR on the frequency of interest there should be enough signal energy available to listen to the signal satisfactorily.

The Barker and Williamson 30MHz lowpass filter also acts as a reactance moderator, somewhat reducing the wildly swinging SWR as compared to when it is absent from the antenna system. It also cuts off signals above 30MHz, which protects sensitive RTL dongle innards from unwanted radio energies. Energies such as literal megawatts of FM and TV broadcast stations, pager systems, cellular systems, and business/government communications that populate the V/UHF bands. These signals would all impinge upon the hapless RTL dongle, greatly hindering HF performance if not for some lowpass filtering.

The masses of AMBC stations in the 530 to 1700KHz Medium Wave range also can perturb a RTL dongle if too much MW energy is placed at the input to the dongle. Here a highpass filter or attenuator is in order, or a preselector.

I find that the use of a preselector or transmatch can reduce images and IMD, but even at their best the dongle still has issues with faint images and spurs, which is the nature of an 8 bit device. You learn to live with it and tune around the worst if you can.

The antenna feed also sports a Polyphaser emp rated coaxial transient arrester. This two port device, grounded via its casing, presents the radio side of the transient suppressor with capacitive coupling of the coax center conductor to the rest of the antenna system, the antenna side sees a high inductance to ground at the transient suppressor, which bleeds off static and yet passes rf. It also has a gas discharge arrestor and is overall designed to handle emp events. This device alone cost several times what the RTL SDR V3 dongle cost, much of the antenna system and PC hardware also cost far more than the V3 dongle.



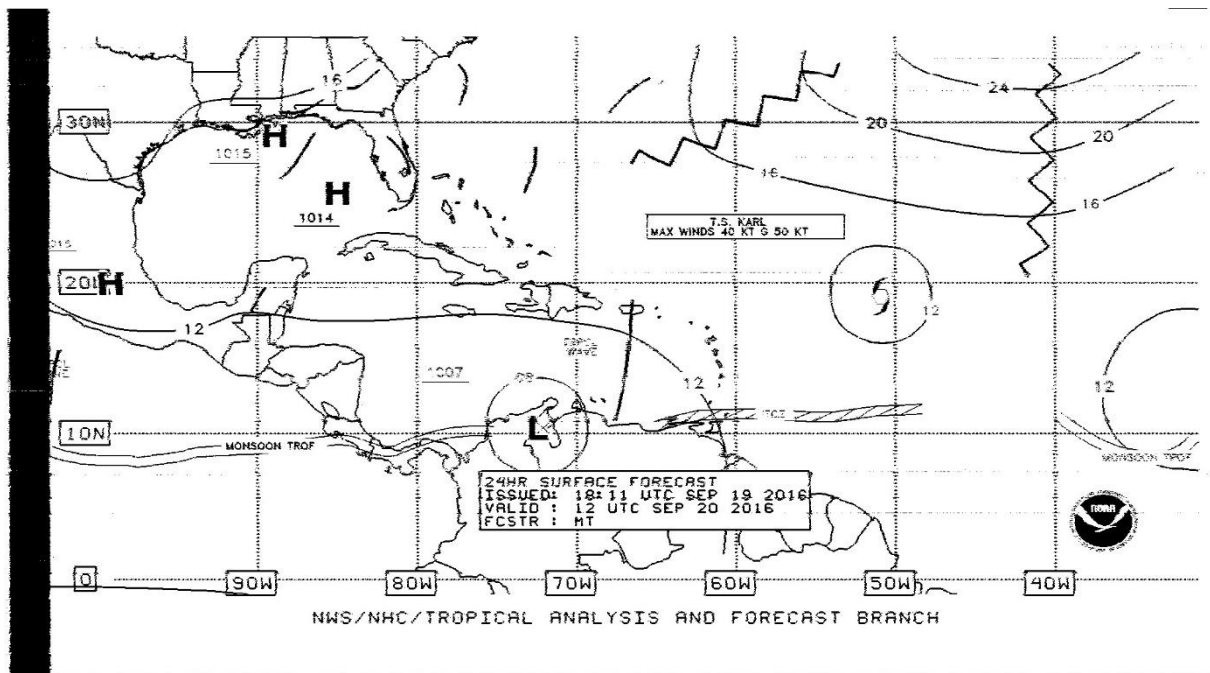
Crazy Wideband HF Thing

Preselection;

In this review a Grove TUN3 Minituner Preselector (of the bandpass variety) was employed, as well as an MFJ manual HF Transmatch (of the T circuit highpass variety).

The TUN3 is no longer manufactured but can be found on the used market. A rather small black box with two tuning knobs, one for band, the other for fine tuning of the resonating/matching capacitor, it tunes from VLF to 30MHz and slightly above. It's intended for reducing images in single conversion general coverage and AMBC receivers and does a pretty fair job of that in my estimation. It peaks a desired signal nicely even on HF but quite spectacularly on AMBC and below while also reducing the out of band rf. You can find schematics online for building your own preselector, from simple to elaborate as you desire.

The MFJ Transmatch used in this review is also one no longer made, it's a simple and compact aluminum box with three controls; one for the input side capacitor, the next for the inductance which selects the band of use, and the last for the output variable cap. The back of the tuner has input and output UHF sockets. As it's a T circuit it's highpass in nature, meaning it cuts off RF energies below the frequency it's tuned to. That's a side benefit as its design goal was to match an high reactance antenna system to the 50ohm antenna port found on HAM rigs. The T circuit offers about the most tuning range as far as input reactance or Z from the antenna system. It's not a preselector but it can reduce the RF energies present in the antenna system below the frequency it's tuned to, and that is a benefit to many radios, not just the RTL SDR. You can find schematics online for building your own Transmatch, from simple to elaborate as you desire.



Fax received with the SDR

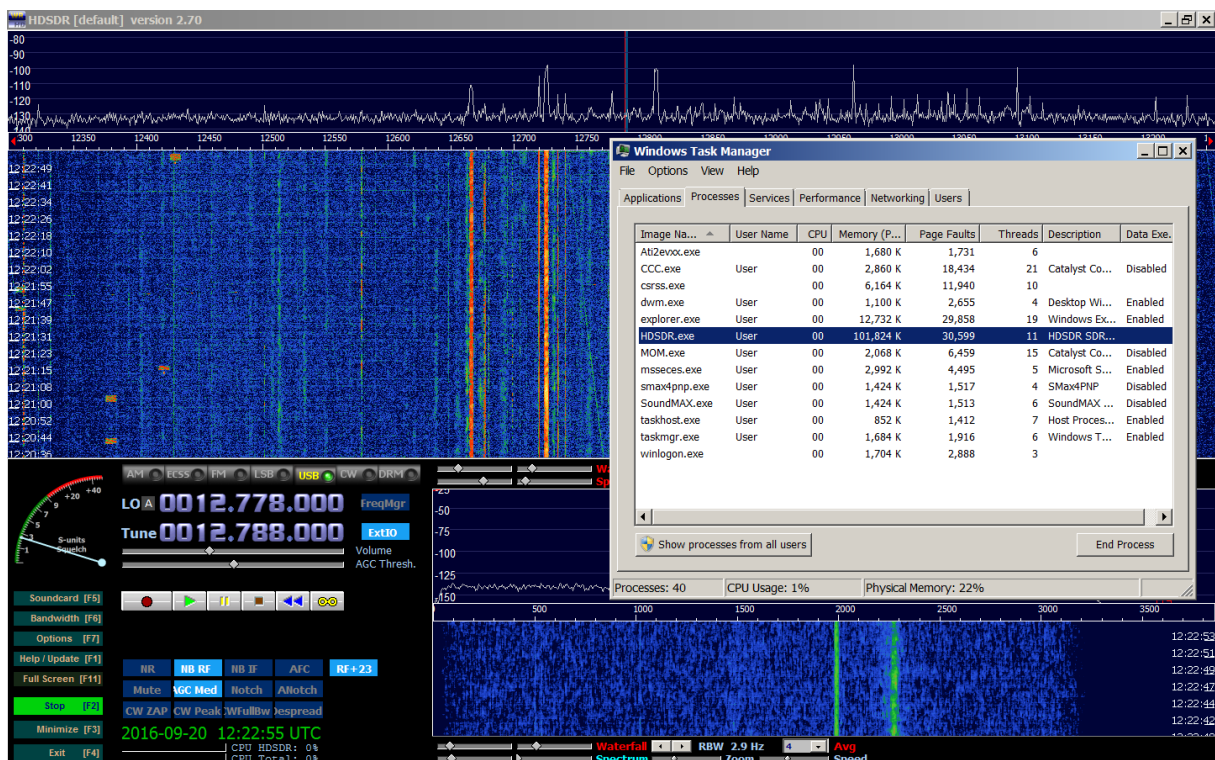
SDR Control Applications;

Employed were HSDR, SDR#, SDRuno, and other SDR control apps for this review. These apps you can download and use for free. A simple driver installation procedure is sometimes called upon by the apps, Zadig, also free to dl and use, is one of these driver installer packages, it installs the driver for the OS to recognise and communicate with the RTL dongle.

Out of all the SDR control apps I've used so far I like HSDR best, it has amazingly low CPU usage and acceptable Memory footprint as well as thread count. Threads are the portions of an app that actually gets run on the CPU. Many programs have only one thread, which is run in a loop on the CPU and can only be run on one CPU core at a time. Threaded apps can improve performance by spinning off portions of a process that can wait - freeing up other portions of the app to do a job, or improve performance by having multiple CPU cores working on the app at the same time. Of course to take advantage of a multithreaded application one needs a system with more than one CPU core. Today this is almost a given as even phones and embedded systems offer multiple cores. This was not so just ten years ago, when the majority of pc's around the world had a single CPU.

Anyway, I notice SDR#, SDRuno, SDRConsole and CubicSDR have low Memory footprints in some cases but are thread heavy (as in 20 plus threads), are CPU hogs, and not to my liking as far as glitchyness or stability on my system in the case of SDR# and CubicSDR, but SDRuno was rock stable. HSDR wins here, even though it may take up twice as much ram as the other apps doing the same job, HSDR has never crashed or displayed glitchyness in use at my station. For example of a CPU hog, SDR# used around 9 percent CPU time doing the same thing HSDR did at 0 percent. If you have high performance CPUs or multiple CPUs what's the deal about not liking the SDR app to use CPU time?

Well, my answer to that is I consider the SDR app to be the OS of the SDR, and as such want as little CPU and Mem footprint as possible so the system is devoted to catering to the needs of apps used in conjunction with the SDR control app such as Sigmira, Sorcerer, MultiPSK and others. Don't get me wrong, I like a fancy waterfall and spectrum display too but when you can get similar information from HSDR at a fraction of the CPU time used by many other SDR control apps, HSDR seems a no brainer. By the way, Sigmira currently doesn't do the direct sample mode needed to place the V3 on HF without an upconverter.



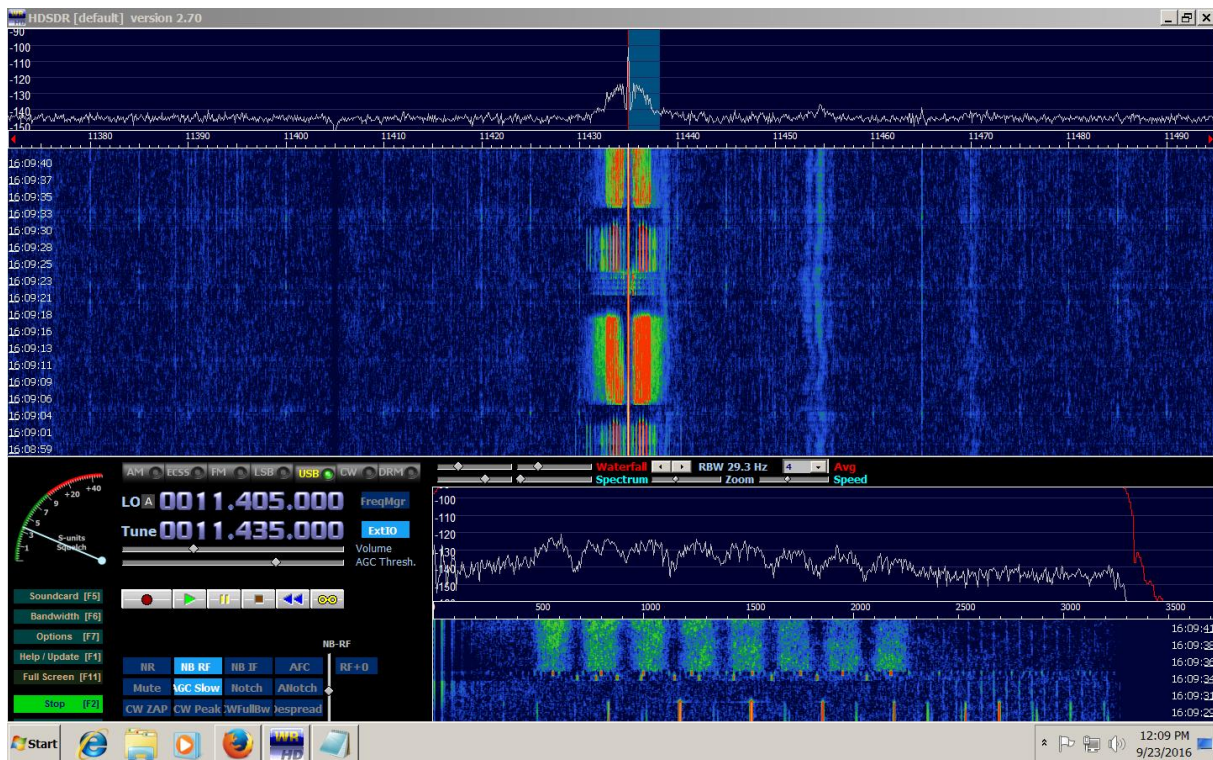
HSDR CPU and MEM Usage

I'm impressed with SDRuno, which is a version or fork of Studio 1 SDR app, it's very stable and has wonderful agc action and audio reproduction. That said, it has tiny fonts due the lack of font resizing, and cryptic controls that at this time cannot be enlarged, also the control panels are not docked to the main screen but are tiny windows to themselves. It also seems to use a bit of CPU time. It also refuses to save any settings at least in my case so you must renegotiate all the controls for use with the V3 with each use. I hold out hopes for this app in regards usability in the future. One thing to keep in mind with SDRuno is keeping the sample rate to less than 1msps or the app will not operate the V3 dongle.

I'm likewise still testing SDRConsole, it's a interesting and very stable SDR app but so far I can't find out how to direct the app to use the Q branch of the V3 dongle that enables use on HF.

Feeding Audio To Decoder Apps;

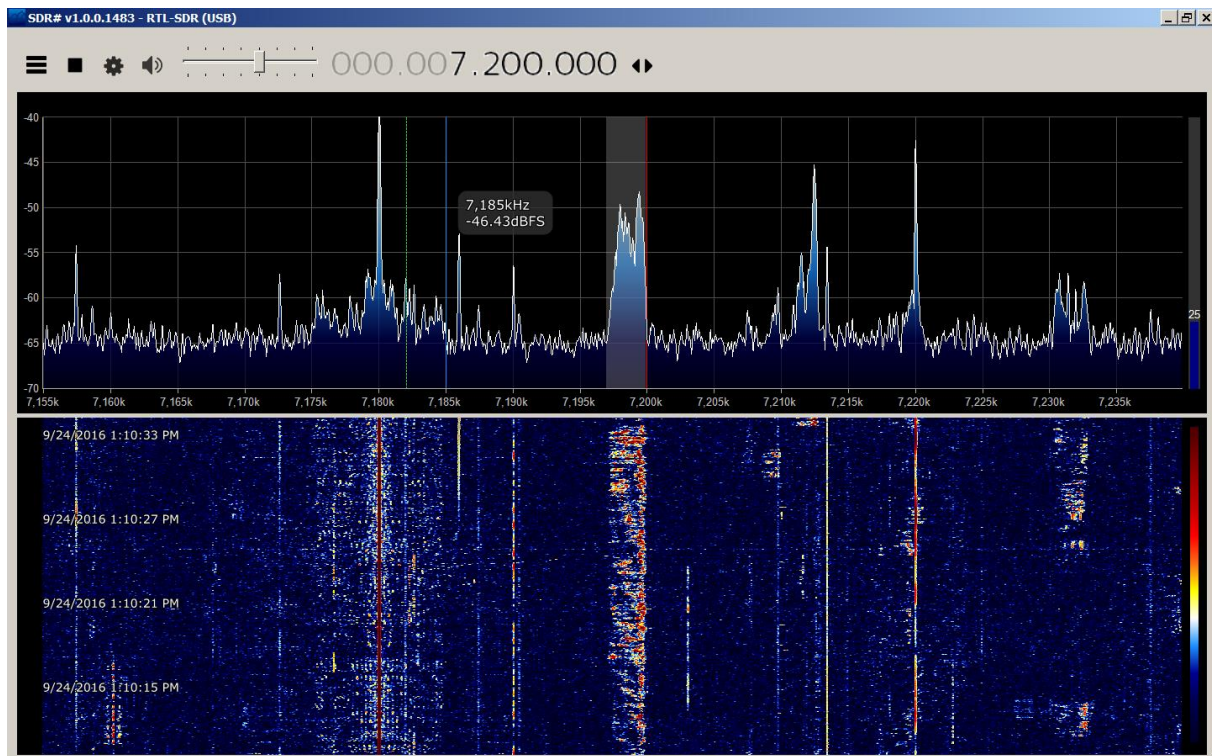
The Win7 Stereo Mixer api piped audio to every decoding app used. If the mixer isn't seen in your Windows install you can normally force it to appear and enable it, google for details. Some decoding enthusiasts prefer to use VAC or other means to pipe audio around and that's an alternative to the mixer.



HM01 Spanish Lady and RDFT protocol

USB Sample Rates;

One thing noted regarding all apps is the higher the sample rate of the SDR dongle, usually the better the imd and spurious resistance. By that I mean if you hear some imd creeping in, say a stray carrier from AMBC or SWBC that shouldn't be there on your frequency of interest, increase the sampling rate and that spur may drop into the noise where it should be. Only increase the rate to what reduces the images and spurs, as more samples just increases USB traffic and CPU time – there is a definite point of diminishing returns with dongle sample rates. Increased samples may up the CPU usage a bit, and may also increase the heat produced by the dongle but this should be tolerable. Speaking of heat, the dongle should be located where airflow can get around it.



HAM radio hijinks on 40m

General RTL SDR V3 HF/Direct Input Mode Best Practices;

First off, disable RTL AGC in any app that provides for it. This defeats the internal AGC of the dongle and relies instead on a manual RF gain setting, usually around 20dB of gain or so is in order for best sensitivity and least imd and images.

You can disable Tuner AGC as well since the tuner is bypassed when in Direct Sample mode. You may have to adjust the gain (usually via a slider) on a band per band case, but the adjustment will likely be slight.

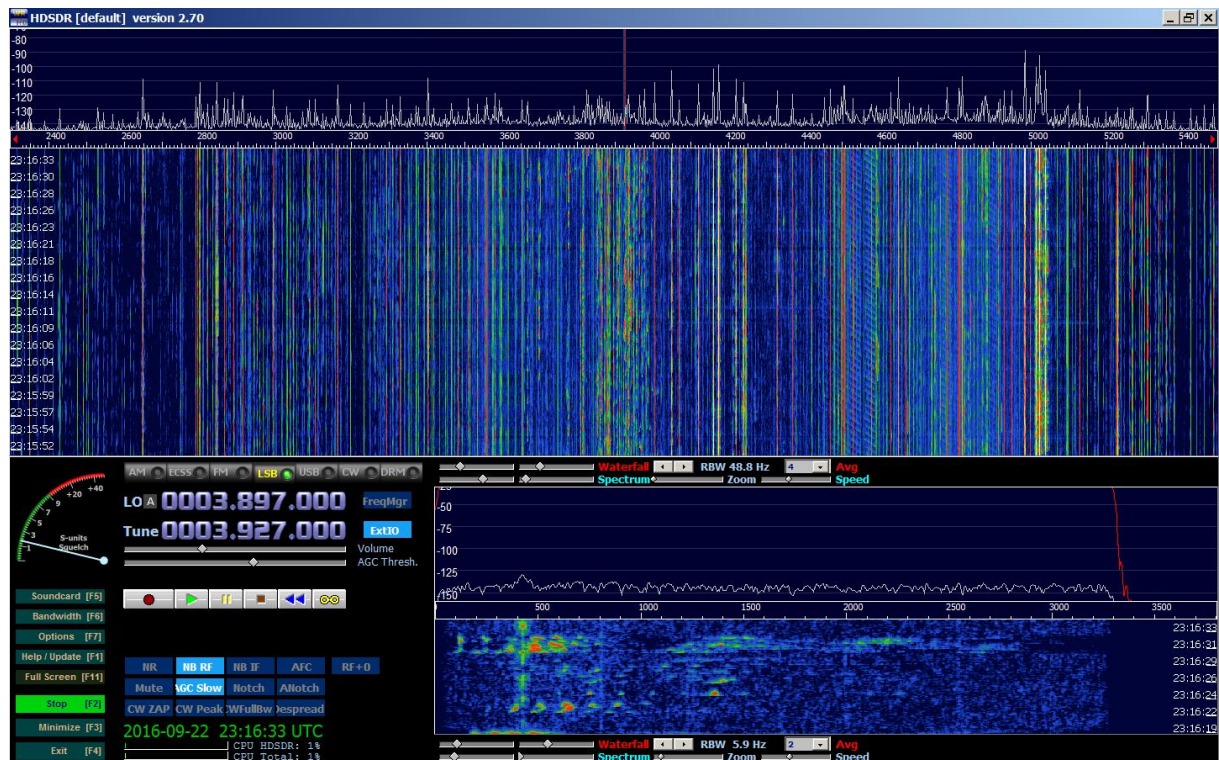
Secondly, use no more USB sample rate higher than results in least IMD and Images, try .96MSPS to 1.44MSPS. This will reduce USB bus traffic as well as CPU time and heat production.

Thirdly, the larger the antenna system, the greater the gain - usually. It doesn't take too much RF to overload the V3 dongle, so a 20ft piece of wire will do better than say a 430ft wire loop atop some phone poles. Use an attenuator if you have one. Remember the issue with AMBC swamping where AM stations pop up in various parts of the HF spectrum and use a preselector and/or attenuator if you can. Shortwave broadcast stations can also create spurs in the V3.

Some radio parts houses carry a variable attenuator meant for cable TV or VCR player use that can be employed at HF with the use of some F to UHF or whatever connector your antenna system employs adaptor, this can be installed in the antenna system and adjusted to result in least usable signal getting to the V3, which assures best dynamic range. Then one would adjust the FFT Spectrum gain in your SDR control app of choice to best level on a quiet band, say 14MHz. This will ensure you don't have to mess with adjusting the gain on lower frequency bands just to keep the band noise baseline above the bottom of the FFT window

Fourthly, try different USB ports on your PC with the dongle so as to find the one with least interference. Some USB ports are quieter in regard to system noises than others. These will typically be seen on the waterfall display and are present without an antenna being connected to the dongle.

And lastly, learn to deal with imd and spurs as they will be common when using the RTL SDR V3 on HF, it's just the nature of the beast. Not the V3's fault alone, pretty much every 8bit ADC is going to react the same way to a diet of HF signals. However, even with the imd and spurs, I found plenty of signals in the clear that were worth listening to.



HSDR and RTL SDR V3 at 3.2 Msps showing 3.2 MHz

SDR Control Application Best Practices;

On any given SDR control app there will be certain settings that have immediate and dramatic impact upon CPU usage. The goal here is to obtain cleanest audio and the desired information such as waterfalls, spectrum displays, s-meter indicators and the like at least cost in CPU time.

In the waterfall controls, set the waterfall speed to the slowest you can tolerate and enable averaging of the waterfall if it's not enabled, this will reveal hidden details about a given signal and as a bonus reduces CPU and Video Card usage.

Adjust the RBW to a coarser setting on the spectrum display for the RF Spectrum of the app to reduce CPU and Video Card usage. Try a setting one step finer than makes stair stepping in the spectrum display apparent. This should give you just the right amount of detail on signals with the least in CPU and Video Card time. Do the same for the Audio spectrum display, but this being a much smaller area on the screen can be set to its highest resolution without much if any impact upon CPU and Video Card use due to the fraction of spectrum being processed, i.e. 5KHz in the case of a Audio spectrum display versus perhaps a MHz or more for the RF spectrum display.

On the I/Q or Audio input settings to the app, if present, use sensible rates, don't force rates higher than your sound card allows in the case of sound card input, and remember that a lot of decoder apps take the 192KHz, 96KHz, or even 48KHz sound card sample and downconvert it to something on the order of 11KHz or even less to perform processing. For example, in the case of HSDR an output of 48KHz was used so that it matched the rate I selected in the sound card property sheet so no conversion had to be made between the rates of sound card and SDR api, the input is the USB sample rate.

Buffer Size can also have a dramatic effect on CPU use, even as little as one step above default buffering can bring the CPU usage up noticeably, so test this control wisely. If the app runs well at default buffer size, leave it or even reduce it till performance is degraded, then increase it one step to restore performance at the least in buffer size.

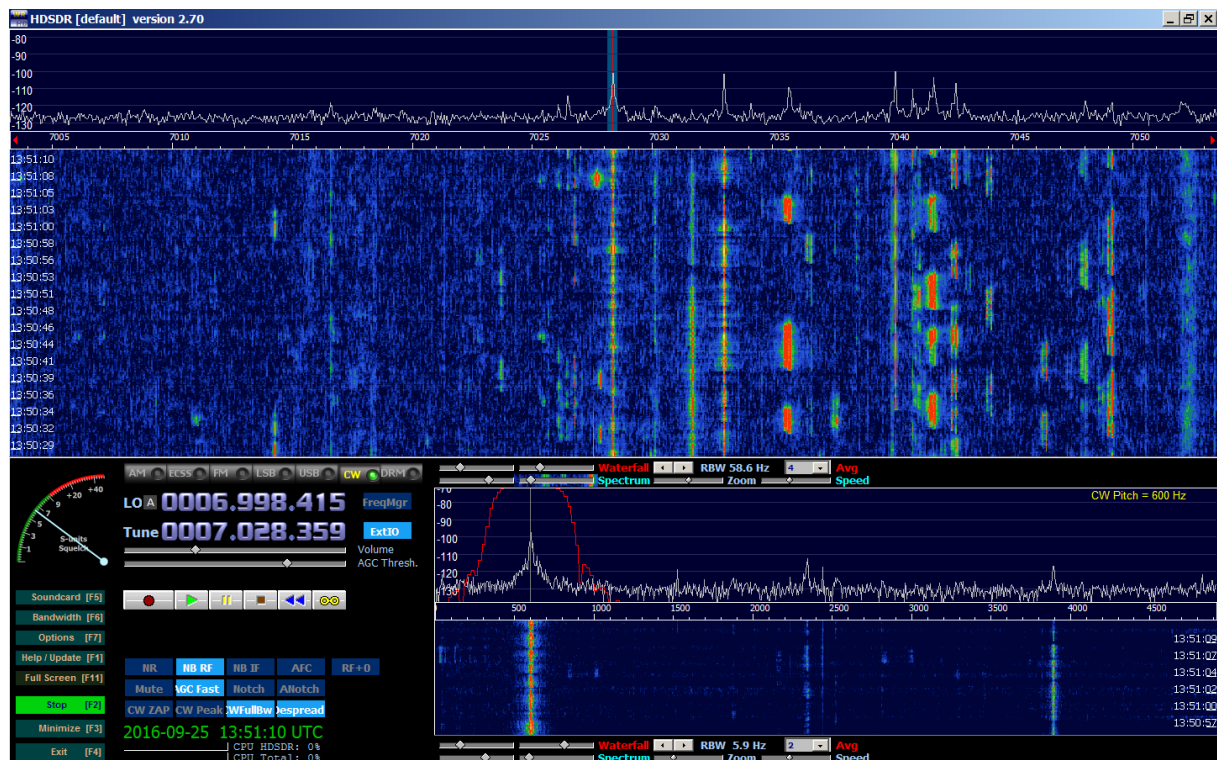
If an image or spur is on your frequency of interest, try adjusting the USB sample rate one way or the other to move the spur off your frequency, this often helps.

If the SDR app has Noise Blanking options for Baseband and IF, use them. Adjust them for least distortion of signals, they will make for better audio out. Imagine HF without the pops and static!

A DSP Noise Reduction option may be present, these can reduce Decoder app effectiveness when enabled so keep that in mind when using Decoder apps.

RTL SDR V3 Pros; a tiny SDR package that tunes MF to 2000MHz or so and has a TCXO, can't be beat for the price, and free software to run the thing is all over the net.

RTL SDR V3 Cons; Has no suboctave bandpass input filters thusly resulting in images and imd and spurs due to the dynamic range of 8bit ADCs, as well as internal mixing of unwanted and wanted signals from phase noise of the reference oscillator. Many of the apps used to control the RTL SDR dongles need additional drivers, dlls, and etc to function, some of which is a pain to deal with but can be overcome for the most part following the many guides and howto's that can be found online. If only it had a few more bits in the ADC. I'm wanting to try a 12bit machine now, they're a few times more spendy than the 8bit SDR rigs but I've heard they do better at HF.

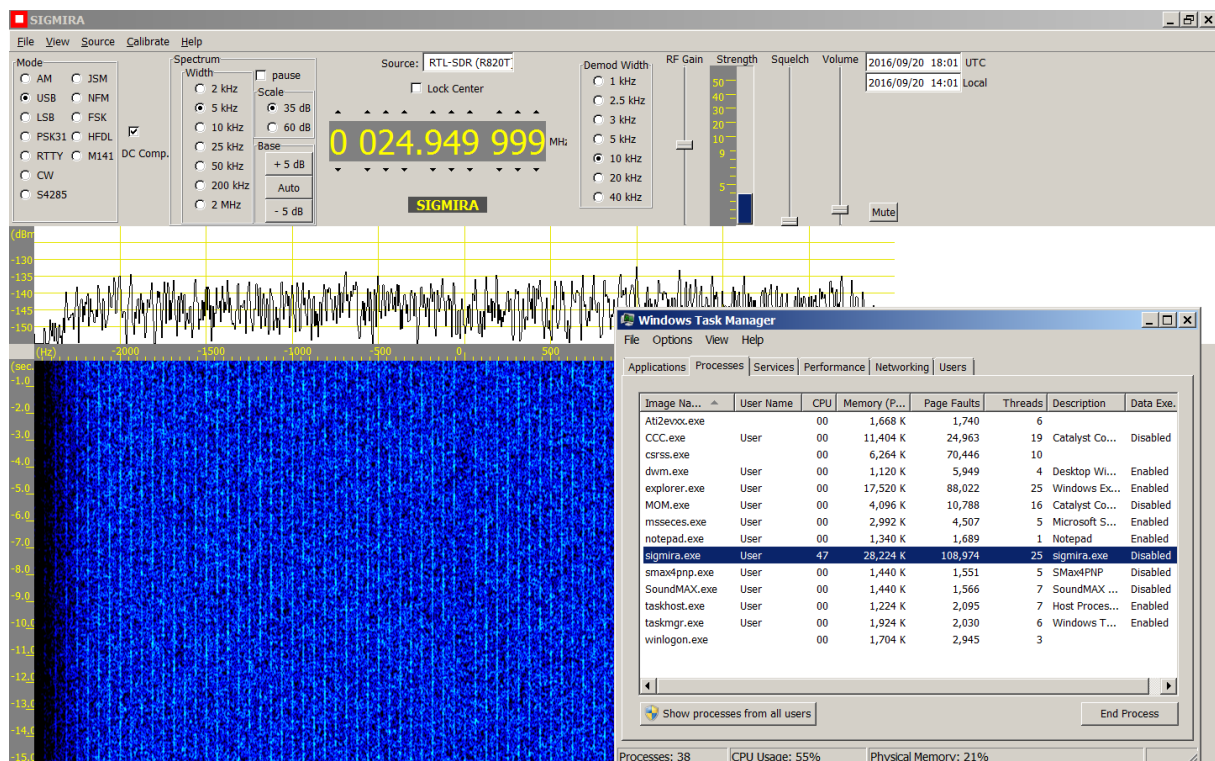


HSDR and RTL V3 tuned to 40m CW signal

In Summary;

I found the RTL SDR V3 to be a fun bit of kit well worth the money, and taking the dynamic range into consideration have done a fair bit of successful digital mode decoding with several different decoding apps as depicted by the multitude of screenshots.

I've decoded HF FAX (and never had to adjust the clock correction in the FAX decoder to get a perfectly aligned image), STANAG4285, ARINC 635 (HF ACARS), 188-141ALE, FSK RTTY, and listened to a multitude of HAM and utility SSB communications as well as SWBC and AMBC stations, all with very nice audio. I've listened to plenty of SSB and CW stations with the V3 and it beats most portables I've used for HF listening by a long shot, especially with regards to audio quality, filtering, demodulation, etc etc. If you're a knob twiddler and don't mind fidgeting around with preselectors and transmatches and gain levels and myriad checkboxes in SDR apps, the RTL SDR V3 just might be for you.



Sigmira CPU and Mem Usage

Resources;

Get you a RTL SDR

<http://www.rtl-SDR.com/new-rtl-SDR-blog-units-now-available-in-store-hf-via-direct-sampling-software-switchable-bias-tee-less-noisespurs/>

RTL SDR blog

<http://www.rtl-SDR.com/>

RTL Setup Guide

<http://www.rtl-SDR.com/rtl-SDR-quick-start-guide/>

The Hobbyists Guide to RTL SDR

<http://www.qsl.net/yo4tnv/docs/The%20Hobbyists%20Guide%20To%20RTL-SDR%20-%20Carl%20Laufer.pdf>

RTL wares

<http://www.rtl-SDR.com/big-list-rtl-SDR-supported-software/>

How tough is the RTL SDR V3?

<http://rtISDRblog.rtlSDRblog.netdna-cdn.com/wp-content/uploads/2016/09/Dotcom-dongle-meets-a-bus.jpg>

Videos of the V3 in use on HF;

<http://www.rtl-SDR.com/more-videos-showing-hf-reception-on-the-rtl-SDR-v3-dongle/>

Build you a preselector

<http://yu1m.qrpradio.com/BP%20FILTER%20HF-YU1LM.pdf>

More preselectors

<http://www.bobsamerica.com/swl.html>

Phase noise and why it matters to you

<http://www.sm5bsz.com/osc/osc-design.htm>

HDSDR

<http://www.hdSDR.de/>

HDSDR RTL setup

http://hdSDR.de/RTLSDR_with_HDSDR.pdf

HDSDR setup tweaks

<https://sites.google.com/site/g4zfqradio/installing-and-using-hdSDR>

SDR#

<http://airspy.com/download/>

SDR# and RTL setup

<http://www.wirelesshack.org/setting-up-a-rtl-SDR-dongle-and-SDR-for-the-first-time.html>

SDRuno

<http://www.SDRplay.com/windl2.php>

SDRuno RTL setup

<http://www.rtl-SDR.com/SDR4everyone-tutorial-on-using-the-rtl-SDR-with-SDRuno/>

SDRConsole

<http://SDR-radio.com/>

SDRConsole RTL setup

<http://m3ghe.blogspot.com/p/adding-support-for-rtl-SDR-USB-dongles.html>

Sigmira SDR app/decoder

<http://www.saharlow.com/technology/sigmira/>

Sigmira RTL setup

<http://www.rtl-SDR.com/sigimira-decoder-now-supports-the-rtl-SDR-directly/>

MultiPSK decoder

http://f6cte.free.fr/index_anglais.htm

Decode the entire DGPS spectrum in one go

http://www.blackcatsystems.com/software/dgps_decoding_software_SDR.html

How to enable stereo mixer in Windows

<https://www.youtube.com/watch?v=Ex69ma5vDoc>

Virtual Audio Cable

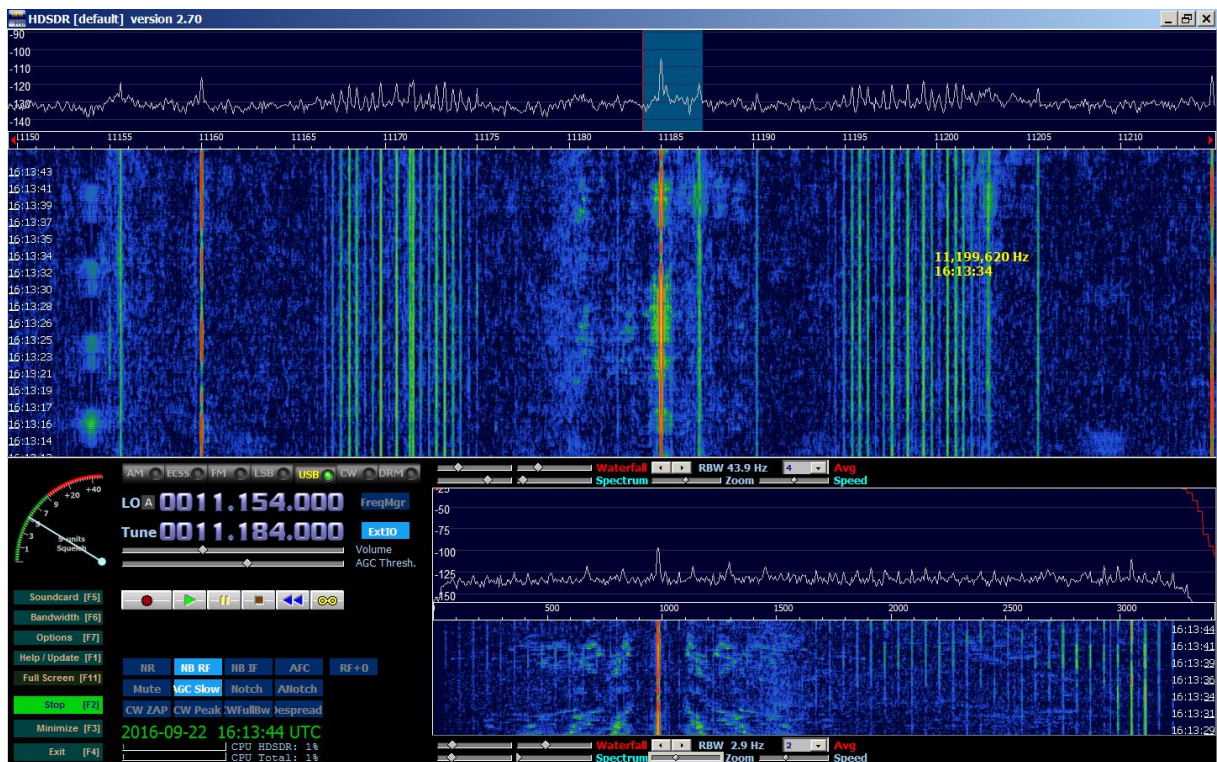
<http://software.muzychenko.net/eng/vac.htm>

Bits about ADC bits

<http://www.analog.com/media/en/training-seminars/tutorials/MT-003.pdf>

Things Harry Nyquist said

<http://www.ni.com/white-paper/2709/en/>



SWBC images and PC noise as displayed by HSDR