

MOUNTAIN SPARK GAPS

**NPARC—The Radio Club for the
Watchung Mountain Area**



**Website: <http://www.nparc.org>
Club Calls: N2XJ, W2FMI
Facebook: New Providence Amateur Radio Club
(NPARC)**

VOLUME 52 NO. 2 February 2017

UPCOMING EVENTS

Regular Meetings

**3/13 & 3/27
Mondays at 7:30
DeCorso Community Center**

Meeting Schedule

Regular Meeting: 7:30—9:00 PM
2nd Monday of each month at the
NP Senior & Adult Center
15 East Forth Street
New Providence

Informal Project Meeting: 7:30—9:00
PM

4th Monday of each month
Same location

Everyone is Welcome
If a normal meeting night is a holiday,
we usually meet the following night.
Call one of the contacts below
or check the web site

Club Officers for 2017

President: W2PTP Paul Wolfmeyer
201-404-6914
Vice President: K2GLS Bob Willis
973-543-2454
Secretary: K2AL Al Hanzl
908-872-5021
Treasurer: K2YG Dave Barr
908-277-4283
Activities: AC2GL David Hartman
908-665-1419

On the Air Activities

Club Operating Frequency
145.750 MHz FM Simplex

Sunday Night Phone Net

Murray Hill Repeater (W2LI) at 9:00 PM
Transmit on 147.855 MHz
With PL tone of 141.3 Hz
Receive on 147.255 MHz
Net Control K2AL

Club Internet Address

Website: <http://www.nparc.org>
Webmaster K2MUN David Berkley
Reflector: nparc@mailman.qth.net
Contact K2UI, Jim

MOUNTAIN SPARK GAPS

Published Monthly by NPARC, Inc.
The Watchung Mountain Area Radio Club
P.O. Box 813
New Providence, NJ 07974
©NPARC 2010 All Rights Reserved
Editor: K2EZR Frank McAneny
Contributing Editors:
WB2QOO Rick Anderson
W2PTP Paul Wolfmeyer
K2UI Jim Stekas

Climatological Data for New Providence for January 2017

The following information is provided by
Rick, WB2QOO, who has been recording
daily weather events at his station for the
past 35 years.

TEMPERATURE -

Maximum temperature this January, 62 deg. F
(January 12)

Last January (2016) maximum was 60 deg.
F.

Average Maximum temperature this January,
46.7 deg. F

Minimum temperature this January, 6 deg. F
(January 9)

Last January (2016) minimum was 6 deg. F.

Average Minimum temperature this January,
28.7 deg. F

Minimum diurnal temperature range, 3 deg.
(39-36 deg.) 1/17

Maximum diurnal temperature range, 28 deg.
(58-30 deg.) 1/13

Average temperature this January, 35.2 deg.
F

Average temperature last January, 30.6 deg. F

PRECIPITATION -

Total precipitation this January - 4.5"
rain/melted snow; 5.3" snow

Total precipitation last January - 5.19"
rain/melted snow; 22.35" snow;

Maximum one day precip. event this January -
January 7, 3.0" snow; January 23, 1.2" rain
Measurable rain fell on 13 days this Janu-
ary, 3 days last January.

Measurable snow fell on 4 days this January,
3 days last January.

YTD Precipitation - 4.5" (includes rain +
melted snow, as of 1/31/17)

=====

Rick Anderson

2/10/17

243 Mountain Ave.

New Providence, NJ

(908) 464-8912

rick243@comcast.net

Lat = 40 degrees, 41.7 minutes North

Long = 74 degrees, 23.4 minutes West

Elevation: 380 ft.

CoCoRaHS Network Station #NJ-UN-10

President's Column February 2017

Our annual auction certainly seemed to go well! Camaraderie was good, attendance was up, the club made about \$600, and everyone pitched in on their assigned tasks. In fact, it is the latter I am always impressed with—members do what's asked of them, making most tasks manageable and a good team-work experience. So thanks to all!!

At Monday's meeting, Rob Roschewsk, KA2PBT, who is running for Section Manager, gave us a short program on his background. Of particular interest was his work in involving youth in ham radio. And then Brian, KA2MPG provided a video on Hallicrafters efforts in support of WWII—so we had a range from essential vacuum tube equipment to robotics.

It was also nice to have new members and prospective members present: Phil, KC2ONL, (who I learned after the meeting just passed his General), Bill KD2JRI, who joined at the auction, and Henry, who is studying for his Tech. Your Executive Committee—the officers—met also before the meeting. Of particular note: if you have been missing meetings due to night driving or other transportation difficulties, maybe we can arrange some transportation assistance from a nearby member. If you fall in this situation, give Bob Willis, K2GLS, a call (973-543-2454) to begin to explore possibilities

At our next meeting (March 13) Bill Hudzig W2UDT will give a program on QSLing. He previously brought to our attention how, as an "ARRL affiliated club", we could "pool" and sort our outgoing QSLs to the bureau and save substantially over the new fee structure versus sending the cards as individuals. We'll talk more about it at the next meeting.

73 for now
Wolf
W2PTP
201-404-6914 or W2PTP@arrl.net

NPARC Auction 2017
Some of the photos taken by Jon, AE2JP
The rest will be in March Spark Gaps



Registration



Some of the "Good Stuff"



Pay the tab



Dinner

Antenna Modeling with Linux

Jim Stekas - K2UI

Back in the spring of 1983 I bought my first computer, an IBM PC with 64KB of RAM and a 320kB 5.25in floppy. For another \$500 I expanded to 256kB of RAM and two floppies. Total investment, about \$3500. Upgrading to DOS 2.0 increased floppy capacity to 360kB and added directories (folders), and with a few simple environment settings I could get DOS to use forward slashes in file paths and specify a PATH to allow my programs to be found. If I didn't look too closely, I could almost believe I was logged into a Unix system at work and not sitting at my PC at home.

On a single bootable floppy I had DOS, Lotus 123, edix (my editor), Turbo Pascal (development), nroff with mm (the Bell Labs "Office suite" of the day), and 120kB left over for whatever. When Windows 3.1 came out and I upgraded to a 33Mhz 486 machine I couldn't imagine how things could get any better. But they infinitely better when I started working on Sun workstations in a networked Unix environment and Windows was relegated to the occasional Power Point presentation.

These days LibreOffice offers Linux users an office suite almost as capable, bloated, and buggy as MS Office that torments their Windows colleagues. But Linux isn't great because it's a free alternative to Windows with a bunch of me too GUI applications. Linux is great because it is built upon the powerful Unix architecture and tools that have been getting sh*t done for almost 40 years: grep, sed, awk, sort, join, ksh, bash, cat, test, mknod, ln, make, etc. Yes, these are old tools, but so are the hammer, socket wrench, pliers, and grid dip meter.

This article will review some of the antenna modeling tools that are available in Linux and provide an roughly equivalent capability to EZNEC in a Windows environment (minus the segment limit). Then we'll take a look at how to combine Unix and NEC into a powerful analysis environment far beyond what is possible in Windows.

Numerical Electromagnetics Code

Almost all antenna modeling software used today traces its lineage to NEC, the Numerical Electromagnetics Code developed in the 1970's at Livermore Labs. The NEC was coded in FORTRAN to run efficiently on large mainframes and supercomputers. Input to the program was provided on 80-column punch cards in a very rigid format. Output came out as a series of large tables on 135 column line-printers with 14x11 inch tractor fed paper. (Ironically, that old computer paper was ideal for solving E&M problems with a pencil.)

Sometime in the 1980's the FORTRAN source code for the second revision of the program, NEC-2, was released and it slowly made its way onto personal computers. The first PC version was MININEC, basically a scale down rewrite in BASIC. Gradually large chunks code got ported to C and an almost full blown version of NEC found its way onto the PC and eventually into user friendly Windows formats. Today, EZNEC from W7EL has become is defacto standard for radio amateurs. For \$100 you get NEC in a nice point-and-click wrapper with beautiful graphics. EZNEC will accept antenna definitions NEC input format, but once you've entered data into EZNEC it will remain in EZNEC's proprietary format.

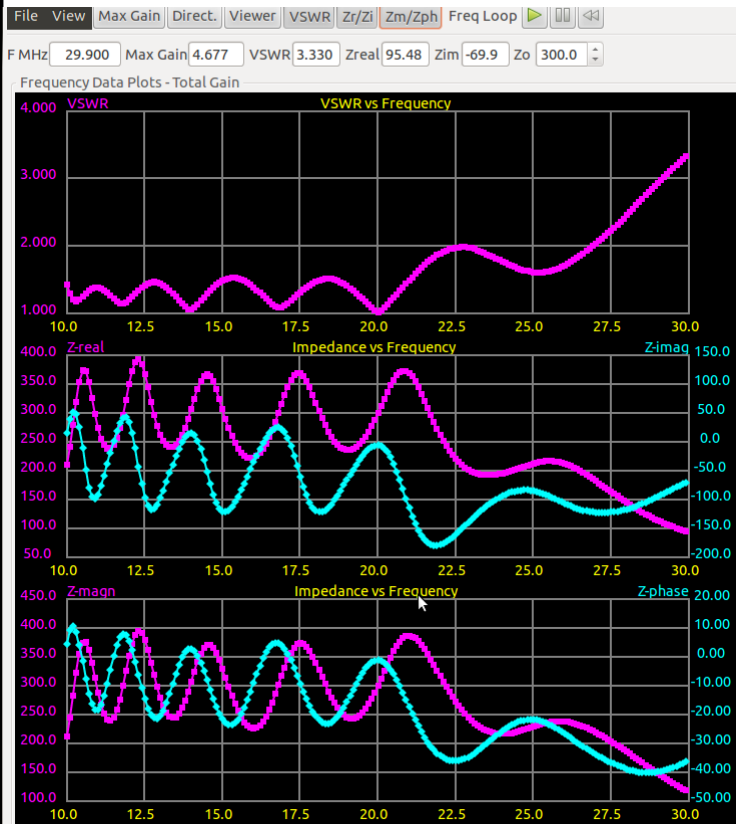
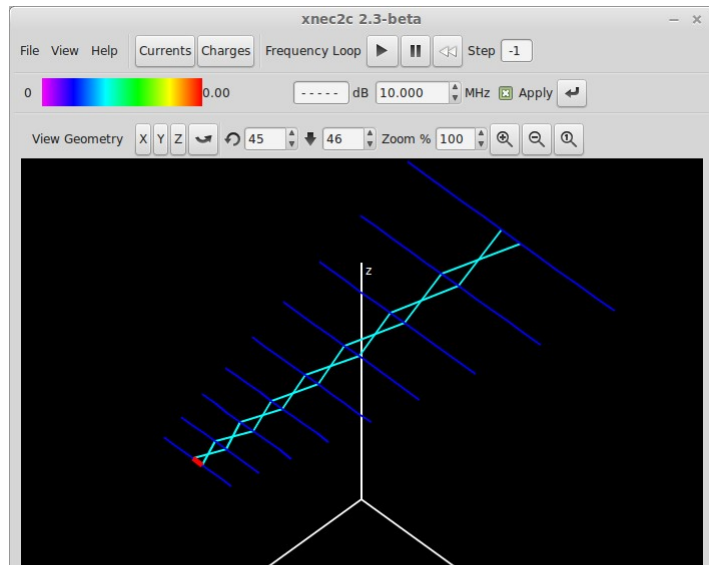
xnec2

Xnec2c is a NEC-2 based antenna modeling tool for Linux with a rich GUI interface. The easiest way to run it is to call it with a NEC file as input:

```
xnec2c -j8 -i yagi.nec
```

The argument `-j8` tells `xnec2c` that the system has 8 CPUs and it should use all of them in parallel.

When it opens `xnec2c` shows the antenna structure (figure right). The antenna can be rotated, dragged, expanded and shrunk using the mouse. Buttons allow display of the current (or charge) along the antenna elements. The Frequency Loop will force `xnec2c` to conduct a frequency sweep as specified in the input file and all open displays are animated during the sweep.

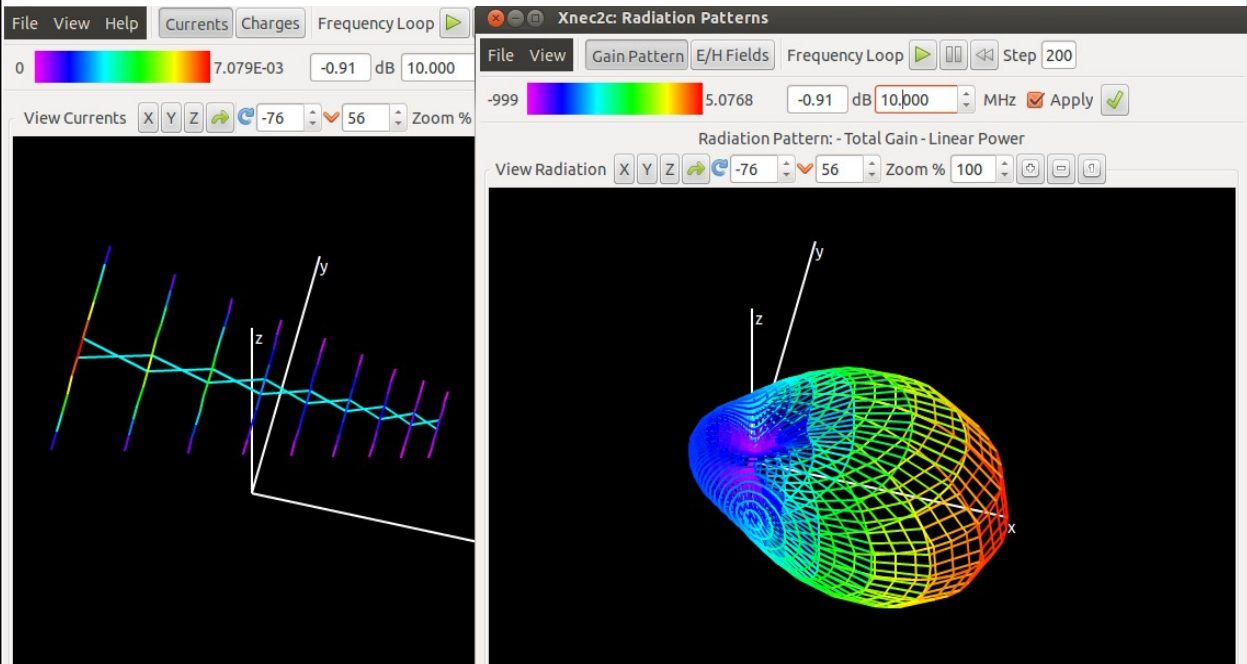


Selecting View->Frequency Plots allows one to view various antenna characteristics versus frequency. The figure at left shows VSWR and impedance in both complex and polar form. Note that the real part of the antenna impedance oscillates around 300 ohms, which suggest using a 300 ohm feed arrangement. Entering $Z_o = 300$ ohms causes the SWR to be calculated for a 300 ohm feed.

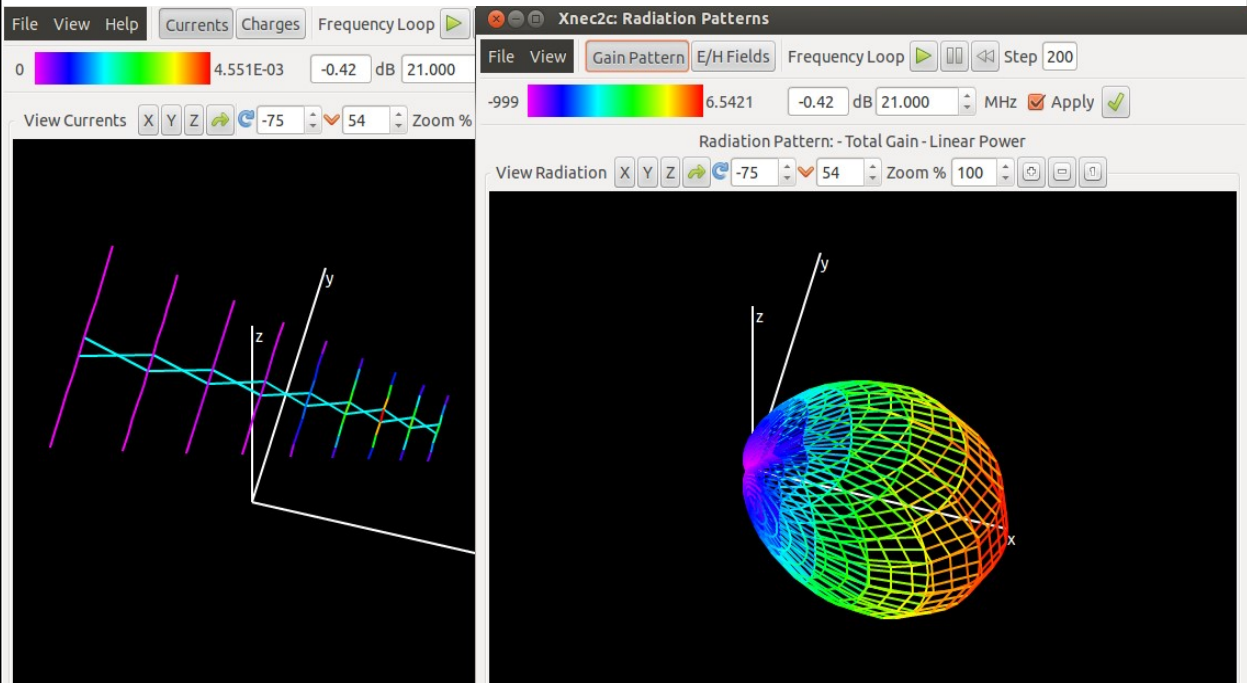
The ability to adjust Z_o is very handy for experimenting with different feedpoint impedances for a multiband antenna (e.g. and OCF). Select a feed impedance that gives the best compromise SWR over the frequency bands of interest and can be realized with 50 or 75 ohm coax and a standard balun. You may determine that a balun/transformer with a non-standard

turns ratio would be worth the effort.

Selecting View->Radiation Patten allows one to view 3D gain plots. The plots below show the antenna element currents (blow left) and 3D gain (below right) at a frequency of 10MHz.



Changing the frequency to 21MHz gives the figures below. Note that the largest currents now flow in the shorter elements.



Note that all the plots are tied together, so rotating the one 3D image will also rotate the others. Also, clicking on a frequency plot will change all the 3D displays to that frequency.

Xnec2c has one serious drawback, its only output is visual, and no results are available for further analysis. If you want to capture all the analytical results NEC produces you must run the command line version:

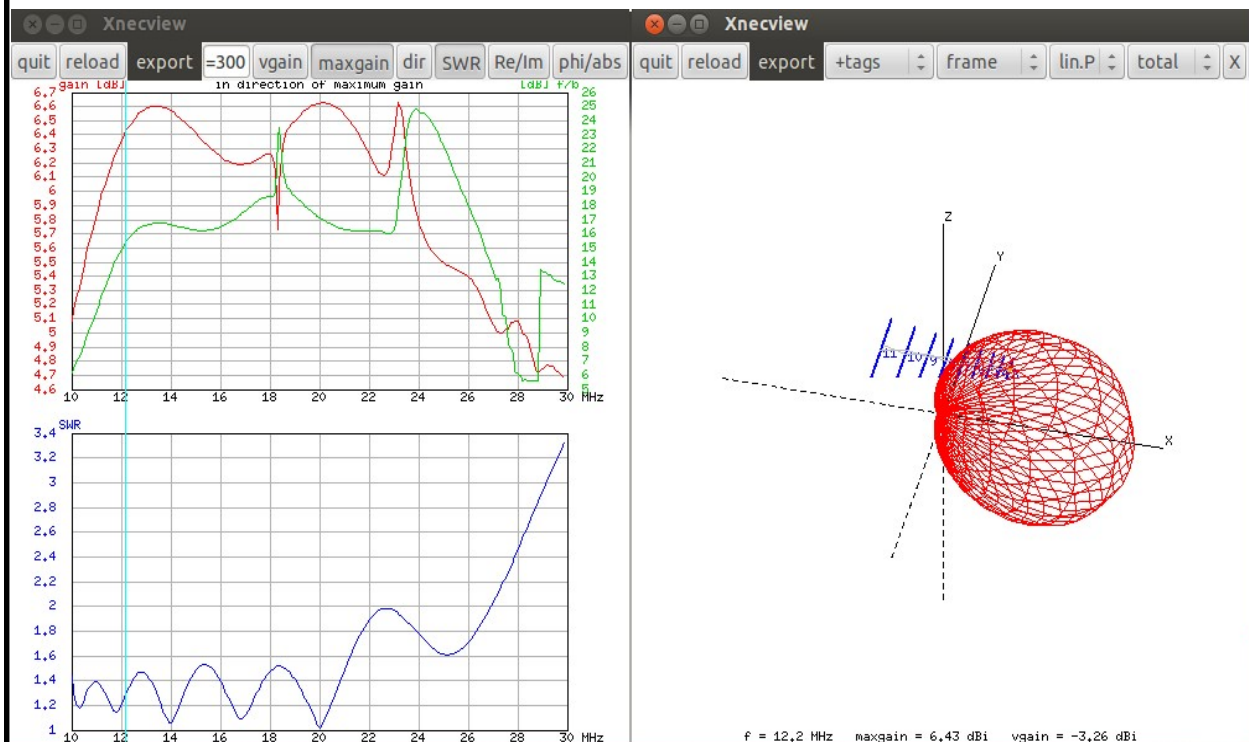
```
nec2c -i yagi.nec
```

Nec2c will crunch the numbers and produce a file called `yagi.out`. For every frequency of the sweep, `nec2c` dumps the current and charge for all antenna segment as well as gain for every point on an azimuth and elevation grid. In this case, `yagi.out` is a 35 MB file with 300,000 lines of ASCII tables. On Linux I routinely open 100MB files in the vim text editor. It takes about 2 sec to open and display `yagi.out` on my 10yr old Dell D620, but that's not what I have in mind for `yagi.out`.

So what good is that huge `yagi.out` file? What do we do with it? Well the first thing to do is look at it with `xviewnec`:

```
xviewnec yagi.nec yagi.out
```

This will let us view the `yagi.out` results in a set of plots very similar to `xnec2c` (see figure below).



In addition to the plots, `xnecview` also produces a nice table of summary output (below). This tabular data is easily imported into a spreadsheet or analysis program for further massaging.

```

user@ubuntu: ~/Antenna/Unix
File Edit View Search Terminal Help
user@ubuntu:~/Antenna/Unix$ xnecview yagi.nec yagi.out
XNECVIEW 1.35

#  freq.      Zr      Zi      SWR      gain      f/b      phi      theta
#  10      210.82    16.314  4.24314  5.08      6.13     0        90
#  10.1    241.32    39.799  4.96339  5.16      6.5      0        90
#  10.2    278.7     51.831  5.77297  5.24      6.9      0        90
#  10.3    318.92    48.034  6.52665  5.32      7.31     0        90
#  10.4    353.87    26.205  7.117    5.4       7.74     0        90
#  10.5    373.59    -9.6248 7.47685  5.48      8.19     0        90
#  10.6    372.56    -48.642 7.5805   5.56      8.65     0        90
#  10.7    353.93    -79.254 7.44041  5.63      9.12     0        90
#  10.8    326.39    -95.633 7.10057  5.71      9.61     0        90
#  10.9    298.1     -98.269 6.62671  5.78      10.1     0        90
#  11      273.91    -90.63  6.09646  5.84      10.59    0        90
#  11.1    255.77    -76.384 5.58817  5.91      11.08    0        90
#  11.2    244.07    -58.362 5.17202  5.97      11.57    0        90
#  11.3    238.61    -38.538 4.90224  6.03      12.06    0        90
#  11.4    239.11    -18.321 4.81155  6.09      12.53    0        90
#  11.5    245.38    1.1166  4.90771  6.14      12.99    0        90
#  11.6    257.28    18.563  5.17343  6.19      13.42    0        90
#  11.7    274.64    32.55   5.57256  6.24      13.84    0        90
#  11.8    296.9     41.235  6.05581  6.28      14.22    0        90
#  11.9    322.69    42.481  6.56835  6.32      14.58    0        90

```

Since NEC only allows us to sweep frequency, if we want to vary other parameters (e.g. height, ground conductivity, loading coil inductance, etc.) We need to make multiple runs of `nec2c`. This is the point where Unix leaves Windows in the dust.

The “prime directive” of Unix is to build solutions upon a set of simple tools which work together, each of which is designed to do one thing well. This approach is so powerful that it is often possible to type a one line program at the command prompt that will produce a result that might take the better part of a day produce in Windows. As an example, let’s find out the frequency of occurrence of GW (wire definition) cards in all the NEC input files on my computer. Here is a simple solution Unix style (execution time 0.4 sec):

```

$grep -c ^GW $(locate '*.nec') | awk -F: '{ print $NF } ' |
sort -n | uniq -c
 4 0
213 1 <== 213 files with only 1 wire.
68 2
. . .
 1 441
 2 844
 1 3790 <== 1 file had a grid of 3790 file.

```

This says to first find all the file ending in `.nec` and search them for GW at the start of the line (^) and return the count (-c). Then print the count at that appears at end of each line after `!:`. Sort the counts numerically (-n) and remove duplicates (uniq) replacing them with a count (-c). If you would rather see the data graphically add `| xgraph -m` onto the end and you get a plot.

Note that the Unix approach is to pick the right tool(s) for the job. The only way I know to do this in Windows is to install cygwin (a Unix-like emulation that runs on Windows) and run the same script.

Below is the listing of the bash script 40m_20m_loaded.pre that I used to analyze a shortened 40/20m dipole with loading coils. It runs 189=7x3x9 different NEC simulations with varying antenna length, load coils positions, and load coil values and compiles the results. Here are the major steps:

- Line 1 – Unix “shebang” identifying a bash shell script file.
- Line 6 – Defines the BASE name of the output files.
- Lines 8-10 - Iterates the values of the half length of the dipole, loading coil segment position, an loading coil value in uH.
- Line 13 - Executes a script that generates a NEC input file, pre.nec with the parameters.
- Line 16 – Runs NEC on pre.nec to produce output in pre.out.
- Line 20 – out2sum extracts results of freq sweep, one line per freq (like xneview output). Additional arguments are appended to the end of each line.
- Line 24 - Extracts lines in summary where SWR is a minimum (resonances).

```
1  #!/bin/bash
2  ## Simulate a 40/20m loaded dipole
3
4  ## Base file name
5  ## We will produce files 40_20.out 40_20.sum 40_20.min_swr
6  BASE=40_20
7
8  for HALFLEN in 8 8.25 8.5 8.75 9 9.25 9.5 ; do
9  for LDSEG in 5 6 7 ; do
10 for LOAD in 12 13 14 15 16 17 18 19 20 ; do
11
12     ## Generate NEC input file in pre.nec
13     40m_20m_loaded.gen $HALFLEN $LDSEG $LOAD > pre.nec
14
15     ## Run NEC on pre.nec
16     nec2c -i pre.nec
17     cat pre.out >> $BASE.out
18
19     ## Extract summary data from NEC output
20     out2sum pre.out $HALFLEN $LDSEG $LOAD
21     cat pre.sum >> $BASE.sum
22
23     ## Extract resonances (min SWR)
24     resonance pre.sum >> $BASE.min_swr
25
26 done ; done; done; done
```

When the script is done, we will have a file 40_20.min_swr that will look like this:

MHz	R	X	SWR	L/2	Seg	uH
7.6	73.354	-1.0198	1.46761	8	6	16
14.68	33.877	-139.49	13.5669	8	6	16
7.5	71.507	-5.5228	1.44665	8	6	17
14.68	40.923	-91.99	6.00951	8	6	17
etc...						

This is easily imported into your favorite analysis tool: Matlab, Octave, or Excel.

Appendix – Script Listings

40m_20m_loaded.gen

This file includes a bit of shell jargon.

$\$(X + Y*A)$ – evaluates to the numerical result of the math expression inside.

$\${LOAD}E$ – interpreted \$LOAD with E appended as opposed to \$LOADE

```
1  #!/bin/bash
2  # Generate a NEC input file with variable parameters
3  # 40m_20m_loaded.gen half_length load_segment coil_uh
4
5  HALFLEN=$1
6  LDSEG=$2
7  LOAD=$3
8
9  cat <<EOFEOF
10 GW 1 61 $HALFLEN 0 9.65 -$HALFLEN 0 9.65 0.001
11 GE 1 0 0 0 0 0 0 0 0
12 EX 0 1 31 01 1 0 0 0 0 0
13 LD 0 1 $(LDSEG+1) $(LDSEG+1) 0 ${LOAD}E-6 0 0
14 LD 0 1 $(SEGS-LDSEG) $(SEGS-LDSEG) 0 ${LOAD}E-6 0 0
15 FR 0 200 0 0 6.0 0.1 30 0 0 0
16 GN 0 0 0 0 12.0 0.01 0 0 0 0
17 PT -1
18 XQ 0
19 EN
20 EOFEOF
```

out2sum

Note that the VSWR is computed assuming a feedline impedance of 50.

```
1  #!/bin/bash
2  ## out2sum : Generate summary file from NEC output file.
3
4  BASE=${1%.out}
5  shift
6
7  awk -v "pars=$*" '
8  /UNNORMALIZED IMPEDANCE/,/CARD/ { n++ }
9  n > 3 && NF == 9{
10     f = $1 + 0
11     r = $2 + 0 ; x = $3 + 0
12     gamma = sqrt( (r-50)*(r-50)+x*x) / sqrt( (r+50)*(r+50)+x*x)
13     vswr = (1+gamma) / (1-gamma)
14
15     print f, r, x, vswr, pars
16 }
17
18 ' $BASE.out | column -t > $BASE.sum
```

resonance

```
1  #!/bin/bash
2  # resonance : Find SWR minima
3
4  awk '
5  BEGIN {
6      swr_1 = swr_2 = swr_3 = 999999;
7  }
8
9  {
10     id = $5
11     if( id != id_last ) {
12         swr_1 = swr_2 = swr_3 = 999999;
13     }
14     swr_1 = swr_2
15     swr_2 = swr_3
16     swr_3 = $4
17     if( swr_2 < swr_1 && swr_2 < swr_3 ) { print NR-1, rec_2 }
18     rec_2 = $0
19     id_last = id
20 } ' $1
```

References:

<http://www.nec2.org/> - The place to go for NEC2 documentation and history.

http://www.nec2.org/part_3/toc.html - Here you will find documentation on all the different “cards” used in NEC files. A good reference but a bad tutorial.

<http://www.gsl.net/m5aiq/nec-tutorial-a.html> - A short NEC tutorial outlining some of the key card types.

1. L. B. Cebik, W4RNL - “A Beginners Guide to Modeling With NEC”. A multi-part series starting in QST, Nov 2000. This tutorial is base on EZNEC, but technical pitfalls to avoid apply to all versions of NEC.

L. B. Cebik, W4RNL - “So You Want to Read A NEC Deck” is a decent intro to NEC cards with comparison to EZNEC. <http://www.antennex.com/w4rnl/col0102/amod47.html>

<http://www.arrl.org/antenna-modeling-files> - Various zip archives of NEC antenna models. Hacking an existing model is the easiest way to create your own model.

https://en.wikipedia.org/wiki/The_Unix_Programming_Environment - This is a book by Brian Kernighan and Rob Pike outlining the philosophy behind Unix and how to write simple programs using standard Unix tools like awk, sed, and C.