ezRA - Easy Radio Astronomy - Demonstration

- Sep-30-2022

The ezRA Easy Radio Astronomy set of programs are free PC tools to help explore Radio Astronomy. The programs run on the Python3 programming language, on Windows and Linux.

This is a demonstration of collecting radio data with the ezCol program into ezRA frequency spectrum .txt data files, and then analyzing that data with the ezCon, ezPlot, ezSky, and ezGal programs.

These are examples using the Windows 10 operating system.
The ezRA_20b_Software_Installation_Windows.pdf documentation describes the ezRA installation, leaving Python installed, and a directory structure at least like this, with a “radioSky” directory containing “ezRA”, “demo1”, and “demo2” subdirectories:

```
radioSky
├── ezRA
├── demo1
└── demo2
```

Start a Command Prompt window with demo1 as the current directory.

You can start a Command Prompt from the Windows 10 Start Menu or Taskbar, and "Change Directories” (cd) to your demo1 directory.

Or try this simpler way: In a common Windows 10 File Explorer, with the demo1 directory opened, left-mouse-click on the right empty blank part of the Address Bar at the top, and type

```
cmd
```

and tap the Enter keyboard key, and a black-background Command Prompt window with demo1 as the current directory should pop up.

Maybe hold the Windows keyboard key (bottom left) down and tap the keyboard up arrow key (bottom right) to maximize the size of that focused Command Prompt window.

We begin data collection with the ezCol program.

ezCol Data Collector

ezCol - COLlect radio signals into integrated frequency spectrum ezRA .txt data files.

The heirarchy of directories on a computer are like an ordered maze of rooms, with levels.
If needed, Change Directory to that “demo1” directory, with the command

```
cd demo1
```

We now are “standing”, with our feet, in the demo1 directory.
One level above us, is a directory with this demo1 directory, with the ezRA directory is beside it.

Now standing in the demo1 directory, to access the ezCol.py program file, we need to reach up one level (..) and over into the neighboring “ezRA” directory. On the command line we precede all that with the Python “py”, and ask the “ezCol.py” program for “help”:

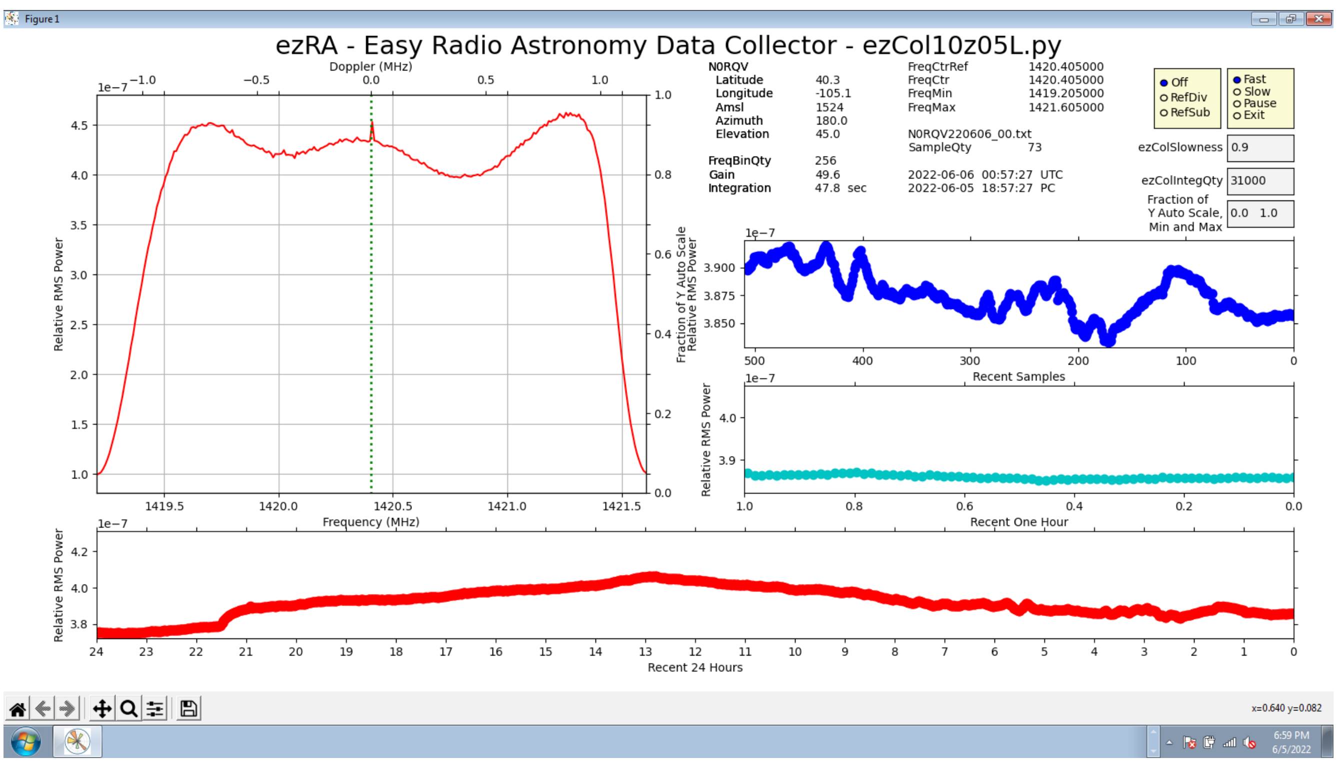
```
py  ..\ezRA\ezCol.py  -help
```

and a bunch of ezCol usage help text is printed on the screen.

Ignoring that help, we try the simpler command:

```
py  ..\ezRA\ezCol.py
```

and eventually a graphics window appears:



The computer is very busy taking thousands of readings, averaging them together into samples, writing samples to a data file, and eventually updating this graphical Dashboard. The computer will be very slow to respond, but it will respond. Please be patient with it. Hopefully, speed improvements are coming.

ezCol is already writing a data file, into a “data” directory (that it just created, if needed).
The data may be useful, and the timestamps are probably good.
But ezCol is using default values for the antenna earth location (latitude, longitude, Altitude-above-Mean-Sea-Level), antenna name, and antenna bearing (azimuth and elevation). We can see those values displayed in the text of the window’s upper right. Those wrong values could be repaired later with ezFix. But, we will soon run ezCol again, with the correct values, below.

First, a screen tour. In the top left is a plot of the latest frequency spectrum, with a dashed green vertical line at the center frequency. The plot will update occasionally, dependent on the speed of our PC. The “Integration” time is displayed in the text of the window’s upper right. Hydrogen emission may appear as an odd pointy bump near the top center. Some receiver hardware may also provide a very sharp spike on the center frequency, but that can hidden later with ezCon analysis.

As time passes, more samples appear in the 3 other plots, as blue, green, or red dots. Some SDR radios transition from high values to low values as the radio hardware warms up. ezCol or ezFix or ezCon can remove those misleading start-up data samples later.

The bottom stripchart plot (red) shows auto-scaled data from the previous 24 hours, with the newest data at the right. Values will vary with temperature. ezCon can try to remove that variation.

The middle stripchart plot (green) shows the same data, but only from the most recent one hour, with the newest data at the right. The Y-scale is auto-scaled.

The upper stripchart plot (blue) shows the same data, but only the most recent 500 samples, with the newest data at the right. The Y-scale is auto-scaled.

The current data filename is displayed in the text of the window’s upper right.

At the start of a new UTC (Coordinated Universal Time) day, ezCol will automatically start a new data file. The filename will begin with some word (commonly taken from the antenna name), the starting UTC date and hour (YYMMDD_HH), perhaps a single letter (b-z), and ending with “.txt” . Like “bigDish210927_03c.txt” .

ezCol and ezFix try hard to not write over existing data files.
If a “data” directory does not already exist, ezCol will create a “data” directory at your feet.
When creating a data filename, ezCol checks if the name already exists in the “data” directory.
If nameYYMMDD_HH.txt already exists, ezCol will try by inserting a “b” character near the end, like nameYYMMDD_HHb.txt .
If nameYYMMDD_HHb.txt already exists, ezCol will instead try by inserting a “c”, like nameYYMMDD_HHc.txt .
And so on, through the letter “z”, searching for an available (unused) filename.

But as said above, this demo's default antenna location, name, and bearing are wrong. Let us try ezCol again and enter the correct values on the command line.

To stop this ezCol program, click the "Exit" circle in the upper right yellow box, and wait for the ezCol to notice. Or switch to the Command Prompt window and type a Control-C (hold down the Ctrl keyboard key (bottom left) and tap the keyboard C key).

Like above, in the Command Prompt window, still standing in that demo1 directory, we enter:

```
    py  ..\ezRA\ezCol.py  -help
```

and a bunch of ezCol usage help text is printed on the screen.

For this example, after deep consideration, we enter this long command, all on one line:

```
    py  ..\ezRA\ezCol.py
         -ezRAObsLat  40.4  -ezRAObsLon  -105.1  -ezRAObsAmsl  1554
         -ezRAObsName  N0RQV
         -ezConAzimuth 227.9  -ezConElevation  38.9
```

Now ezCol will record data knowing the antenna location of latitude 40.4 degrees, longitude -105.1 degrees, and Altitude above Mean Sea Level (AMSL) 1554 meters
(right-mouse clicking on the map on https://www.google.com/maps may help find your antenna location).
ezCol will use an antenna name (for the top of the plots) of "N0RQV".
ezCol will use an antenna bearing of azimuth 227.9 degrees and elevation 38.9 degrees.

Eventually a graphics window appears with the correct text in the upper right.
The ezFix program can help you correct those parameters later.

But let us fix all that right now.

ezDefaults.txt Files

That long command will work well, but wow, that was a lot of information to type into one line. Better to do that just once, and store all those unchanging arguments into a convenient text file.

Exactly. Store that text in a file of default program arguments, named “ezDefaults.txt”.

So, we interrupt the ezCol data collection to do some quick housekeeping. Like before, to stop this ezCol program, click the “Exit” circle in the upper right yellow box, and wait for the ezCol to notice. Or switch to the Command Prompt window and type a Control-C (hold down the Ctrl keyboard key (bottom left) and tap the keyboard C key).

Tap the keyboard up arrow key to recall that long command, drag your mouse over it to highlight it, and type a Control-C to copy that text. To create or edit the existing ezDefaults.txt in the current directory, still standing in that demo1 directory, we enter:

```
notepad .\ezDefaults.txt
```

and a Notepad application window pops up.
Quick, paste that long command line text into the file, by typing a Control-V .
Then rearrange the arguments as you wish.
Here is from the above example:

```
# this is a comment line
-ezRAObsLat    40.3   # Observatory Latitude  (degrees)
-ezRAObsLon    -105.1 # Observatory Longitude (degrees
-ezRAObsAmsl   1554   # Observatory Above Mean Sea Level (meters)
-ezRAObsName   N0RQV8 # Antenna Name
-ezColAzimuth    227.9  # Azimuth   pointing of antenna (degrees)
-ezColElevation  38.9   # Elevation pointing of antenna (degrees)
```

Any optional text on the line following a comment character (#) will be ignored.
Any order of information should work. Blank lines are allowed.

Commonly, you would physically adjust only the antenna’s Elevation angle.
45 degrees Elevation is a good starting point.
If you can, there are some advantages to point your antenna to the south (180 degrees).
This antenna has trees to the south. This antenna’s lowest sky horizon is to the southwest at 227.9 degrees Azimuth.

These default values can be overruled or appended, in several ways.
An ezRA program’s arguments are read first from inside the ezRA program,
then in order from the ezDefaults.txt in the ezRA program's directory,
then in order from the ezDefaults.txt in the current directory (where you are standing),
then in order from the command line. In the case of duplicates, the last definition read wins.

Wow. Many options.

Save and close this ezDefaults.txt file. Maybe by typing a Control-S and then by typing a Control-W .

ezCol Data Collector Continued

We recontinue the ezCol data collection example. Still standing in that demo1 directory, we enter the simpler:

```
py  ..\ezRA\ezCol.py
```

and the new ezDefault.txt values should appear in the text in the upper right.

If we can wait that long, after a full UTC day, we have a full data file in the "data" directory.

Still standing in that demo1 directory, we enter:

```
dir  data
```

and get:

```
N0RQV200525_00.txt
```

There may be no need to examine the .txt data file's contents, but it is just a simple text file.
Still standing in that demo1 directory, we enter:

```
more  data\default200525_00.txt
```

and promptly the file text appears. Use the keyboard's SpaceBar key to advance a page. Use the keyboard's Q key to exit (Quit). For help on the "more" command, enter

```
help more
```

The Windows Notepad program may take a while to open such a large text file, but it also works.
Still standing in that demo1 directory, we enter:

```
notepad  data\default200525_00.txt
```

and eventually a Notepad window appears. To exit, type a Control-W (hold down the Ctrl keyboard key (bottom left) and tap the keyboard W key).

The .txt file's text will start something like this:

```
from ezCol220826a.py
lat 40.3 long -105.1 amsl 1554.0 name N0RQV8
freqMin 1419.2 freqMax 1421.6 freqBinQty 256
az 227.9 el 38.9
# times are in UTC
# gain 49.6
# frequency spectrums of RMS power = sqrt(mean of sum of squares)
2022-08-25T00:00:06 206.197 206.197 207.678 209.662 …
```

That last line is long, with at least 257 words. Presented here, the end is trimmed off.

The first 10 characters of a valid ezRA frequency spectrum data .txt file must be "`from ezCol`".
This file's first line gives the long revision name of the program that created it.

This file's second line starts with "lat" and gives the antenna observatory's latitude, longitude, Altitude-above-Mean-Sea-Level (in meters), and antenna name.

This file's third line starts with "freqMin" and gives the data's Minimum Frequency (in MHz), Maximum Frequency, and the Quantity of Frequency Bins (the number of frequency channels).

This file’s fourth line starts with “az” and gives the data’s antenna bearing in Azimuth and Elevation degrees.

This file’s fifth line starts with “#” to indicate it is a comment line, and is usually ignored by ezRA programs. You are welcome to insert your own comment lines or blank lines.

This file’s eighth line starts with one timestamp word and 256 number words (one for each frequency bin (freqBin)), each separated by a single space character. There are many similar long data lines which follow.

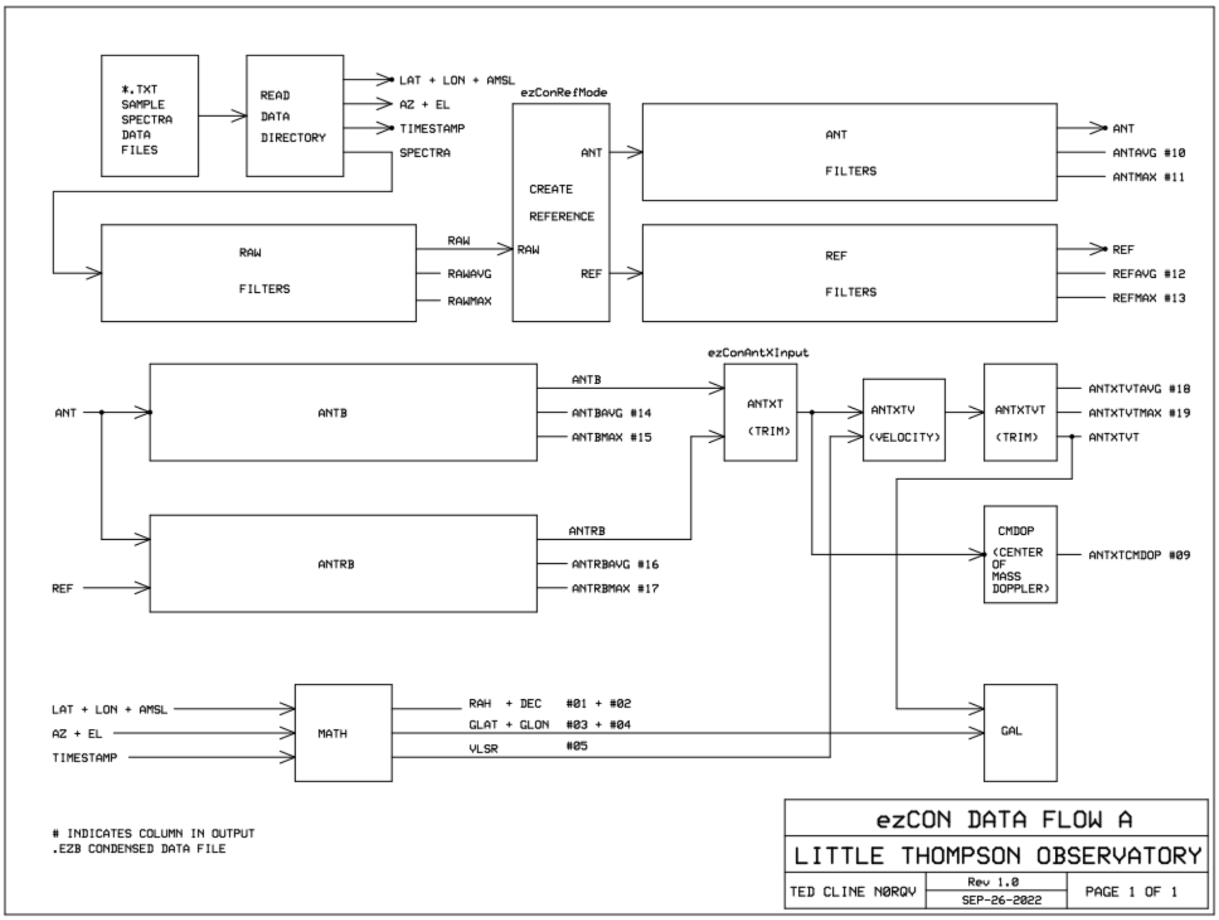
These simple .txt data files can be studied with a spreadsheet program like Excel or LibreOffice Calc.

The following ezRA programs create lots of plots from these data files. The next program is ezCon.

ezCon Data Condenser

ezCon - CONdense one or more frequency spectrum data .txt files into
one .ezb text data file, and perhaps one GALaxy crossing spectra *Gal.npz data file.

Here is simplified diagram of the data flow inside the ezCon program.



Starting at the top left, ezCon reads .txt data files, filters the spectra into Raw data samples, and then separates them into Antenna (Ant) samples and Reference (Ref) samples.

Continuing at center left, ezCon uses those Ant and Ref samples to create the AntB and AntRB signals. It then commonly selects AntB or AntRB, and creates AntXT, AntXTV, and AntXTVT signals.

ezCon eventually writes one .ezb text data file, and perhaps one GALaxy crossing spectra *Gal.npz data file. Along the way, ezCon optionally creates many plot files. All to better reveal Galactic hydrogen radio emission and the information it contains. Default values for the many options provide guidance.

Continuing with examples, still standing in that demo1 directory, we enter:

```
py  ..\ezRA\ezCon.py  -help
```

and a bunch of ezCon usage help text is printed on the screen.

Ignoring that help, we enter:

```
py  ..\ezRA\ezCon.py  data\N0RQV200503_00.txt
```

and a bunch of ezCon text is printed on the screen.
On the left, next to the current data filename, numbers count down the plots-still-to-process, and a bunch of "plotting" lines display the progress. Eventually, the program ends with free advertising,

```
The Society of Amateur Radio Astronomers (SARA)
             radio-astronomy.org
```

Not far above that text, it says how many seconds the program took to run, and the current PC time.
That might be helpful if the program took a long time to execute over a lot of data.
This example says

```
That Python command
 ..\ezRA\ezCon.py  data\N0RQV200503_00.txt
took 88 seconds = 1.5 minutes
```

Still standing in that demo1 directory, we enter:

```
dir
```

and see there are now lots of .png plot files at our feet.
At the bottom, that dir command says there are 93 files and directories at our feet.
Before the ezCon command, there was only the one “data” directory.

Each ezCon plot filename starts with “ezCon”, followed by a 3-digit number, followed by a description, followed by “.png” . The first one is “ezCon001raw.png”.

View that first plot file, perhaps by double-clicking the .png file,
or highlighting the .png file and pressing the keyboard Enter key,
or perhaps just entering

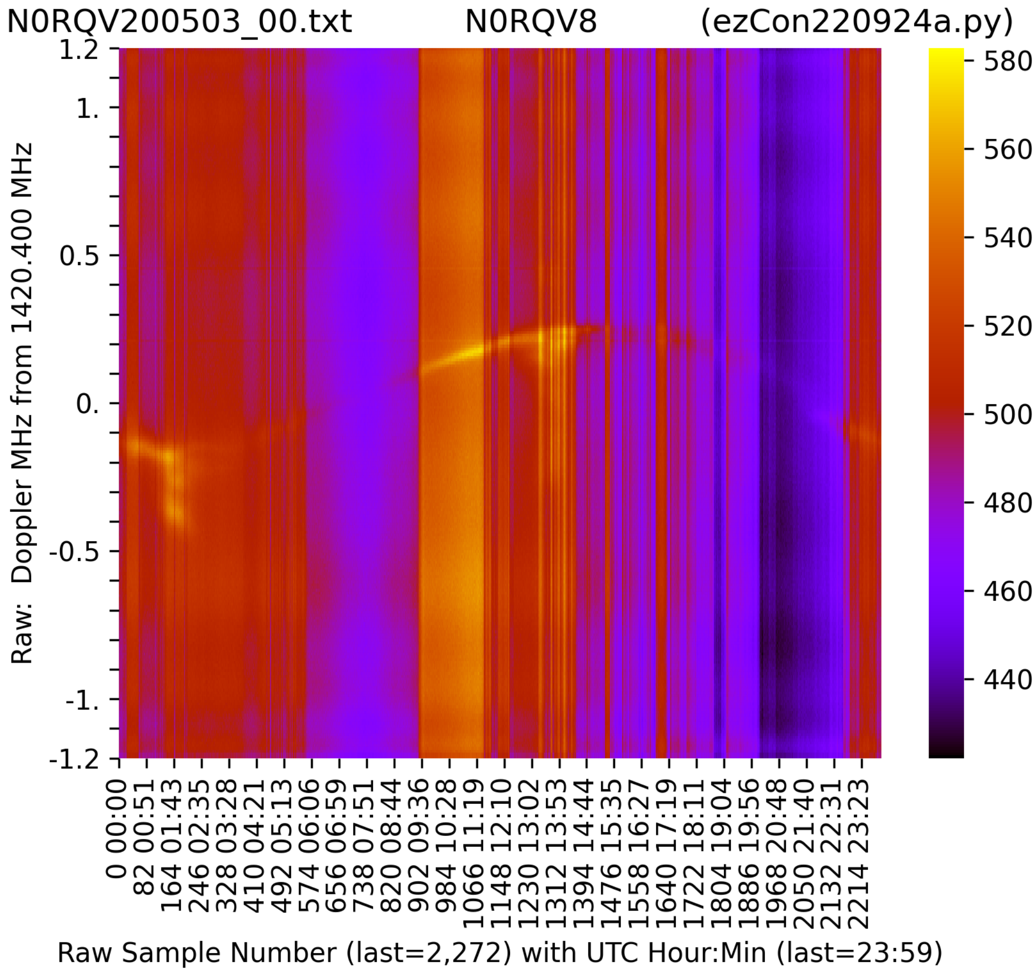
```
ezCon001raw.png
```

on the command line.

The filename is usually displayed in the top title bar.
Perhaps double-click that window title bar to maximize the window.
Maybe tap the keyboard right-arrow key to advance through the plots.
Holding down that right-arrow key longer than a tap, may advance several plots quickly.

Eventually all the ezCon plots will be discussed.
The ezRA philosophy is lots of plots, some for special cases.

Let us explore some of the easy plots first.



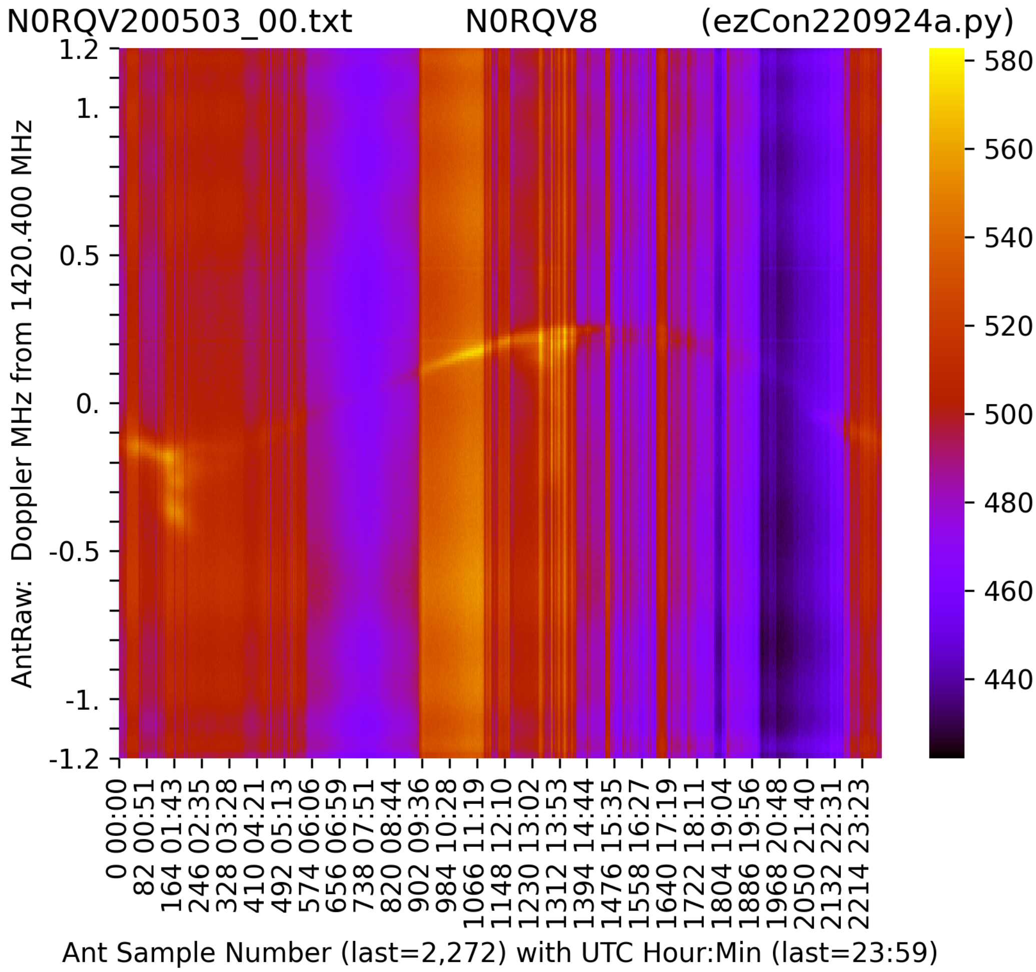
ezCon001raw.png

Across the top, is the filename of the last data file used, the antenna name, and in parentheses, the creating program's long revision filename.
On the left, is the Y-axis Label (description), usually the reason for the plot.
On the bottom, is the X-axis Label (description), commonly sample count with sample time, or Doppler frequency, or Galactic longitude.
On the right of the colorful heatmaps is a color bar index to reveal the values of the plot colors.

ezCon001raw.png, shown above,
displays the frequency spectrum for each Raw sample as a colorful vertical line, as the sample number increases to the right.
Usually (due to sorting of filenames), time increases with sample number, so time generally increases to the right.
Shown here, the 2,272 samples started at 00:00 UTC and ended with 23:59 UTC.
In the center we faintly see a sinusoidal wave, somewhat brighter. That is the weak 1420 MHz emission from Galactic hydrogen.
https://en.wikipedia.org/wiki/Hydrogen_line

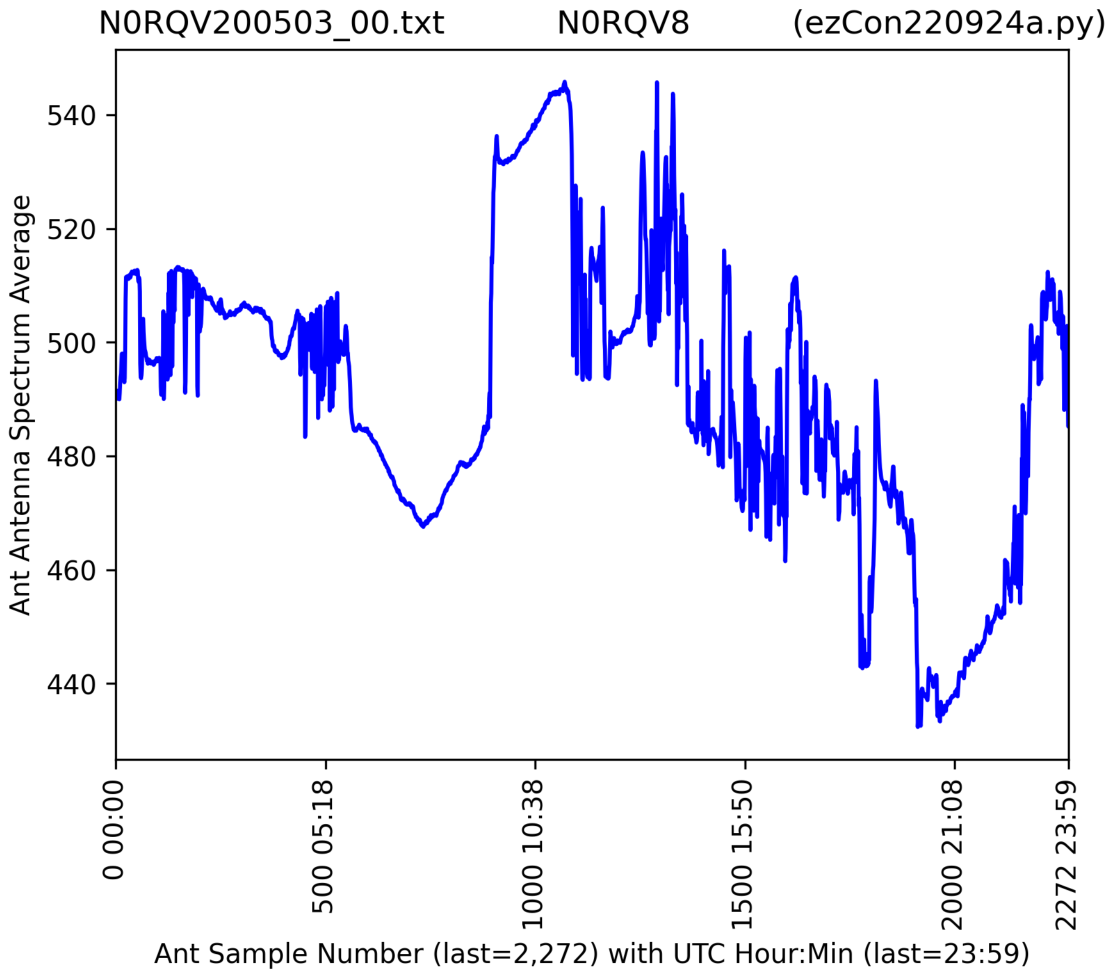
There are darker colors at bottom and top and (low and high frequencies), due to the receiver's uneven frequency response.
Ripples appear as vertical bands, due to uneven amplifier gain response over various time durations.
ezCon will attempt to remove those noises and imperfections.

These "Raw" samples can include Antenna (Ant) samples and Reference (Ref) samples.
This simple data file has only Ant samples.
The creation and use of Reference samples is explored later.



ezCon002antRaw.png
displays only Antenna (Ant) samples, and no Reference (Ref) samples.
This data file has only Ant samples, so this plot looks much like the previous plot.

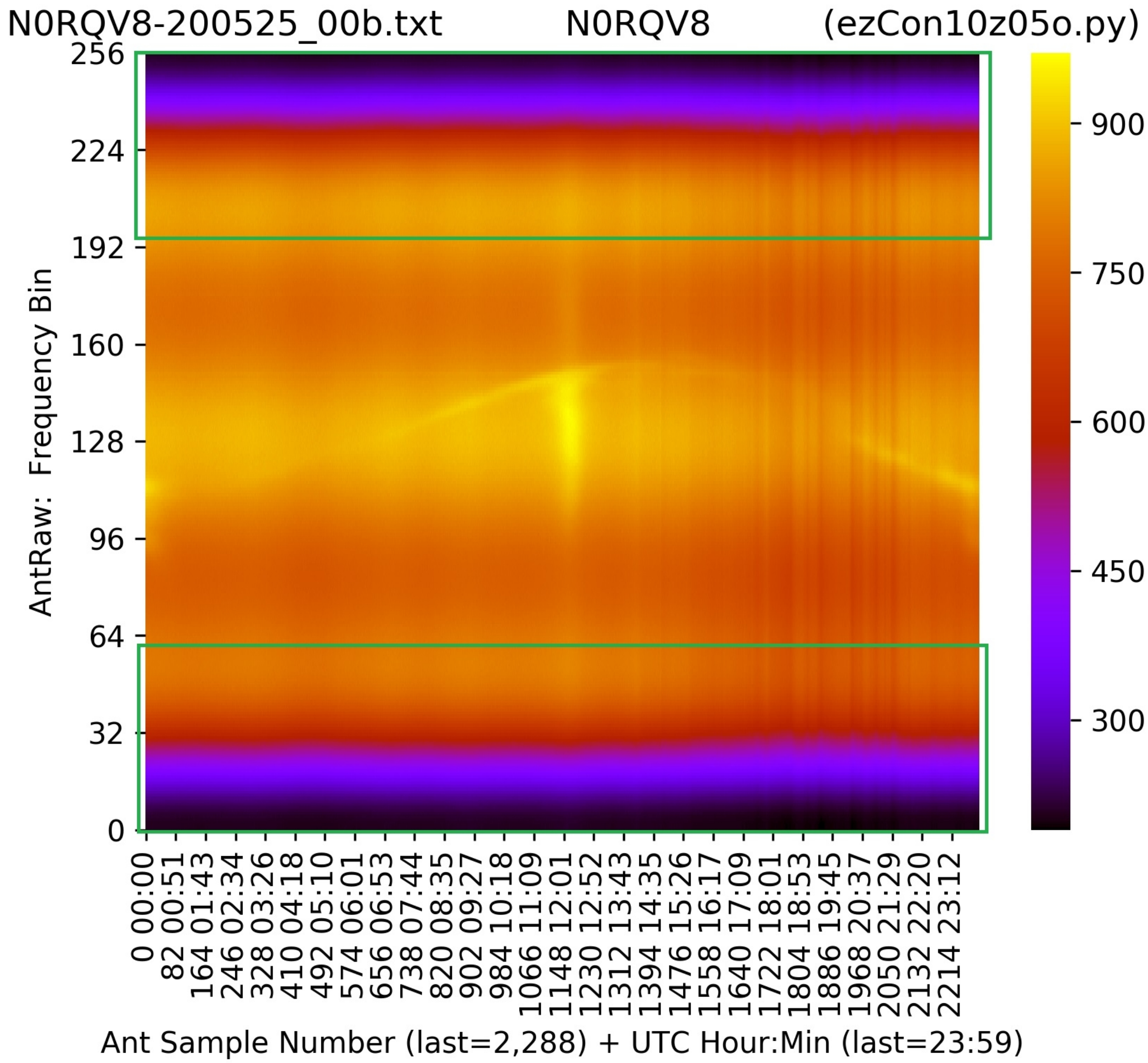
Here is the plot of the average of each vertical line sample spectrum,



ezCon110antAvg.png
displays the average value for each Ant sample spectrum (each colorful vertical line of previous plot).

Soon we will see a lonely Galactic hydrogen radio emission peak in the middle, near sample number 1312, but there is a lot of noise trying to hide it here.

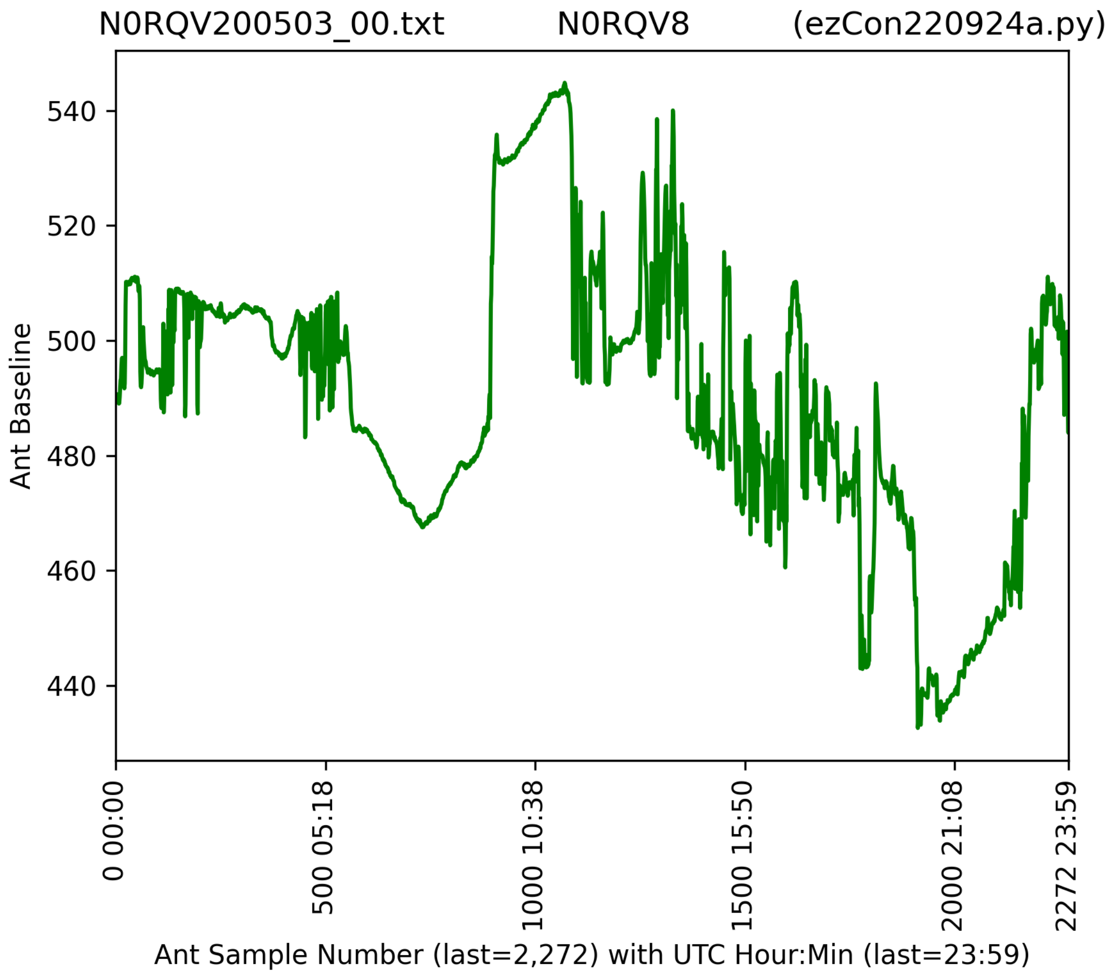
ezCon will process this Ant signal, trying to reduce the noise.



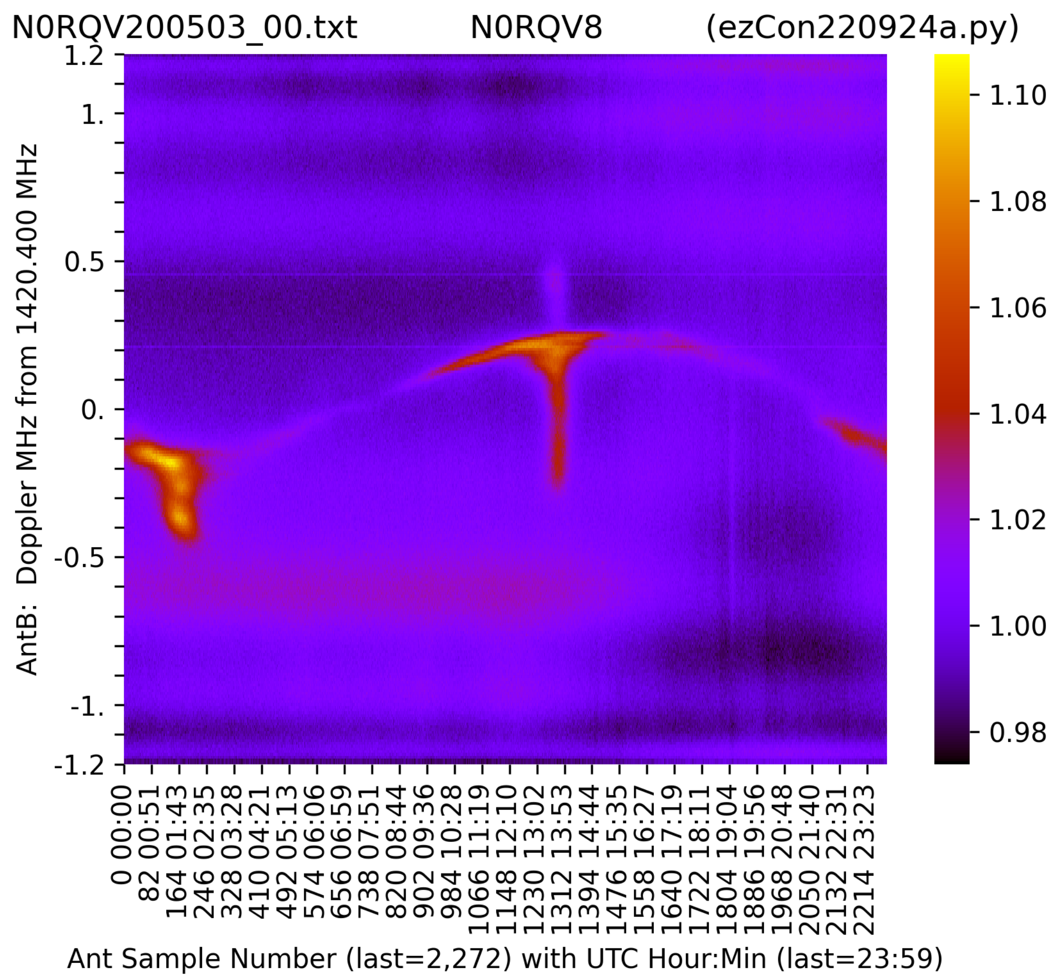
ezCon010antRaw.png with freqBin numbers
This is the same ezCon010antRaw plot, but after using the optional "-ezConDispFreqBin 1", to display the freqBin numbers on the Y-axis.

These low-frequency and high-frequency areas boxed in green show no Galactic hydrogen, but they still vary from sample to sample (vertical stripes like vertical curtains). ezCon uses these selected values in green to create a sample-by-sample AntBaseline value to smooth the plot from left to right.

For the first sample (colorful vertical line on extreme left), ezCon here uses the defaults for the optional -ezConAntBaselineFreqBinsFracL arguments to select and sum together the sample spectrum values of these lowest 60 freqBins and highest 60 freqBins. That sum is divided by 120 (= 60 + 60) to achieve one average for those selected values in that one sample. That process is repeated for each sample (each colorful vertical line). ezCon123antBaseline.png displays those sample-by-sample averages, as AntBaseline:



ezCon241antBaseline.png
displays the average of selected freqBin values, for each sample, as AntBaseline.
Those earlier plot ripples now appear as local minima, with various time durations.



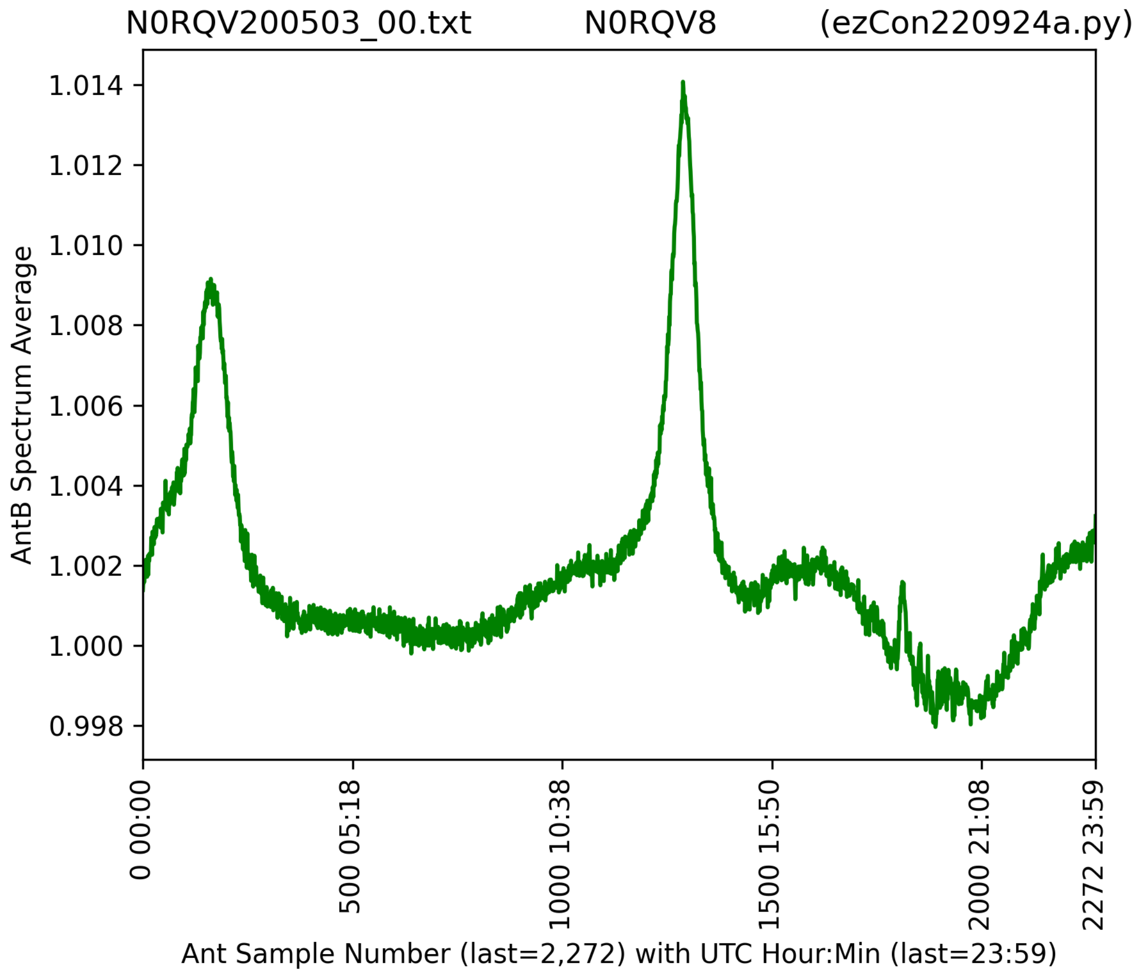
ezCon047antB.png
displays every AntRaw spectra value divided by its sample's AntBaseline value. This greatly reduces the left-to-right gain variability. Compare to the curtain ripples in the ezCon002antRaw.png, 4 plots above.

Now we start to see that lonely Galactic hydrogen radio emission peak in the middle, near sample number 1312.

These are not calibrated absolute values. These are relative values, with the values related to each other.

Without further analysis using Reference samples, this AntB is often the cleanest signal available.

Here is the plot of the average of each vertical sample spectrum,



ezCon114antBAvg.png
displays the average value for each AntB sample spectrum (each colorful vertical line of previous plot).

We usually see 2 bumps in energy per 24 hours. The bumps represent relatively more hydrogen radio emission. We will see these are from samples of the Galactic plane. One bump is usually stronger than the other.

Here, the gain variability ripple has been greatly reduced, to reveal the crossings of the Galactic inner (stronger) and outer (weaker) arms. The inner Galactic arm is closer to the Galactic center.

This AntBAvg signal is popular for plotting on sky maps, using ezSky.

.ezb Condensed Data File

That ezCon command created one condensed data file with a filename ending with ".ezb".

Still standing in that demo1 directory, to list all the files that end with the letter "b", we enter:

```
dir  *b
```

and get:

```
N0RQV200503_00.ezb
```

There may be no need to examine the .ezb data file's contents, but it is just a simple text file.
Still standing in that demo1 directory, we enter:

```
more  N0RQV200503_00.ezb
```

and promptly the file text appears. Use the keyboard's SpaceBar key to advance a page. Use the keyboard's Q key to exit (Quit).

The Windows Notepad program may take a while to open such a large text file, but it also works.
Still standing in that demo1 directory, we enter:

```
notepad  N0RQV200503_00.ezb
```

and eventually a Notepad window appears. To exit, type a Control-W (hold down the Ctrl keyboard key (bottom left) and tap the keyboard W key).

The .ezb file's text will start something like this:

```
# from ezCon220924a.py
# ..\ezRA\ezCon.py  data\N0RQV200503_00.txt

lat 40.4 long -105.1 amsl 1584.0 name N0RQV8
freqMin 1419.2 freqMax 1421.6 freqBinQty 256
ezbMenu: TimeUtcMjd  RaH  DecDeg  GLatDeg  GLonDeg  VLSR  Count
Spare1  Spare2  AntXTCMDop    AntAvg  AntMax    RefAvg  RefMax
AntBAvg  AntBMax    AntRBAvg  AntRBMax    AntXTVTAvg  AntXTVTMax
#        0           1    2        3        4        5     6
7       8       9             10      11        12      13
14       15         16        17          18          19
58972.00006 5.230 -3.571 -23.294 -155.349 3.331e+01 1 0 0 0
4.90132e+02 5.07724e+02 4.90132e+02 5.07724e+02 1.00139e+00
1.03733e+00 1.00000e+00 1.00000e+00 1.00161e+00 1.03671e+00
58972.00049 5.240 -3.571 -23.158 -155.270 3.338e+01 1 0 0 0
4.90526e+02 5.07017e+02 4.90132e+02 5.07724e+02 1.00159e+00
1.03526e+00 1.00020e+00 1.01092e+00 1.00172e+00 1.03508e+00
…
```

If you have a wide display, you may see there are 20 ragged columns of numbers, with a few header lines.

The .ezb file’s first line is blank, with no characters.

The second line gives the long revision name of the ezCon program that created it. As a comment, it is largely ignored by the ezRA programs.

The third line is the command that created this file. Eventually, we will adjust ezCon program parameters, and this line will record what settings were used. As a comment, it is largely ignored by the ezRA programs.

The fourth line is blank, with no characters.

The fifth line records the data’s earth location info, and the antenna Name (for plot titles). This information may be used later to plot the data’s azimuth and elevation.

The sixth line records the data’s Minimum Frequency (in MHz) and Maximum Frequency. No immediate need for this frequency information, but helpful to describe the data set.

Next is the ezbMenu line, with the titles of the 20 data columns to follow.
Next is a comment line, with the 20 column numbers

Then comes one line for each sample’s condensed data, in 20 columns.

The ezbMenu of 20 .ezb file columns may be separated into 2 groups of related values:
The 9 “Coordinate” columns, and the 11 “Signal” columns (AntXTCMDop, then 5 Averages and 5 Maximums).

All Count values are currently 1 .
All Spare1 and Spare2 values are currently 0 .

ezSky

ezSky - SKY maps from one or more .ezb condensed data files.

That ezCon command created one condensed data file with a filename ending with .ezb. ezSky reads that .ezb file and creates plots of the radio sky.

Continuing with examples, in the Command Prompt window, still standing in that demo1 directory, we enter:

```
py  ..\ezRA\ezSky.py  N0RQV200503_00.ezb
```

and a bunch of ezSky text is printed on the screen.
Numbers on the left count down the plots-still-to-process.

Near the end of all that text, it says

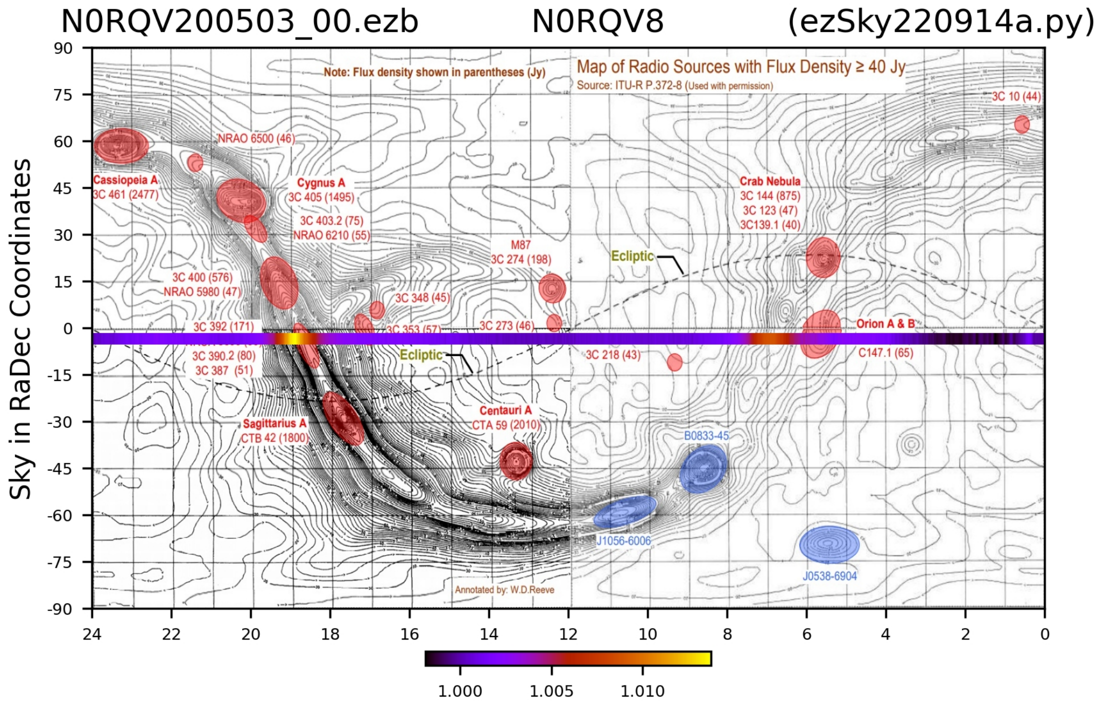
```
That Python command
 ..\ezRA\ezSky.py  N0RQV200503_00.ezb
took 10 seconds = 0.2 minutes
```

There are now more plots at our feet.

Each ezSky plot filename starts with “ezSky”, followed by a 3-digit number, followed by a description, followed by “.png” . The first one is “ezSky001ra.png”.

The first few ezSky plots present the data that was fed into the program. If that input data is faulty, the sky plots will be faulty.

We skip ahead to a plot of the radio sky.



ezSky401RBPF_14AntBAvg.png
displays the AntBAvg signal on the RaDec sky.

This plot shows the complete radio sky with Right Ascension increasing to the left, from zero to 24 hours, and Declination increasing up from -90 to +90 degrees. In the background image we see the Galaxy sinusoidally dipping low in the center, with several objects labeled. The center of the Galaxy is called "Sagittarius A*" at about RaDec 17.7h −29.0 in that red blob.

ezRA uses at least 3 of the 5 coordinate systems described here,
https://en.wikipedia.org/wiki/Astronomical_coordinate_systems
The ezRA antenna bearing uses Azimuth and Elevation (also called Altitude) coordinates (AzEl).
The Right Ascension and Declination (RaDec) plot above, is a popular view of the optical and radio skies. ezSky also plots views using Galactic Latitude and Galactic Longitude (Galactic coordinates).

The background image above was modified from "Important Celestial Radio Sources" by Whitham D. Reeve, and shows the total radio sky temperature at 408 MHz,
http://www.reeve.com/Documents/Articles%20Papers/Reeve_CelestialRadioSources.pdf
See also International Telecommunication Union (ITU) Recommendation P.372-8 (04/03)
https://www.itu.int/rec/R-REC-P.372-8-200304-S/en
which leads to page 20-23 of
https://www.itu.int/dms_pubrec/itu-r/rec/p/R-REC-P.372-8-200304-S!!PDF-E.pdf

This drift-scan data example's two bumps of the default AntBAvg input signal show up as brighter spots on the horizontal trace, and the brighter spots fairly align with the Galaxy on the background.

Bright spots may not align with the Galaxy if data timestamps, or antenna location, or antenna bearing were wrong. The whole horizontal trace might be shifted to the left or right, or up or down.

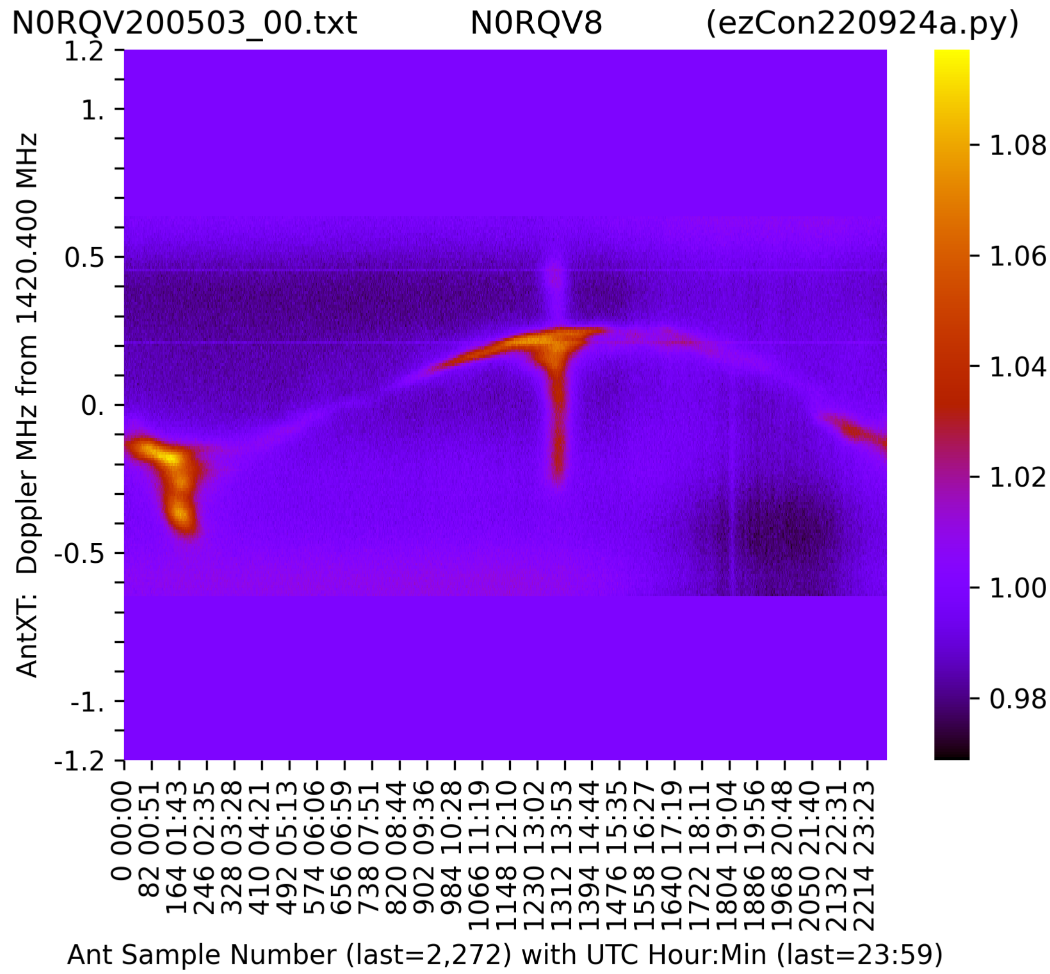
Raising the antenna's elevation, and then collecting data from another 24-hour drift-scan, would create another .ezb condensed data file, to provide a similar, but higher horizontal declination trace across the sky. By plotting many .ezb files together, we begin to plot the 1420 MHz radio sky.

AntX from AntB

This demo has created a useful AntB signal without using any Reference samples. We will explore Reference samples and AntRB later.

This ezCon signal processing continues with AntX, which can use either AntB or AntRB.

Because they contain no Galactic hydrogen, the bottom and top (low and high frequencies) can be trimmed to 1. The sample spectrum values of these lowest 60 freqBins and highest 60 freqBins can be trimmed to 1. ezCon081antXT.png displays those Trimmed samples,



ezCon081antXT.png
("T" added for Trimmed) displays the samples from either AntB or AntRB, after Trimming the low and high frequencies.

In this example, the earlier ezCon047antB.png had a mottled background. This frequency trimming improved the background image of this AntB signal.

Velocity from the Local Standard of Rest (VLSR)

There are differing definitions of "VLSR".
"Velocity from the Local Standard of Rest (VLSR)" works well here.

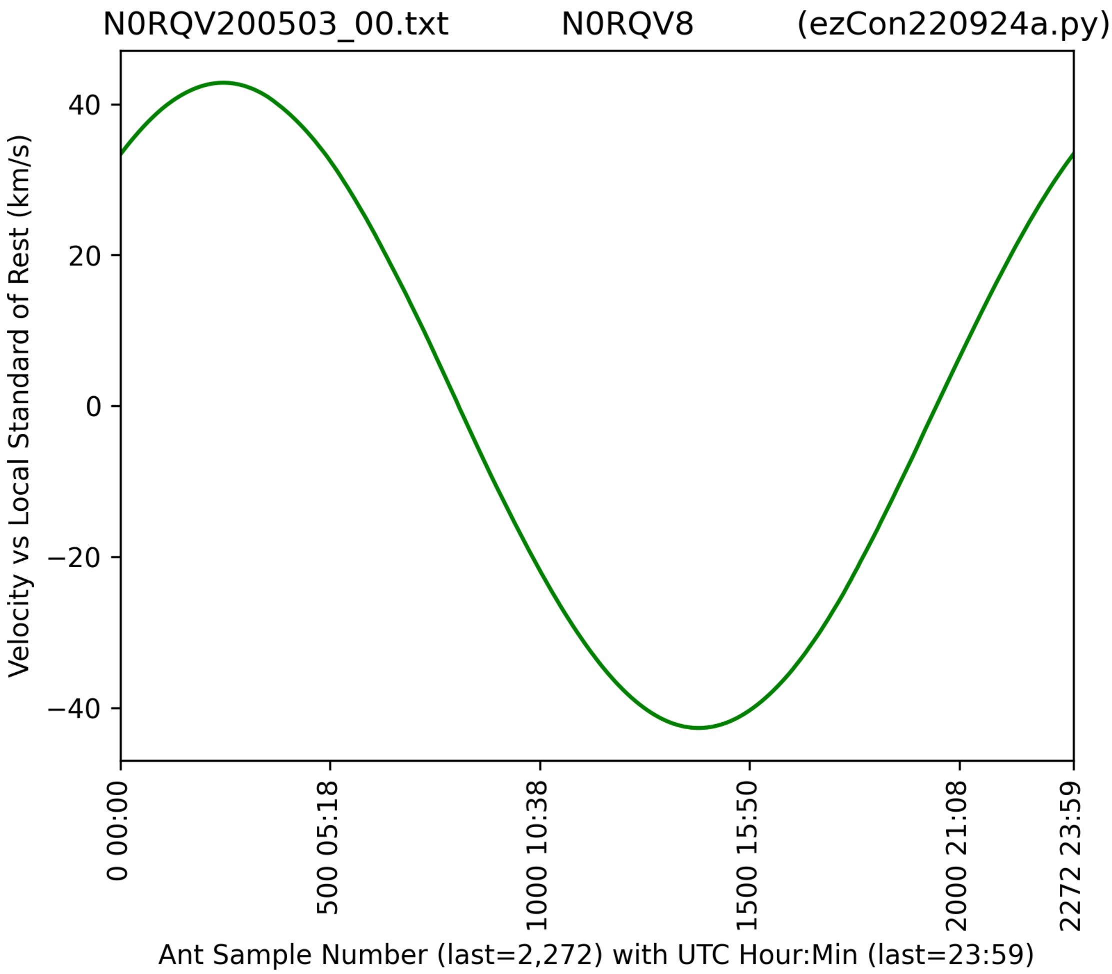
The Sun is moving through the Galaxy toward the Solar apex.
https://en.wikipedia.org/wiki/Solar_apex
ezCon uses the MIT Haystack SRT geom.c velocity standard of “20 km/s towards ra=18h dec=30.0”.
And the Earth orbits around the Sun, in one year.
Those two movements combine, and affect the data sample frequencies, and the sample spectra eventually need to be corrected.

The radio astronomy antenna collects a sample's data from Galactic hydrogen.
If the antenna and earth are approaching that Galactic hydrogen, that Galactic hydrogen appears to be approaching the antenna.
If the antenna and earth are receding from that Galactic hydrogen, that Galactic hydrogen appears to be receding from the antenna.

Edwin Hubble, as early as 1929, arbitrarily defined distant objects moving away from Earth as having "positive velocity".
https://en.wikipedia.org/wiki/Edwin_Hubble

From the antenna's "Local Standard of Rest" (the earth it is attached to), any receding Galactic hydrogen has a "positive velocity".

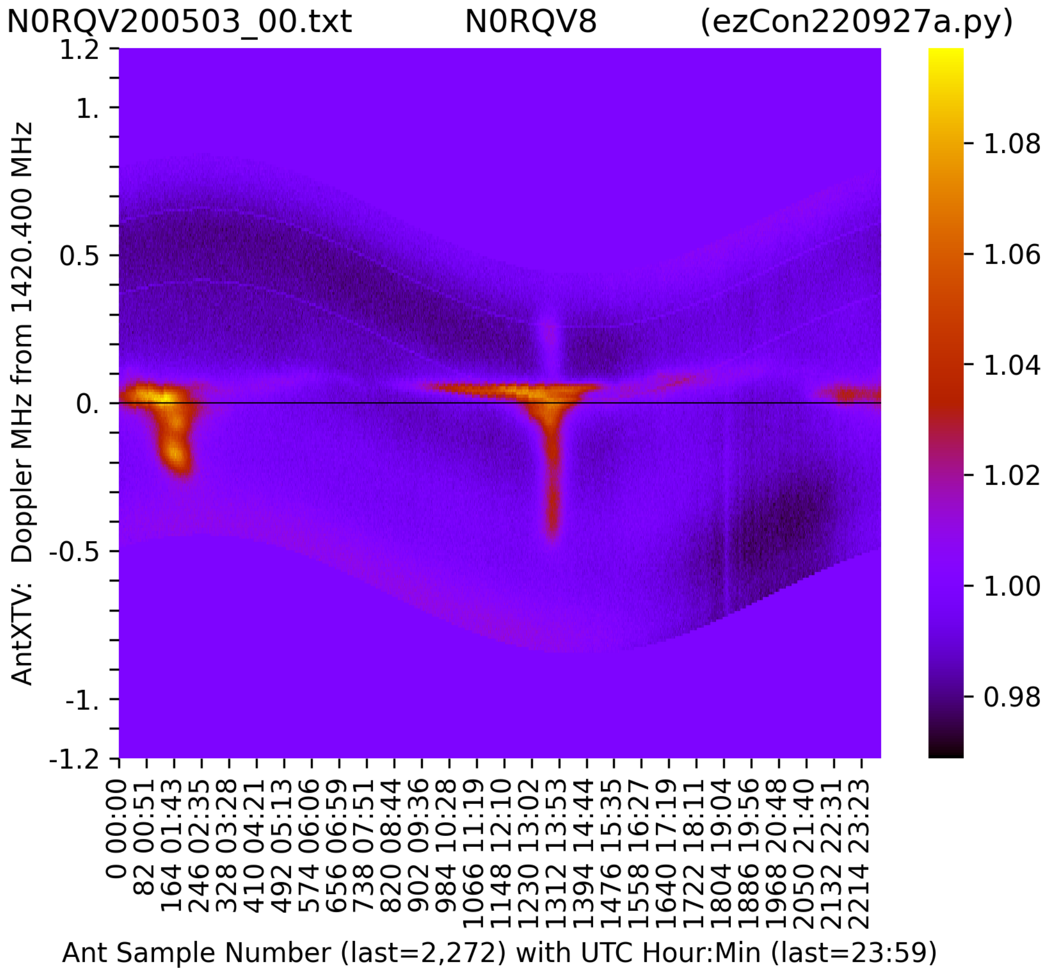
ezCon calculates the data sample's Galactic hydrogen's "Velocity from the Local Standard of Rest (VLSR)".
VLSR is positive when the Galactic hydrogen is receding.
VLSR is negative when the Galactic hydrogen is approaching.
The VLSR can be plotted, and saved as ezCon105vlsr.png .



ezCon105vlsr.png
displays the Velocity from the Local Standard of Rest (VLSR), calculated for each sample.

VLSR varies between about -50 and +50 km per second, depending on antenna bearing and date and time.
For a given antenna drift-scan with unchanging azimuth and elevation, the VLSR varies sinusoidally over 24 hours.
Sometimes the antenna is observing from the “front” of the earth (like looking out a moving car's front windshield).
Maybe 12 hours later, the antenna is observing from the “back” of the earth (like looking out the car's rear window).

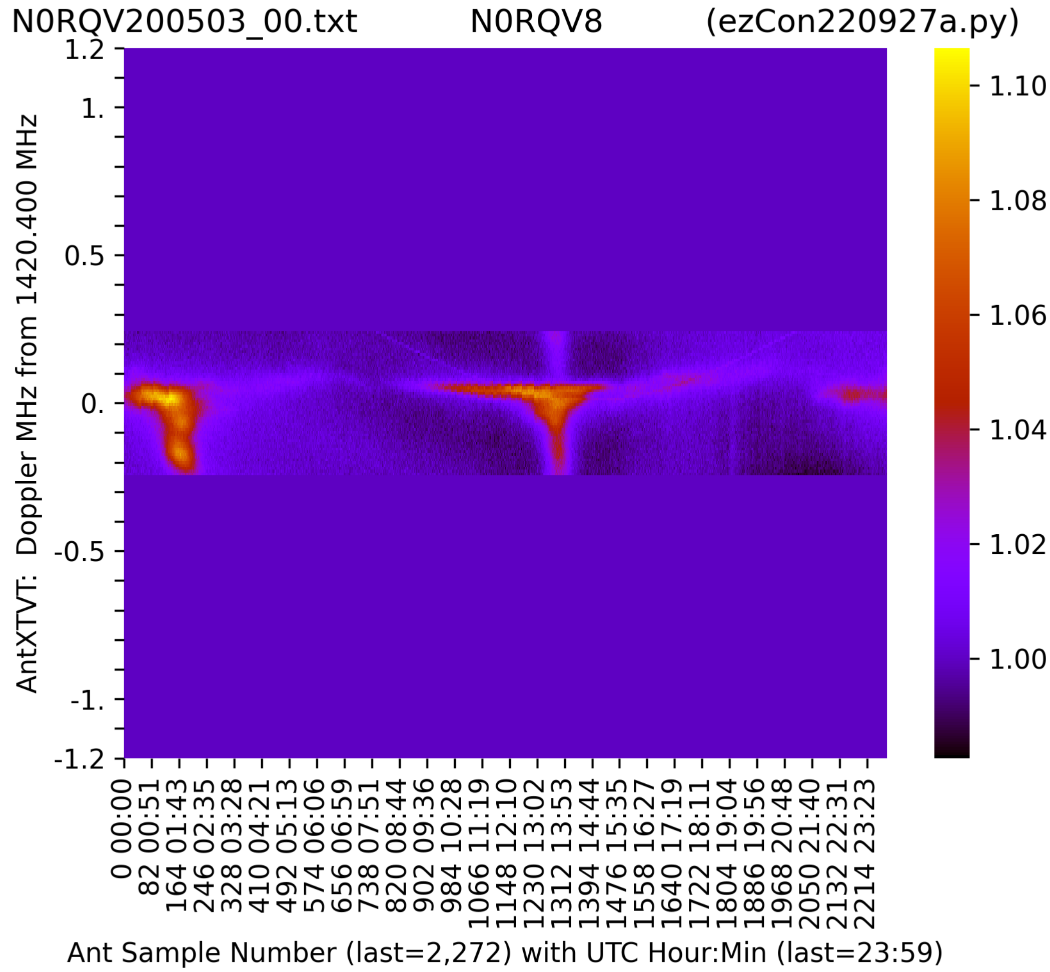
If the antenna bearing has a negative VLSR, the received frequency from the approaching distant hydrogen will be higher, due to the Doppler effect. Conversely, if the antenna bearing has a positive VLSR, the received frequency from the receding distant hydrogen will be lower, due to the Doppler effect.
https://en.wikipedia.org/wiki/Doppler_effect
ezCon uses each sample’s VLSR value to correct for this Doppler effect, and frequency-shift that sample’s spectrum (up or down), to create the AntXTV signal.



ezCon082antXTV.png
("V" added for Velocity-corrected) displays AntXT samples after Doppler-frequency-correcting with the sample's calculated VLSR.

Suddenly that yellow sinusoidal Galactic hydrogen is flattened, to be closer to zero Doppler.
ezCon adds a thin black line at zero Doppler, for comparison.
This AntXTV signal is used to study the true relative velocity of the Galactic hydrogen.

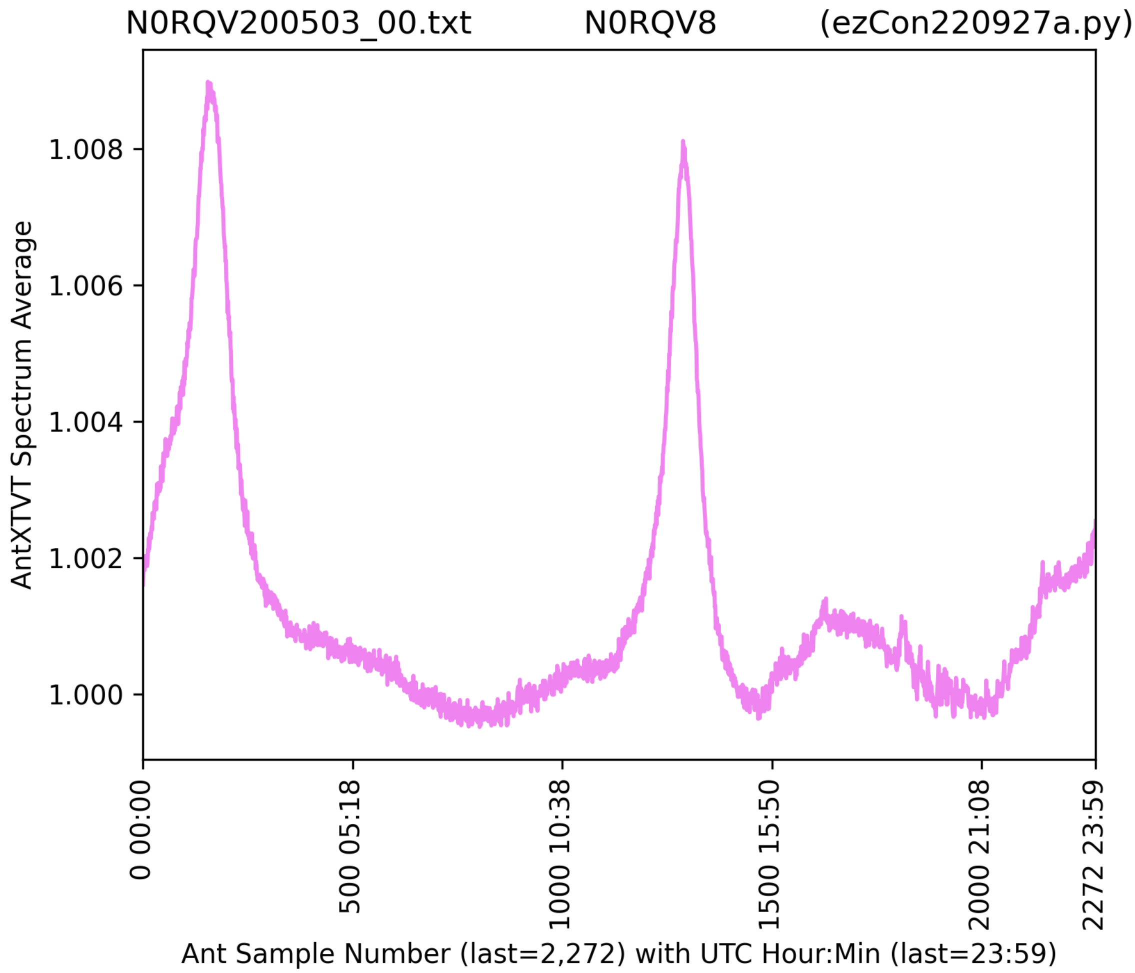
After all that frequency shifting of spectra, the low and high freqBin values of AntXTV look wavy and confusing. Time for another frequency trimming.



ezCon087antXTVT.png
(another “T” added for Trimmed) displays AntXTV samples after Trimming the low and high frequencies.

In this example, the frequency trimming may have improved the background image of this AntXTVT signal.

And here is the plot of the average of each of those vertical sample spectra,



ezCon118antXTVTAvg.png
displays the average value for each AntXTVT sample spectrum (each colorful vertical line of previous plot).

In this example, that last frequency trimming removed too much of the middle peak. Those frequency trim values could be easily adjusted.

Perhaps having AntX use the AntRB input will help. And so we start to explore Reference samples.

Reference Samples

Robert H. Dicke was an American astronomer who worked on the development of radar, during the Second World War. He designed the Dicke radiometer, a microwave receiver.
https://en.wikipedia.org/wiki/Robert_H._Dicke

Dicke's receiver would compare the antenna signal to a stable reference, thousands of times a second. ezCon can make use of Dicke Reference samples to help neutralize minute-to-minute and hour-to-hour system gain variation, and help neutralize non-linear frequency response.

EzCon can use Reference samples created by several methods, using the -ezConRefMode optional keyword. Here are some methods, in order of decreasing effectiveness (and their command line parameter):

A. Measure a resistor on the antenna side of the first radio amplifier (#10, #20).
B. Measure the antenna, but at a different frequency, away from hydrogen emission (#10).
C. Use the file's Nth sample, a quiet one, for comparison, for all file data samples (#-N).
D. Use the file's first sample for comparison, for all file data samples (#0).
E. Use the file's average sample spectrum for comparison, for all file data samples (#2).
F. Use a spectrum of 1.0 for all frequencies for comparison, for all file data samples (#1).

Additional hardware is required for A above. EzCol has supported all choices.

Choices A and B require that the Reference samples are marked in their sample dataflags.
Antenna samples are compared to the most recent Reference sample.

ezConRefMode 10 is the default.

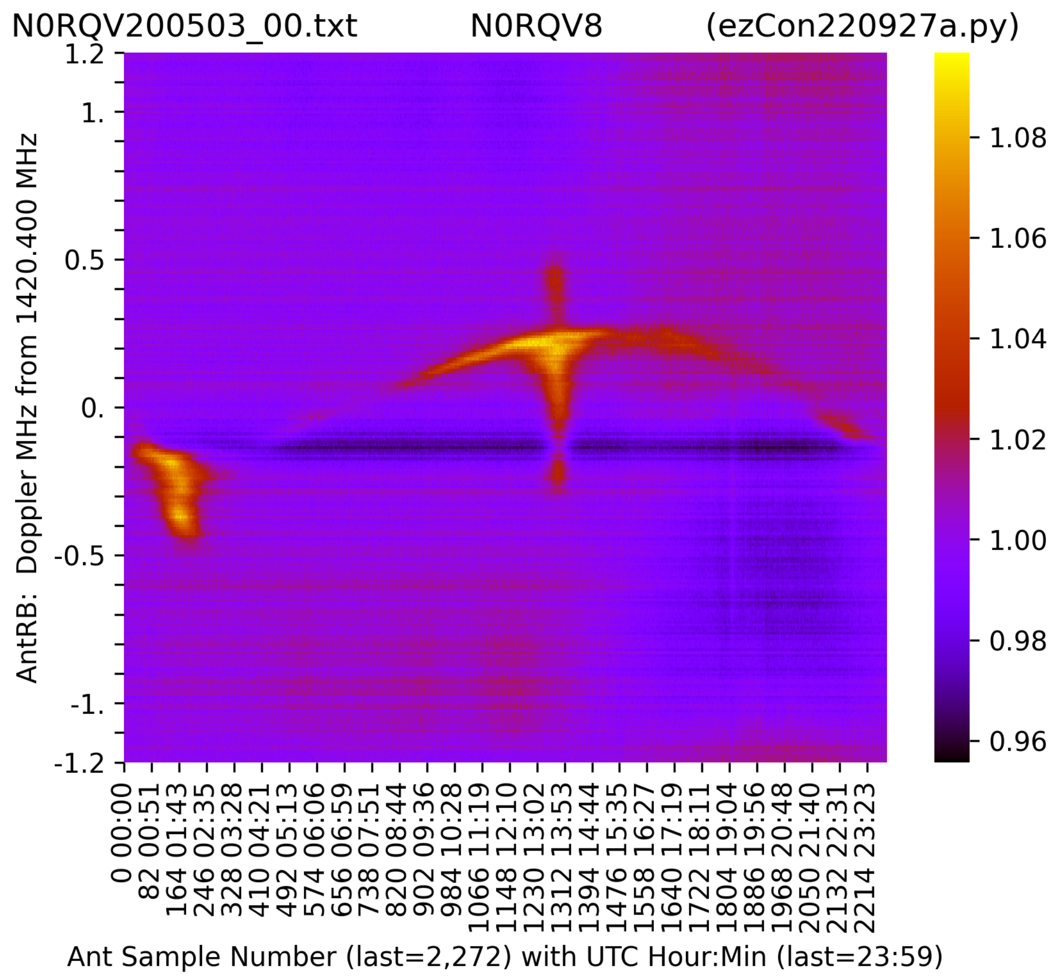
Creating One Reference Sample

We want to better detect Galactic hydrogen by comparing our Antenna signal to a similar signal that does not have Galactic hydrogen (Reference).

We can try choice C of the earlier Reference samples list, on the current data set, by simply running ezCon with a new argument:

C. Use the file's Nth sample, a quiet one, for comparison, for all file data samples (#-N).

The ezCon114antBAvg plot above says there was little hydrogen (a low plot value) at about sample 760. Let us compare all the samples to that one sample 760, that has little hydrogen. This should change the signal plots whose filenames have an "R" in them. Before we do that, we look at this current AntRB plot with no Reference samples,



ezCon067antRB.png.png
without Reference samples.

Still standing in that demo1 directory, we enter:

```
py  ..\ezRA\ezCon.py  -h
```

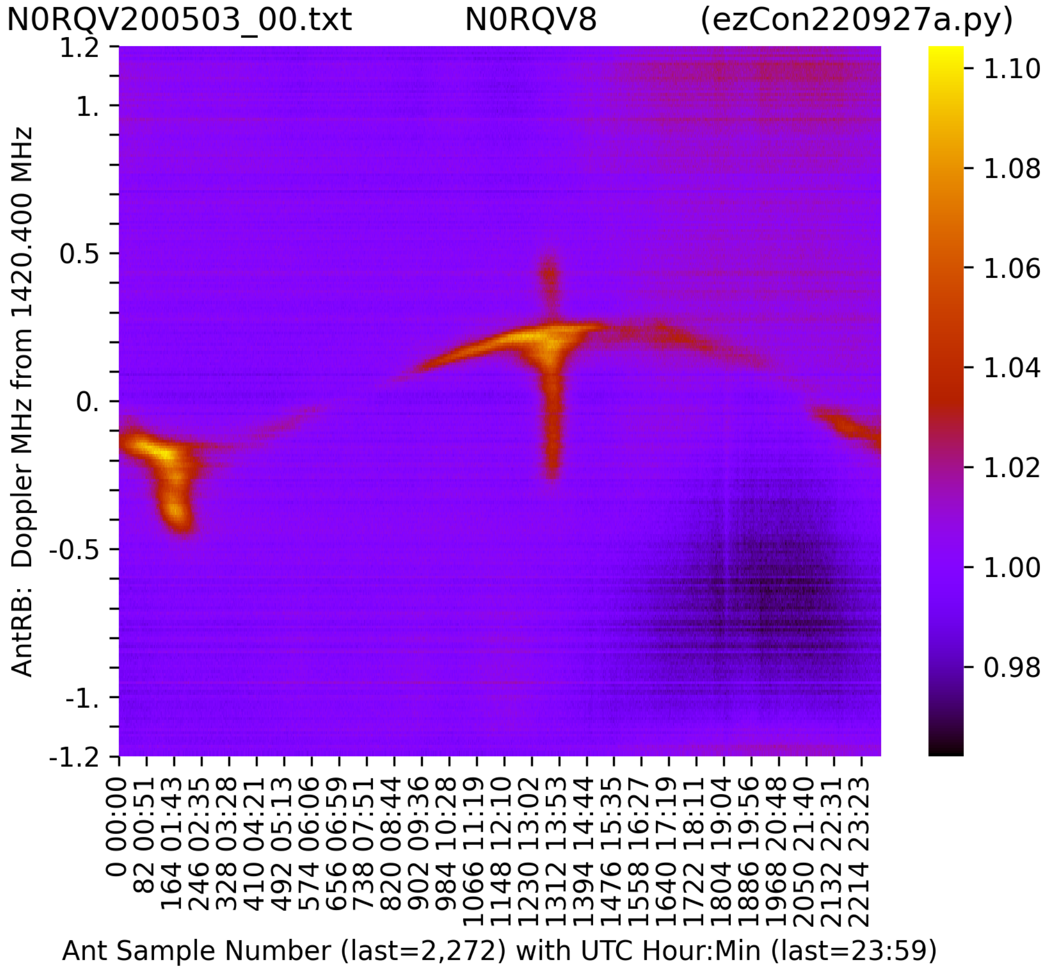
for the ezCon usage help text, which says there are several Reference Mode choices,

(Dicke Reference sample creation method, default = 10)
-ezConRefMode N < 0: REF = spectrum from -Nth ANT sample
-ezConRefMode -1403: REF = spectrum from ANT sample number 1403
-ezConRefMode 0: REF = spectrum from first ANT sample, number 0
-ezConRefMode 1: REF = 1.0 (no REF, neutral spectrum)
-ezConRefMode 2: REF = spectrum from rawByFreqBinAvg spectrum average
-ezConRefMode 10: REF = last REF sample marked in data, if none will use sample 0
-ezConRefMode 20: REF detection by ezCon software (for refPulser hardware)

So, to use sample 760 as a single Reference sample, still standing in that demo1 directory, we enter:

```
py  ..\ezRA\ezCon.py  data\N0RQV200503_00.txt  -ezConRefMode  -760
```

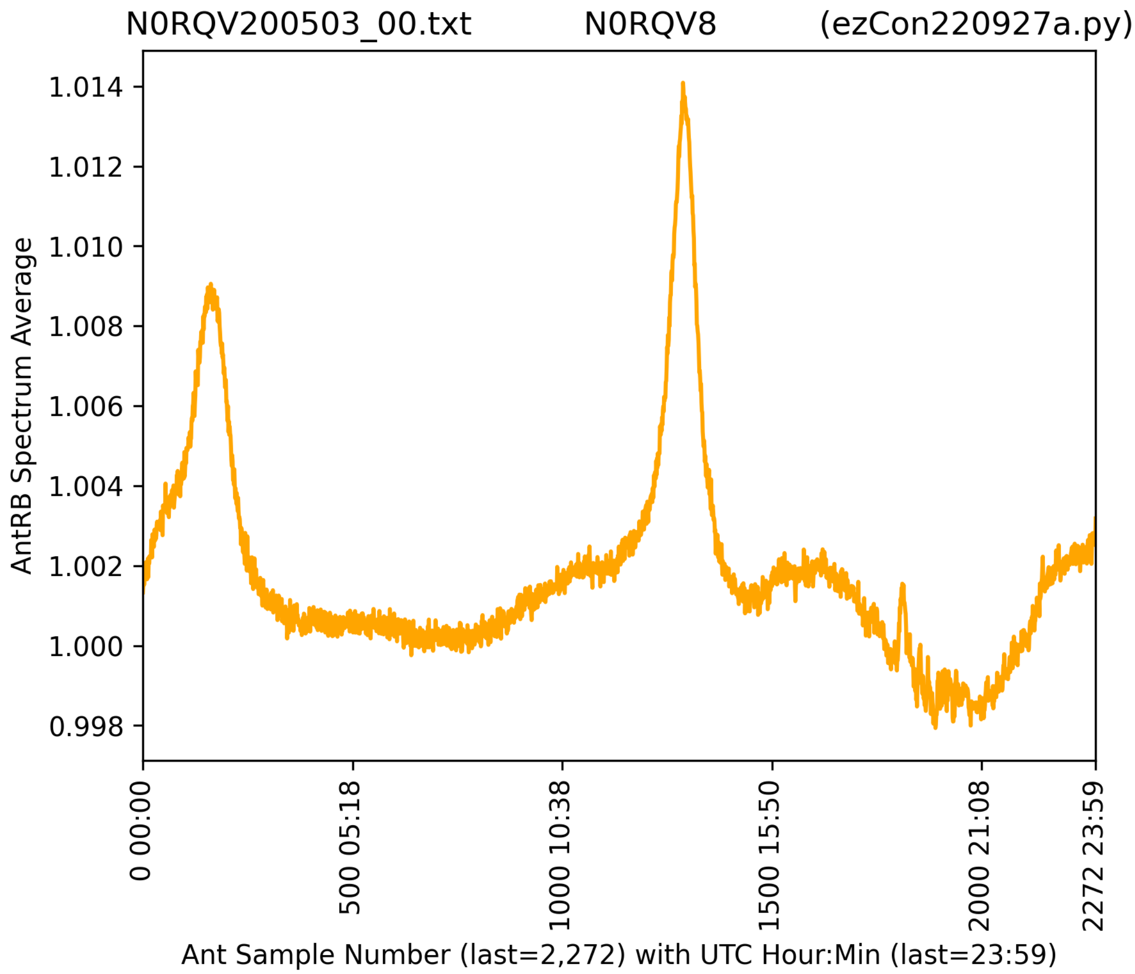
and ezCon plots are replaced with new ones. The plot names with R (for Reference) may have changed, like AntRB,



ezCon067antRB.png
now using a single Reference sample.

The AntRB contrast has improved. Yes, the hydrogen appears to be weak at sample 760, so that is a reasonable choice for a single Reference sample. The single Reference works best for the samples near the sample 760, and not so good for samples far from the sample 760. That becomes apparent in a slight horizontal darkening spreading from the Galactic hydrogen trace at the sample 760 point. More frequent Reference samples is usually better, just like collecting more data is usually better.

And here is the plot of the average of each colorful vertical sample spectrum,



ezCon116antRBAvg.png
displays the average value for each AntRB sample spectrum (each colorful vertical line of the previous plot).

This AntRB Average trace is just about identical to the earlier ezCon114antBAvg plot.

This AntRBAvg signal is popular for plotting on sky maps, using ezSky.

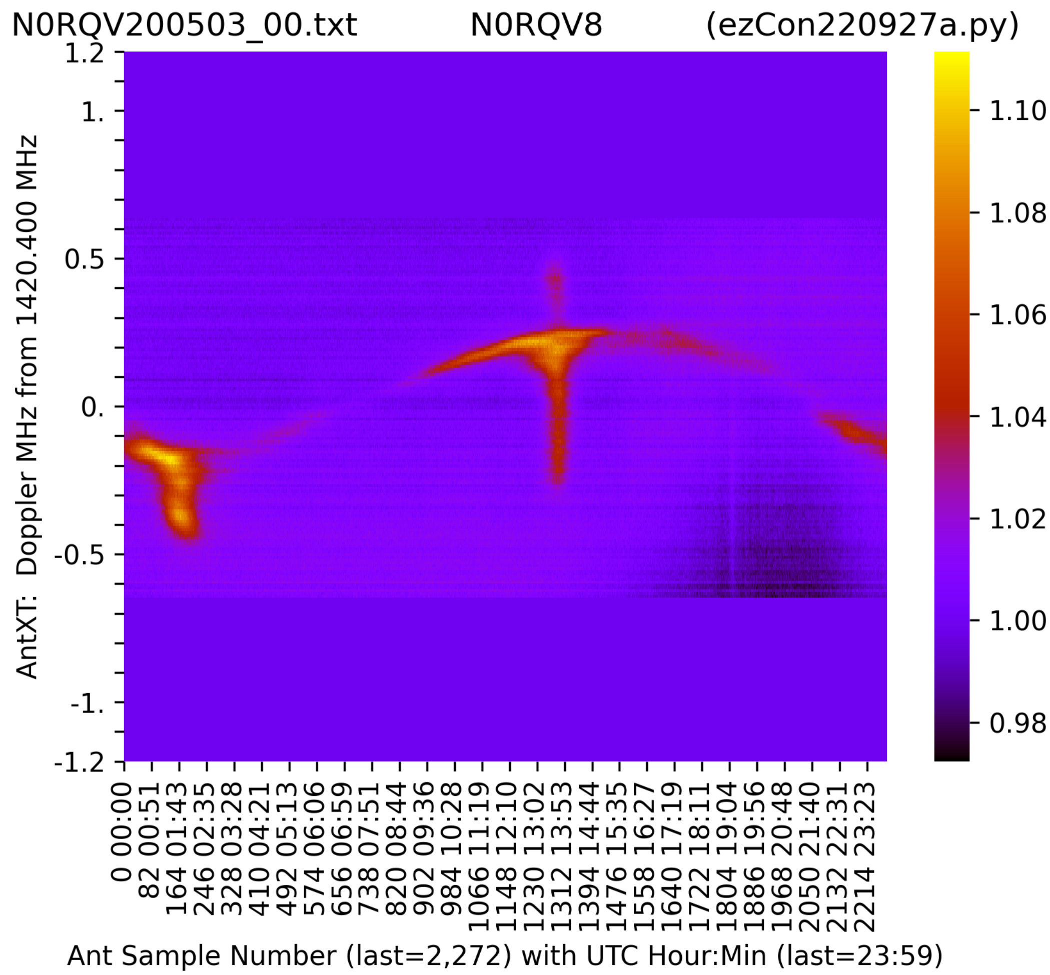
AntX from AntRB

ezCon signal processing can continue with AntX. Above, the AntX input was the the simpler AntB. Using -ezConAntXInput, we continue now with AntRB as the input.

Still standing in that demo1 directory, we enter:

py ..\ezRA\ezCon.py data\N0RQV200503_00.txt -ezConRefMode -760 -ezConAntXInput 3

and ezCon plots are replaced with new ones. The AntX plots may have changed, like AntXT,

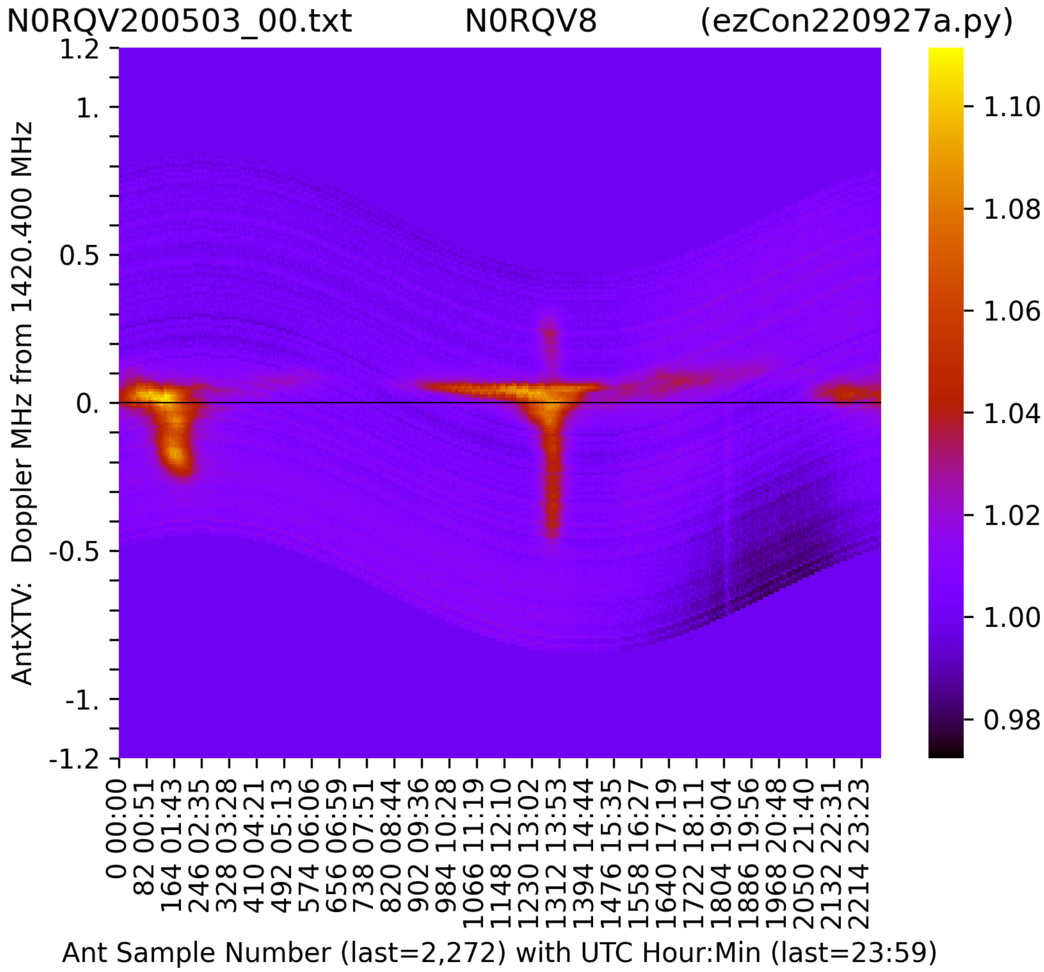


ezCon081antXT.png

("T" added for Trimmed) here displays the samples from either AntRB, after Trimming the low and high frequencies.

Using a single Reference sample, AntXT has improved, with a slightly more uniform background.

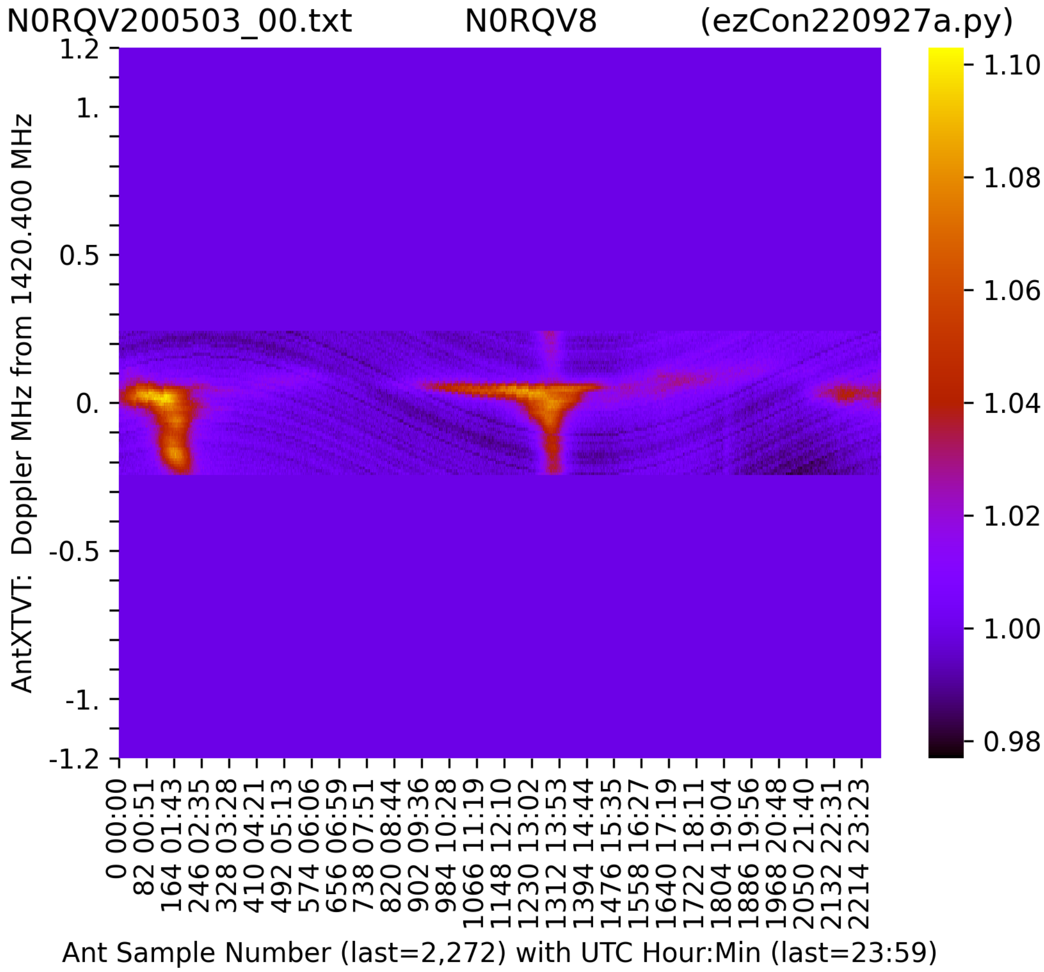
As before, ezCon uses each sample's VLSR value to correct for the Doppler effect, to create the AntXTV signal.



ezCon082antXTV.png
("V" added for Velocity-corrected) displays AntXT samples after Doppler-frequency-correcting with the sample's calculated VLSR.

Using a single Reference sample, AntXTV has improved, with a slightly more uniform background.
The yellow sinusoidal Galactic hydrogen is flattened, to be closer to zero Doppler.
ezCon adds a thin black line at zero Doppler, for comparison.

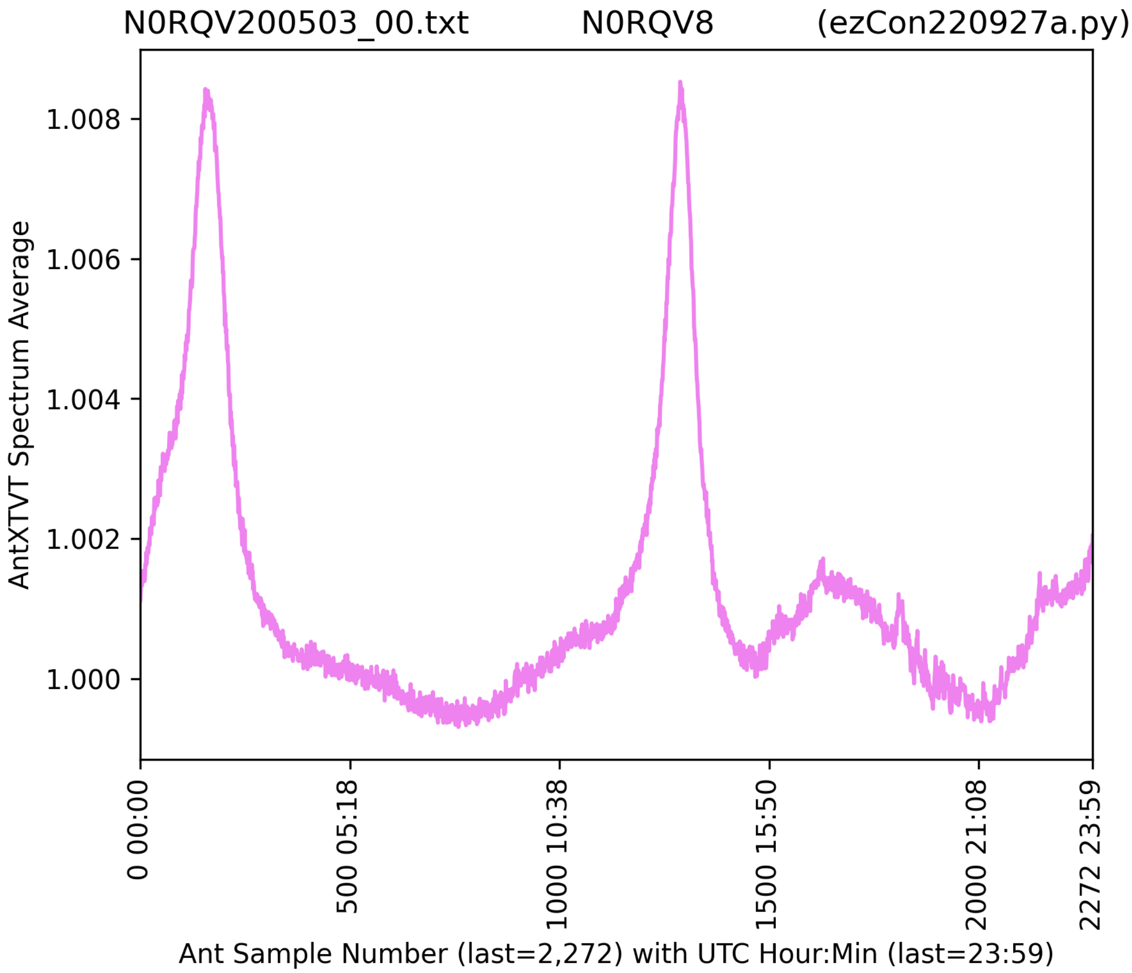
After all that frequency shifting of spectra, the low and high freqBin values of AntXTV look wavy and confusing. Time for another frequency trimming.



ezCon087antXTVT.png
(another “T” added for Trimmed) displays AntXTV samples after Trimming the low and high frequencies.

Using a single Reference sample, AntXTV may have improved, losing background variance.

And the plot of the average of each colorful vertical sample spectrum is improved,



ezCon118antXTVTAvg.png
displays the average value for each AntXTVT sample spectrum (each colorful vertical line of previous plot).

This AntXTVTAvg trace is slightly more bumpy at the lower right, than that of the earlier ezCon116antRBAvg.

In this example, the frequency trimming did not greatly improve this AntXTVT Average signal.

Using Many Reference Samples

It so happens that this drift-scan was actually collected with alternating Reference samples that measured a resistor (instead of the antenna). In the full dataset, one Reference sample (from the resistor) was recorded, then one Antenna sample was recorded, then another Reference sample was recorded, then another Antenna sample was recorded, etc.

We want to better detect Galactic hydrogen by comparing our Antenna signal to a similar signal that does not have Galactic hydrogen (Reference).

On the full data set, we can try choice A of the earlier Reference samples list:

A. Measure a resistor on the antenna side of the first radio amplifier (#10, #20).

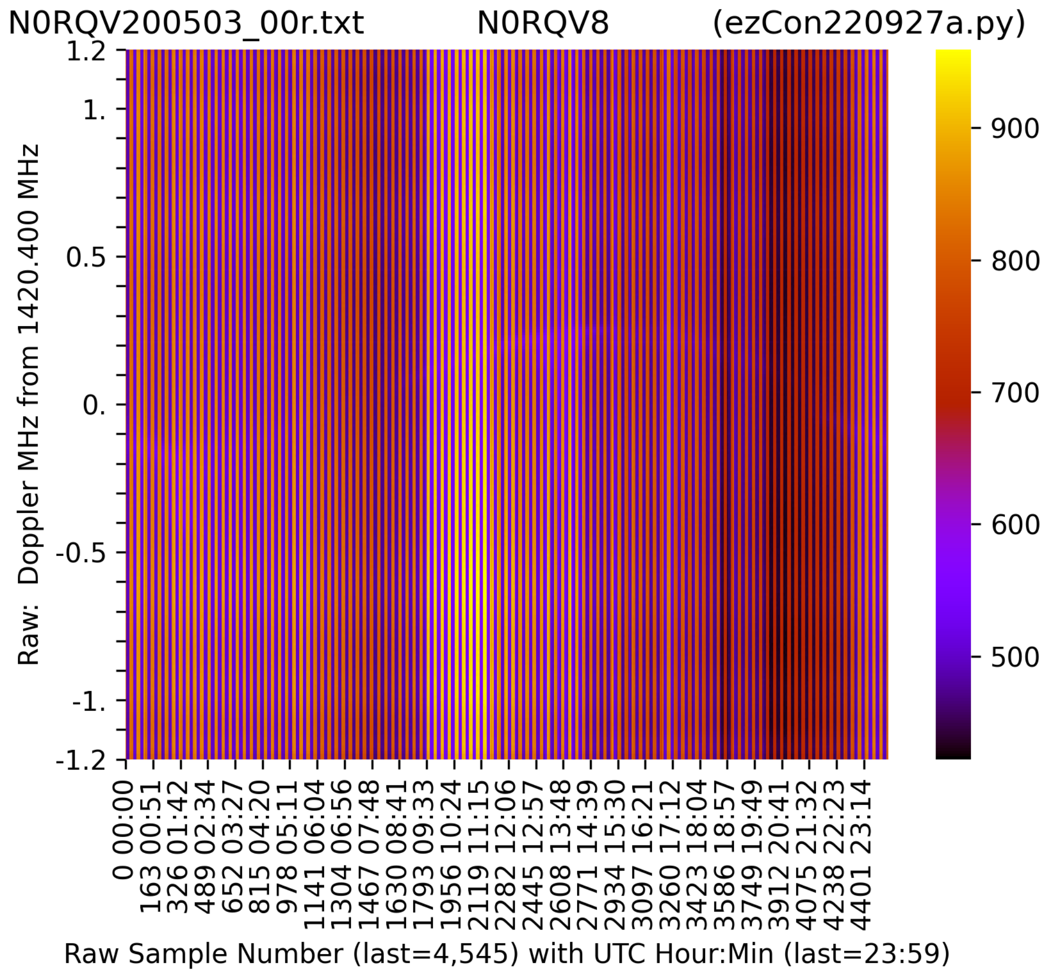
With 0 or 1 Reference samples, the default -ezConAntXInput 0 will choose AntB as the AntX input. With 2 or more Reference samples, the default -ezConAntXInput 0 will choose AntRB as the AntX input.

Using the default -ezConAntXInput 0 and default -ezConRefMode 10, we continue now with AntRB as the input, with the full data set file. Still standing in that demo1 directory, we enter:

```
py  ..\ezRA\ezCon.py  data\N0RQV200525_00r.txt
```

and ezCon plots are replaced with new ones.

Wow, the Raw heatmap plot has changed,



ezCon001raw.png
with the full data set with alternating Reference samples

The sinusoidal Galactic hydrogen on the center frequencies is hidden with lots of thin vertical lines.
The times at the bottom are the same, but the number of samples at the bottom are different.

The earlier demonstration data file was created from this date file (using ezFix -RR 1).
With alternating Reference samples, this file has 4546 Raw samples (first = sample 0, last = sample 4545).
2273 of them are Antenna samples (seen in the earlier plots above, sample 0 and 2272 more).
2273 of them are Reference samples, measuring the resistor.
2273 Ant + 2273 Ref = 4546 Raw samples.

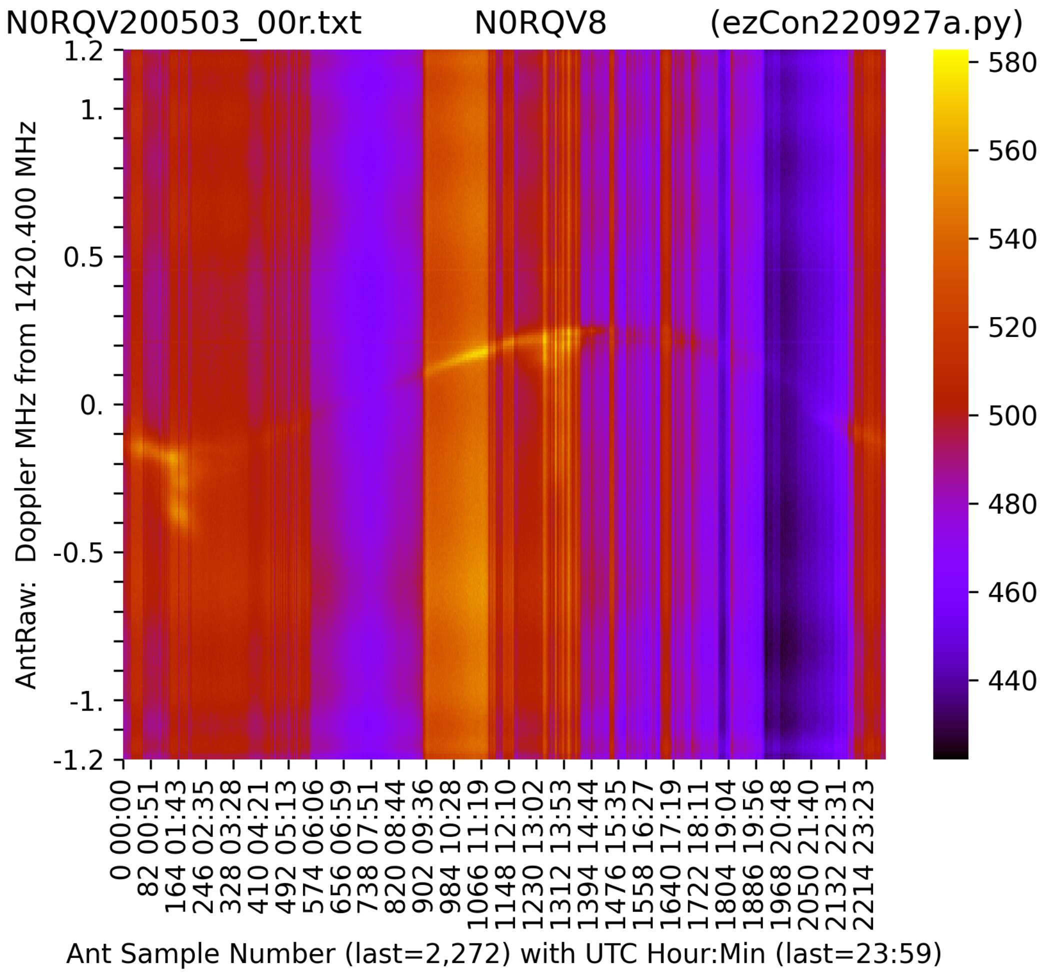
This data file marks the Reference samples with an “R” dataflag.
The .txt file text will start just the same, but some data lines will have extra characters,

```
from ezCol220826a.py
lat 40.3 long -105.1 amsl 1554.0 name N0RQV8
freqMin 1419.2 freqMax 1421.6 freqBinQty 256
az 227.9 el 38.9
# times are in UTC
# gain 49.6
# frequency spectrums of RMS power = sqrt(mean of sum of squares)
2022-09-03T00:00:06 206.197 206.197 207.678 209.662 … 210.365
2022-09-03T00:00:25 345.176 345.176 347.336 348.813 … 348.021 R
2022-09-03T00:00:44 207.330 207.330 207.418 209.639 … 209.607
2022-09-03T00:01:03 344.002 344.002 345.693 348.736 … 349.276 R
…
```

This data file happens to start with an Antenna sample.
The data lines are long, with at least 257 words, but if it ends with a number and then a data flag character “R”, then it is a Reference sample.
If a data line ends with just a number, and no “R” data flag character, then it is an Antenna sample.

Perhaps a data flag of “H” or “X” will indicate a Horizontal polarization sample.
Perhaps a data flag of “V” or “Y” will indicate a Vertical polarization sample.
Perhaps a data flag of “L” or “-’” will indicate a Left-handed circular polarization (LHCP) sample.
Perhaps a data flag of “C” or “+” will indicate a Clockwise Right-handed circular polarization (RHCP) sample.
Dataflags must be grouped into one word, but can be in any order.

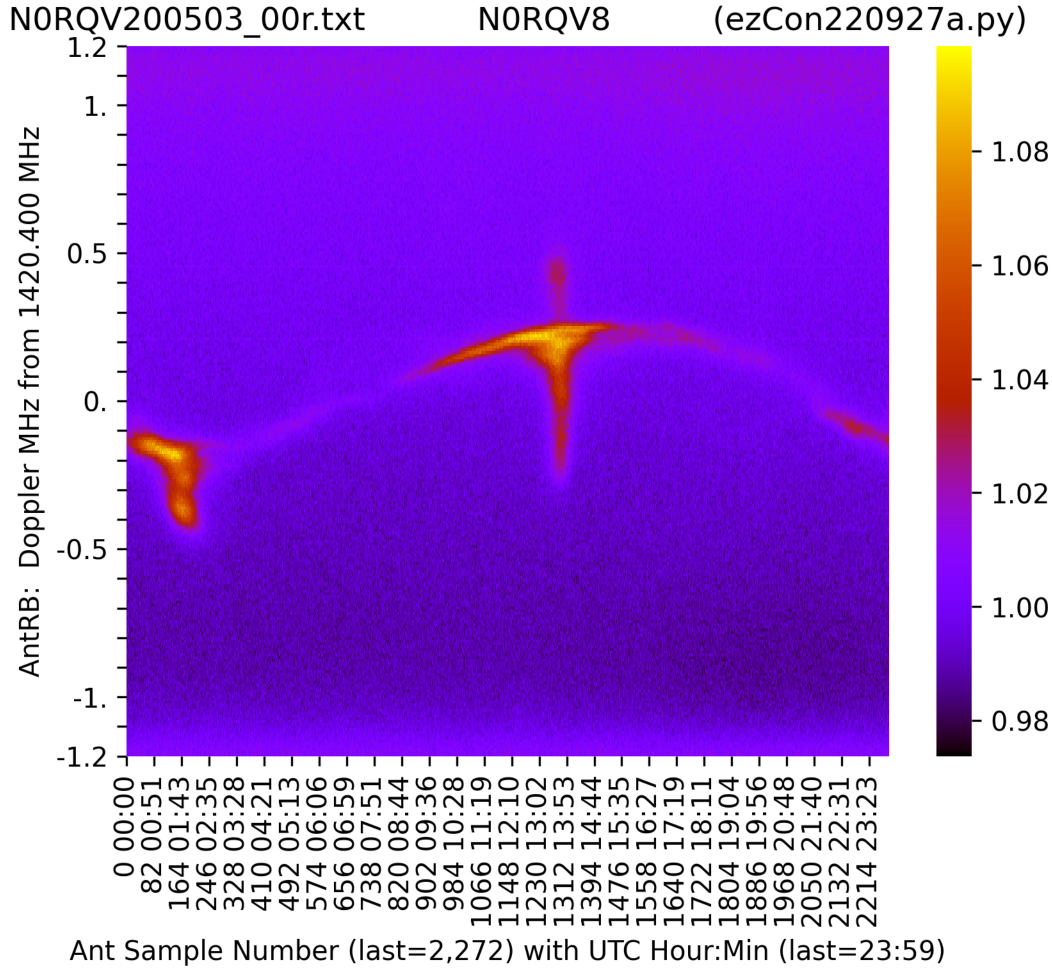
There can be more than one Antenna sample between 2 Reference samples.



ezCon002antRaw.png
displays only Antenna (Ant) samples, and no Reference (Ref) samples.

This plot looks just like the earlier ezCon002antRaw plot, from many pages back.
Note at the bottom, the number of Ant samples is about half of the Raw samples.

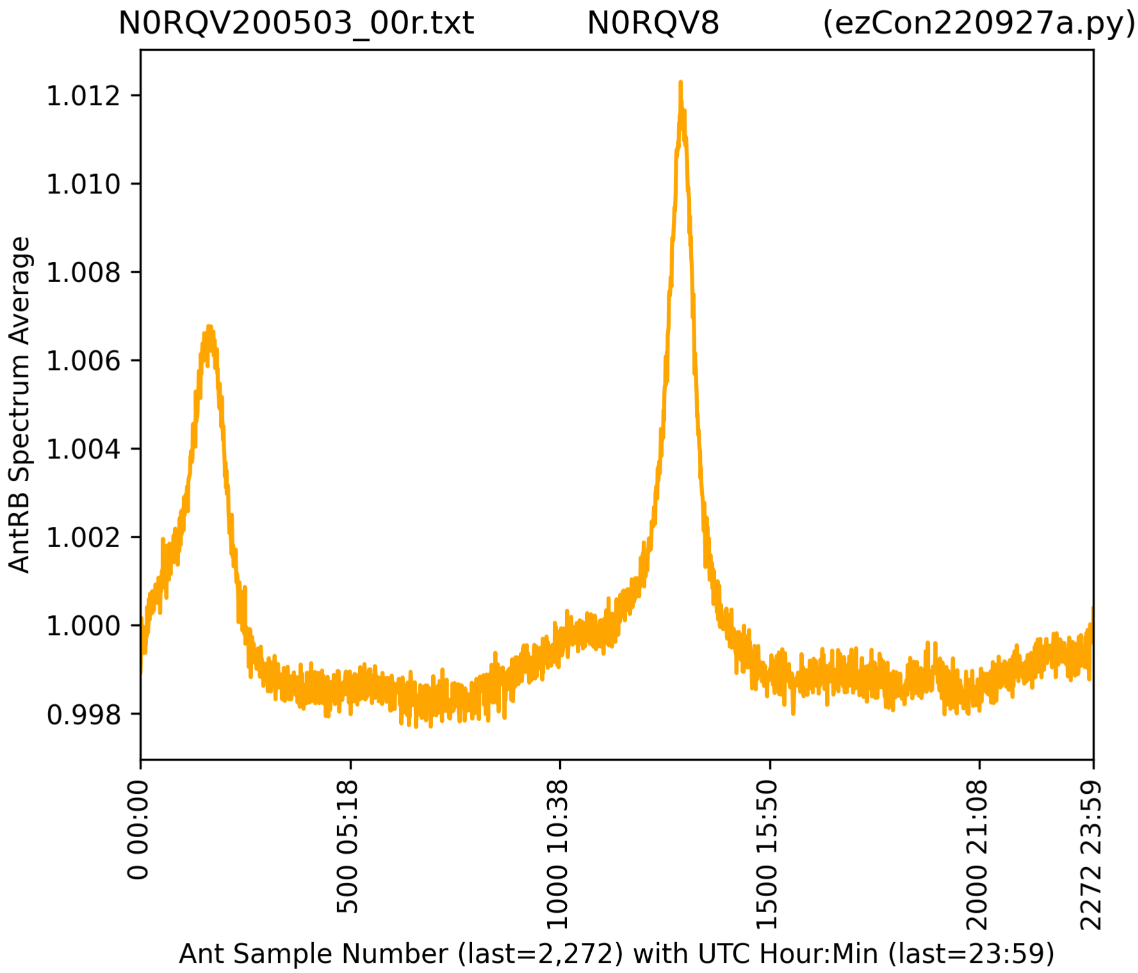
But the plots that use the Reference samples may now be changed …



ezCon067antRB.png
now using alternating Reference samples.

Using frequent (alternating) Reference samples, the background looks better, smoother.
Yes, the hydrogen appears to be weakest at about sample 760.
More frequent Reference samples is usually better, just like collecting more data is usually better.

And here is the plot of the average of each colorful vertical sample spectrum,

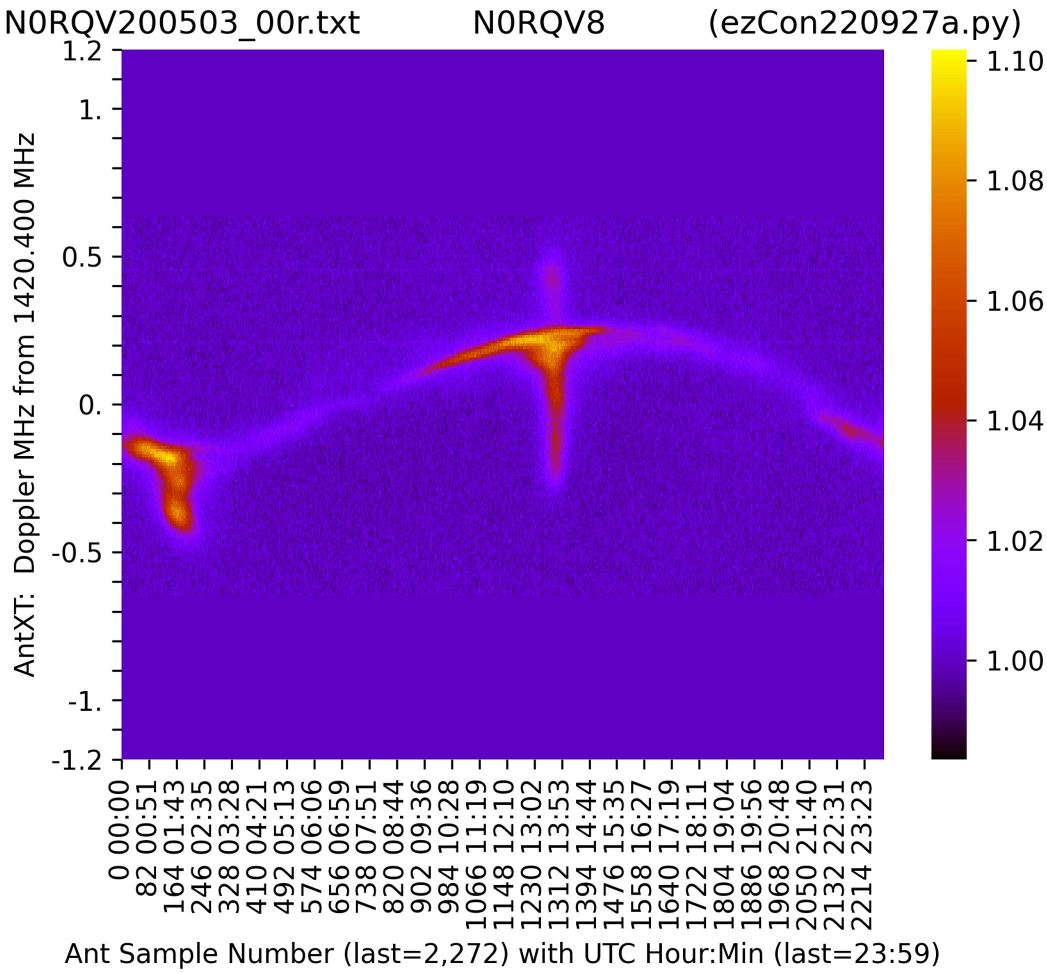


ezCon116antRBAvg.png
displays the average value for each AntRB sample spectrum (each colorful vertical line of previous plot).

This AntRB Average trace at the bottom, is smoother than that of the earlier ezCon116antRBAvg plots.

Yes, the hydrogen appears to be weakest at about sample 760.

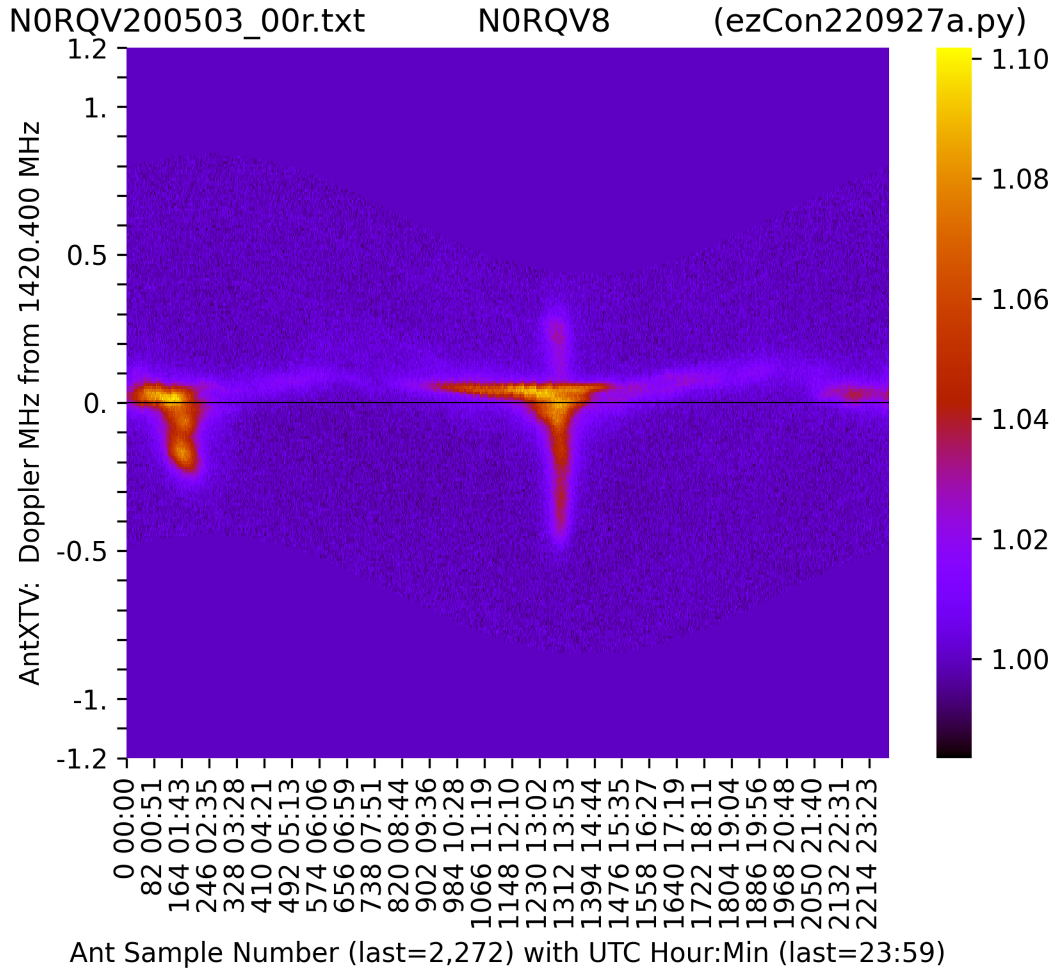
The AntX plots look better also …



ezCon081antXT.png
("T" added for Trimmed) here displays the samples from either AntRB, after Trimming the low and high frequencies.

Using alternating Reference samples, AntXT has improved, losing much of the horizontal banding.

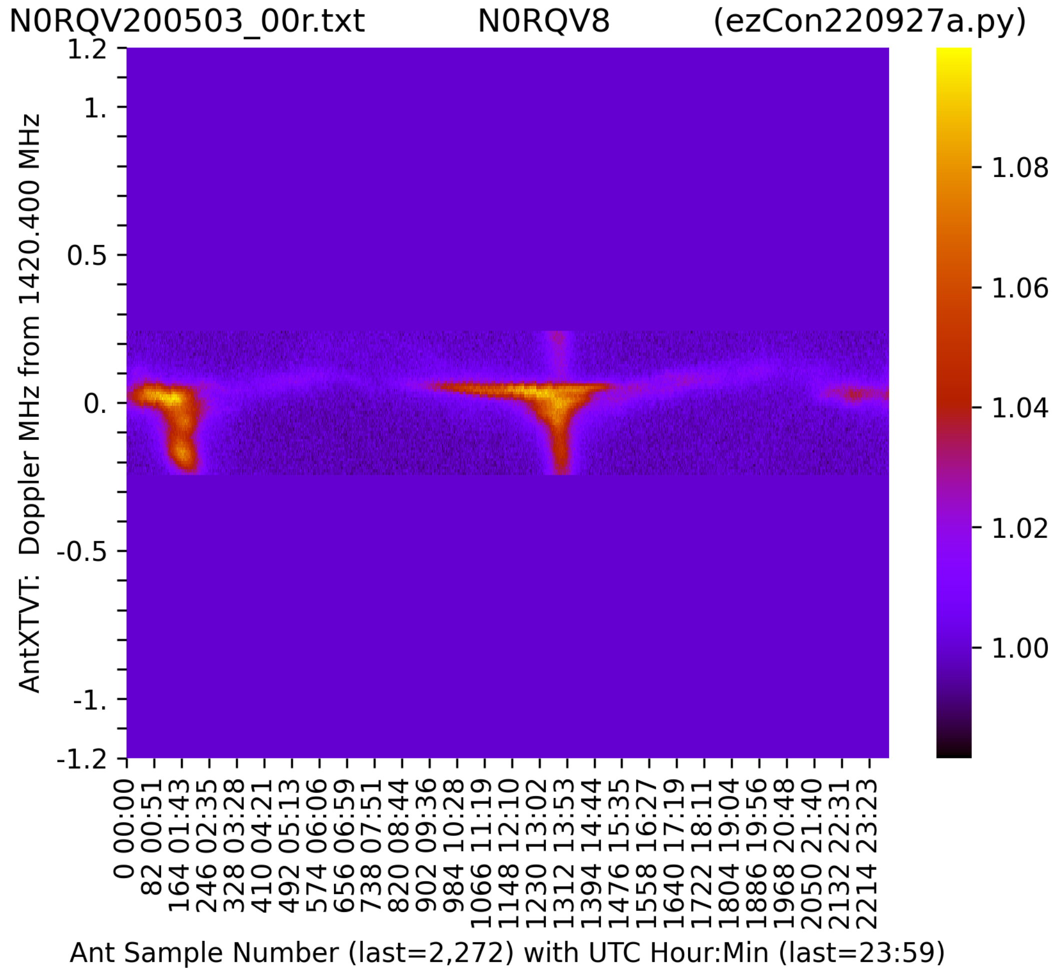
As before, ezCon uses each sample's VLSR value to correct for the Doppler effect, to create the AntXTV signal.



ezCon082antXTV.png
("V" added for Velocity-corrected) displays AntXT samples after Doppler-frequency-correcting with the sample's calculated VLSR.

Using alternating Reference samples, AntXTV has improved, losing much of the horizontal banding.
The yellow sinusoidal Galactic hydrogen is flattened, to be closer to zero Doppler.
ezCon adds a thin black line at zero Doppler, for comparison.
This AntXTV signal is used to study of the true relative velocity of the Galactic hydrogen.

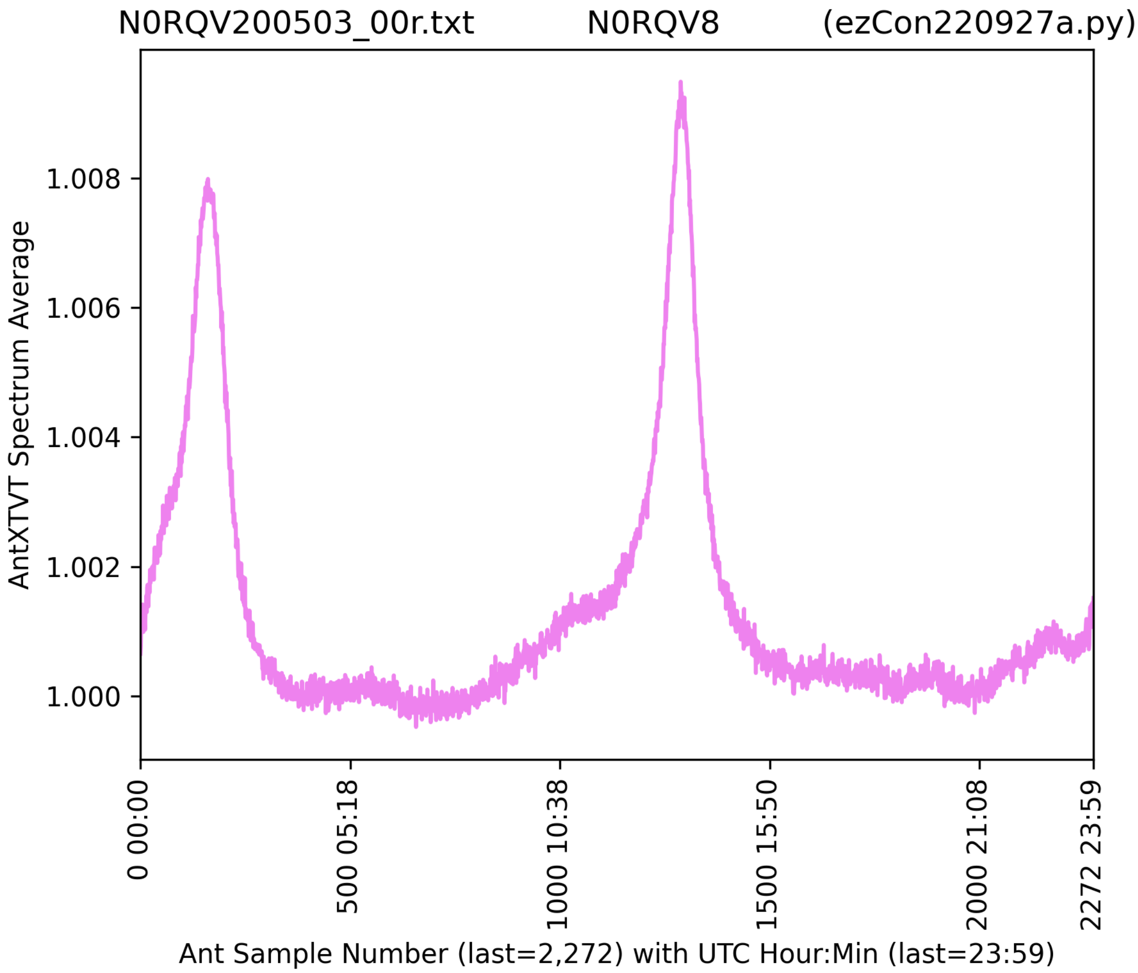
After all that frequency shifting of spectra, the low and high freqBin values of AntXTV look wavy and confusing. Time for another frequency trimming.



ezCon087antXTVT.png
(another "T" added for Trimmed) displays AntXTV samples after Trimming the low and high frequencies.

Using alternating Reference samples, AntXTV has improved, losing much of the horizontal banding.

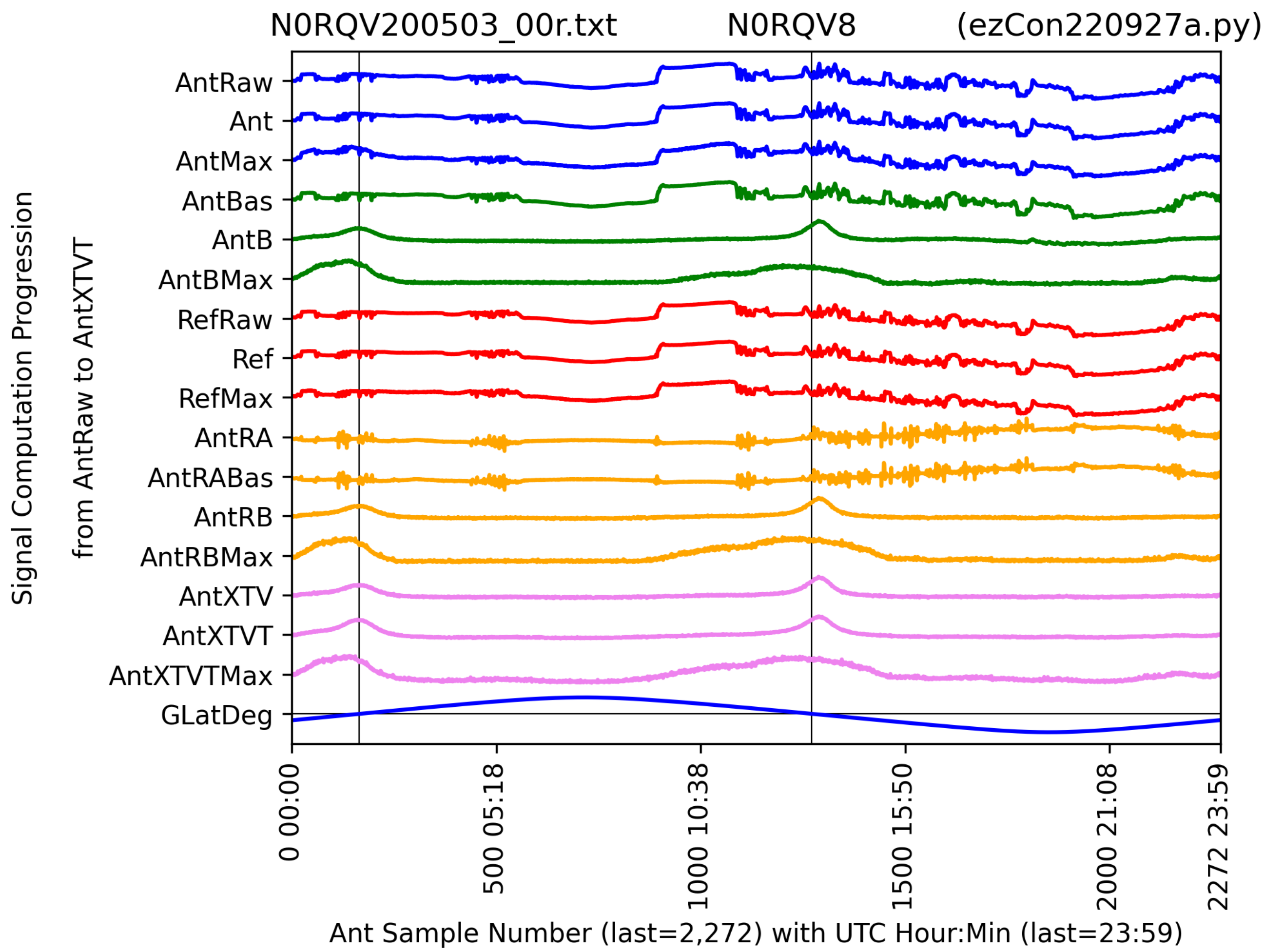
And the plot of the average of each colorful vertical sample spectrum,



ezCon118antXTVTAvg.png
displays the average value for each AntXTVT sample spectrum (each colorful vertical line of previous plot).

This AntXTVTAvg trace is almost the same as the earlier ezCon116antRBAvg.

Wow, so many signals from one antenna. Here is a summary plot of the Signal Computation Progression,



ezCon191sigProg.png
Signal Computation Progression

The Signal Computation Progression plot begins with the averages of Raw Ant samples, and after the AntHide (freqBin masking, for annoying "birdies") and AntRfi filters, followed by the AntB signals. Then come the Raw Ref samples, and after the RefHide (freqBin masking) and RefRfi filters, AntRA and AntX signals, and finally the Galactic Latitude with a zero line.

Hopefully, apparently random drift-scan signals on the top develop into clear bumps showing Galactic hydrogen energy.

With sample timestamps increasing to the right, the Galactic Latitude trace crossing its zero line down to the right (Galactic north to south), marks a drift-scan antenna bearing passing through the Galactic plane near the Galactic center (the strong "inner arm"). GLat crossing zero up to the right (Galactic south to north), marks a weaker "outer arm" Galactic plane crossing. These Galactic crossings are marked with a thin black vertical line. The spectra of these crossings are collected in *Gal.npz files, for collective processing with the ezGal program.

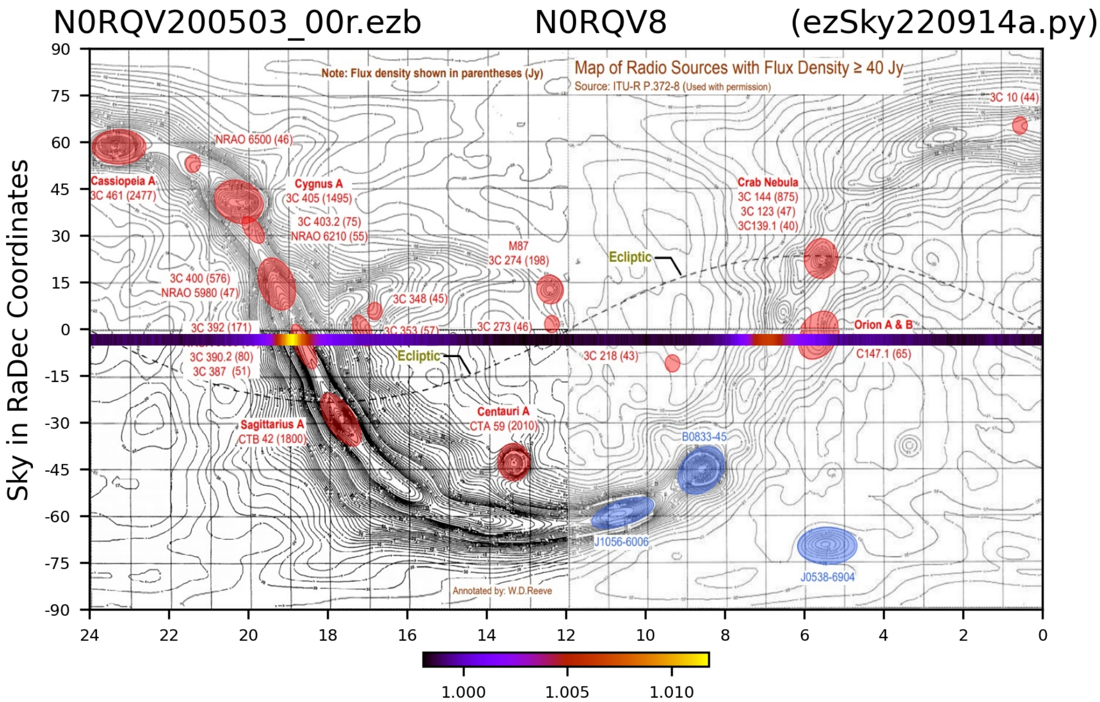
This plot shows the AntRB signal (AntRBAvg) looks to be good for plotting on sky maps, using ezSky.

The AntRBAvg signal (.ezb file column 16) is popular for plotting on sky maps, using ezSky.

Still standing in that demo1 directory, we enter:

```
     py  ..\ezRA\ezSky.py  N0RQV200503_00r.ezb  -ezSkyInput  16
```

There are now more plots at our feet.



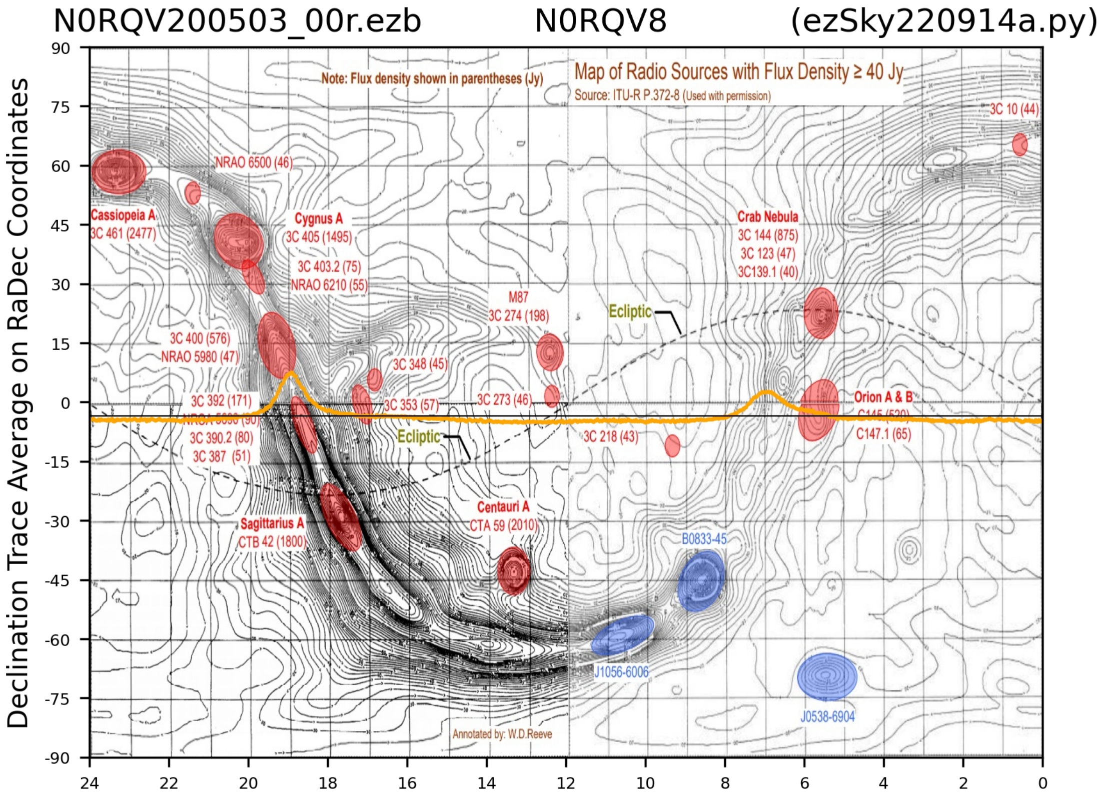
ezSky401RBPF_16AntRBAvg.png
displays the AntRBAvg signal on the RaDec sky, with larger values as brighter colors.

This example's two bumps of the chosen AntRBAvg input signal show up as brighter spots on the horizontal trace, and the brighter spots fairly align with the Galaxy.

Bright spots may not align with the Galaxy if data timestamps, or antenna location, or antenna bearing were wrong. The whole horizontal trace might be shifted to the left or right, or up or down.

Raising the antenna's elevation, and then collecting data from another 24-hour drift-scan, would create another .ezb condensed data file, to provide a similar, but higher horizontal declination trace across the sky. By plotting all the .ezb files together, we begin to plot the 1420 MHz radio sky.

Or perhaps you prefer displaying the AntRBAvg data this way, with the ezSky200RP_* plots, showing vertical offset from a thin black Declination line.



ezSky200RP_16AntRBAvg.png
with the same AntRBAvg signal on the RaDec sky, with larger values displayed as greater vertical offsets from a thin black Declination line.

ezCol with Reference Frequency

We start another example, where ezCol generates alternating Reference samples, from a nearby frequency. Samples from near 1423 MHz should have no Galactic hydrogen.

In that earlier “radioSky” directory, create a new directory called “demo2”.
In that demo2 directory, ezCol will automatically create a directory called “data”.
The ezRA set of program files are still available in the neighboring “ezRA” directory.

Change Directory to that “demo2” directory, perhaps with the command

```
cd  ..\demo2
```

Standing in that demo2 directory, we enter:

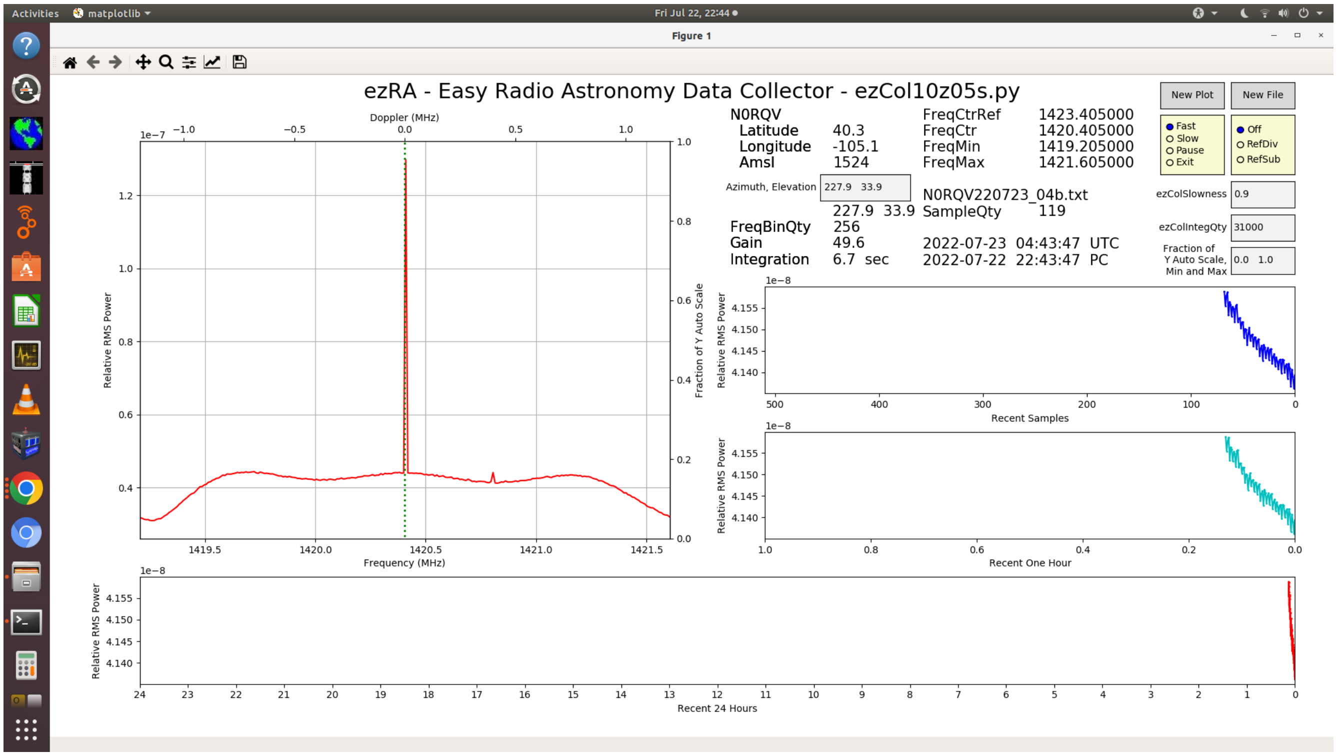
```
py  ..\ezRA\ezCol.py  -h
```

for the ezCol usage help text.

Last time, we used some ezCol default values.
This time, we use a slightly longer command line to also create Reference samples from a nearby 1423.405 MHz.
For this example, after even deeper consideration, we enter this longer command, all on one line:

```
py  ..\ezRA\ezCol.py
     -ezRAObsLat  40.4  -ezRAObsLon  -105.1  -ezRAObsAmsl  1554
     -ezRAObsName  N0RQV
     -ezConAzimuth 227.9  -ezConElevation  38.9
     -ezColCenterFreqRef  1423.405000
```

Eventually a graphics window appears:



After a few samples, we can see the samples alternate between low and high frequencies, between Reference (Ref) and Antenna (Ant) frequencies. Watch the center frequency change, shown at the bottom center of the upper left spectrum graph. In the text in the upper right, the “FreqCtrRef” text now says “1423.405000”. The nearby “SampleQty” numbers, for the Reference samples, will have a trailing “R” data flag.

First, a Ref sample is recorded, then an Ant sample is recorded, then a Ref sample, etc.
A Ref sample may have a higher value spectrum, or a lower value spectrum, than its neighboring Ant sample. And that may be a function of temperature ?

We expect to see hydrogen emission in only the Ant samples, from near 1420 MHz.

Now with Ref samples available, we have 2 new options for the upper left spectrum graph.
Upon left-mouse-clicking the “RefDiv” circle in the right yellow box in the upper right, the upper left spectrum graph eventually changes to show each Ant sample value Divided by the corresponding most recent Ref sample value.   Hydrogen emission will appear as a larger peak near the center frequency.
Each Ant sample is being compared to its most recent Ref sample. The graph center frequency now remains unchanging as the Ant center frequency.

Left-mouse-clicking the “RefSub” circle in the upper right yellow box, does likewise, but replacing the Division, with Ant Subtracting Ref.
Multiple mouse clicks on the little circles may help to attract the Dashboard’s slow attention.

The left yellow box in the upper right has 4 options.
“Fast” focuses on data collection, with little time applied to the Dashboard.
“Slow” focuses less on data collection, allowing more time applied to the Dashboard.
“Pause” temporarily stops all data collection, allowing even more time applied to the Dashboard, or to allow for non-recorded system changes.
“Exit” eventually stops the ezCol program.
Multiple mouse clicks on the little circles may help to attract the Dashboard’s attention.

The ezColSlowness upper right box allows entering a number to define the “Slow” option, with a value between 0.0 and 1.0 . A higher value will make the Dashboard slower, allocating more of the PC’s resources on creating data samples faster.

The ezColIntegQty upper right box allows entering a number to define the Quantity of radio readings to Integrate into each sample spectrum. Increasing this value increases the average time between recorded samples, determined by your PC’s speed.

On the right side of the upper left spectrum graph is a second Y-Scale. The “Fraction of Y Auto Scale, Min and Max” upper right box allows entering 2 numbers (each between 0.0 and 1.0) to define the spectrum graph Y-Scale minimum and maximum. This allows zooming in on the interesting part of the graph.

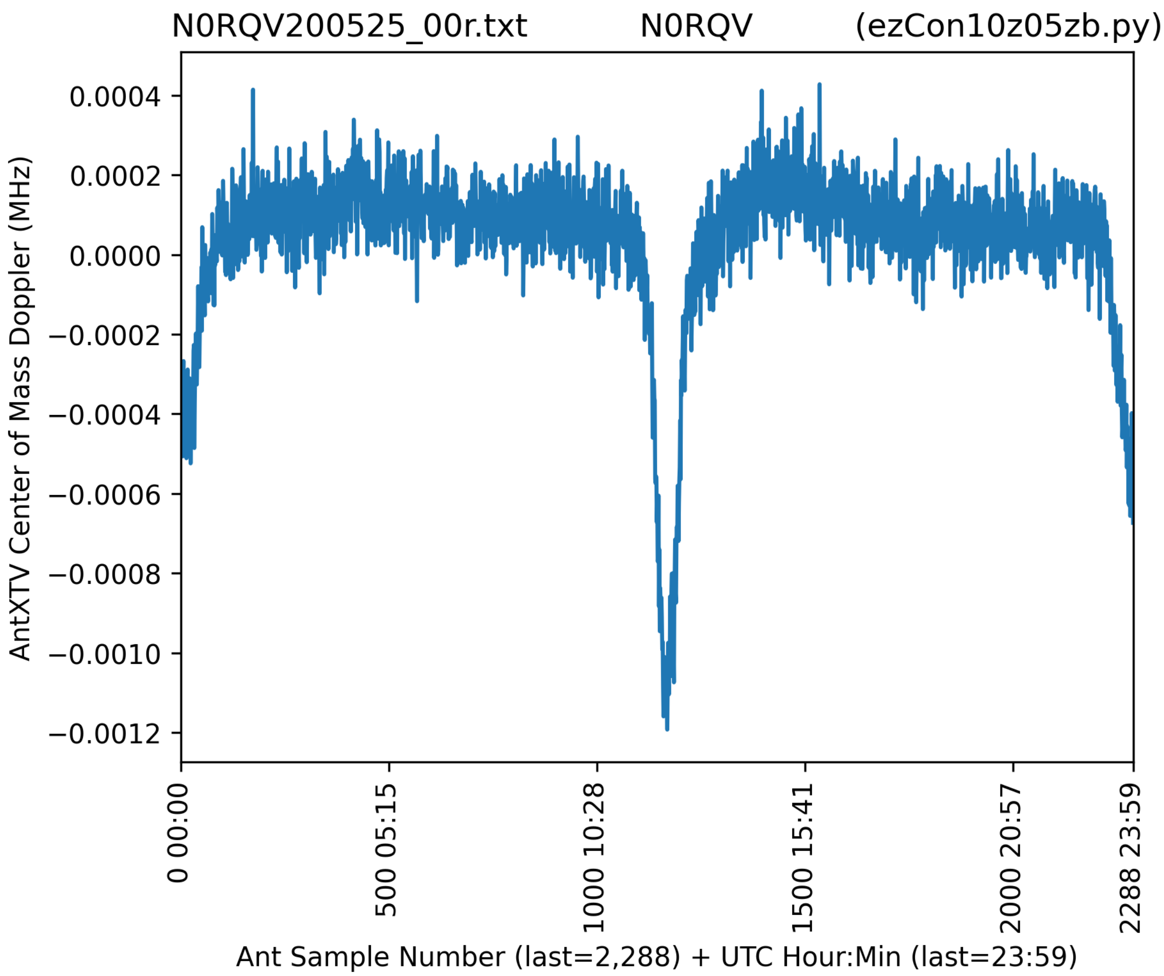
The “New Plot” upper right button will clear the 3 stripchart plots, which will reset the vertical autoscaling.

The “New File” upper right button will likewise clear the 3 stripchart plots, and it will also start a new recorded data file. The new data filename will be displayed in the center of the upper right text section, just above the “SampleQty”. At the start of a data recording session, the receiver electronics eventually heats up to its operating temperature. During this slow temperature increase, the system gain (amplification) decreases, and the recorded signal values slowly decrease. This button allows discarding a recording of such system warm-up gain variance, by quickly starting a “New File”.

The “Azimuth, Elevation” upper right box allows entering 2 numbers (0.0 to 360.0, and 0.0 to 180.0) to define the recorded Azimuth and Elevation. The current values are displayed just below the box.

Center-of-Mass of the DOPpler (CMDop)

The ezCon309antXTCmDop.png plot shows the AntXT signal spectra change in frequency, shape, and intensity. That signal's Center-of-Mass is recorded in the AntXTCMDop signal. That AntXTCMDop signal can be corrected later with the VLSR signal.



ezCon309antXTCmDop.png
displays the Center-of-Mass of the DOPpler (CMDop) intensity for each AntXT signal spectrum.

https://en.wikipedia.org/wiki/Center_of_mass
says the Center-of-Mass calculation should be related to the sum of (distance-from-zero-Doppler * intensity), over all intensity values, for each AntXT signal spectrum.

Math

For every data sample, ezCon calculates the astronomy math, from

Latitude (Lat), Longitude (Lon), and Altitude above Mean Sea Level (AMSL),
Azimuth (Az), and Elevation (El),
Modified Julian Day (MJD) time values

into

Right Ascension (RA), and Declination (Dec),
Galactic Latitude (GLat), and Galactic Longitude (Glon),
Velocity from the Local Standard of Rest (VLSR).

Modified Julian Day (MJD) time is the fractional number of days since the start of November 17 1858, in UTC (Coordinated Universal Time).
https://en.wikipedia.org/wiki/Julian_day
https://en.wikipedia.org/wiki/Coordinated_Universal_Time

Count, Spare1, and Spare2

Currently, a sample's Count value is always 1.0 .
In the future, this may be used to combine data files, and then it may be helpful to know the number of files already combined ?
Currently, a sample's Spare1 and Spare2 values are always 0.0 .

Galaxy Crossings

For a given drift-scan unchanging antenna azimuth and elevation, the antenna will typically record 2 Galactic plane (GLat = 0) crossings in 24 hours, one closer to the Galactic Center (where GLon=0), and one farther away.
ezCon records the spectra of these Galactic Crossings for later study.
If any Galactic Crossings, their velocity spectra and quantity are written to a related *Gal.npz data file, for later ezGal program combined file analysis, to explore Galactic rotation and Galactic Arms.

The command line argument -ezConGalCrossingGLat optionally defines "close to Galactic plane crossing" in Galactic Latitude degrees. Perhaps to include crossings only within 5 degrees of GLat=0.
The command line arguments -ezConGalGLonEdgeFrac and -ezConGalGLonEdgeLevel optionally define "edge of spectrum".

Blah, blah, blah

##
##
##
##
##
##
##
##
##
##
##
##

End of writing progress.

What follows are incomplete sections, but may be useful.

Then maybe a successful 1423 MHz reference example, to create an AntRB.
Show temp inversion.

Eventually a successful Dicke switch resistor reference example, to create an AntRB.
Describe hardware.

Describe the many uses of ezFix.

Describe the many options of ezCon, using full ezCon_Data_Flow block diagram.

Describe the layers of options available with ezDefault.txt files.

ezRA option tricks, and tab-completion.

Explain every plot.

Windows installation, ezCol needs, and USB relay.
Ubuntu installation, ezCol needs, and USB relay.

ezCon filters.

##
##
##
##

##

There are many other ezCon display command line arguments.
Cut and paste their spelling from USAGE.

For USAGE, run with “-help”,
  py ezCon.py -help
Eventually available on all ezRA programs.

For USAGE, run with no valid command line arguments,
  py ezCon.py
Eventually available on all ezRA programs, except ezCol .

ezCon command line arguments and files may be in any order.
Including "-h" on the end, for the USAGE help.

Trick to preserve all that command line typing:
To temporarily ignore one ezCon command line argument, start that one word with "-ezez".
Remove spaces before the values to temporarily combine words, or preface values with "-ezez".
Eventually available on all ezRA programs ?
ezCol controlled a USB relay to control bias-tee coax voltage.
