Digisonde 256 data decoding: 16 channel ionograms

Terence Bullett [1]
Ivan Galkin [2]
David Kitrosser [2]

1- Air Force Research Laboratory, Space Vehicles Division. Hanscom AFB, MA
2- University of Massachusetts Lowell Center for Atmospheric Research, Lowell. MA

Abstract

The Digisonde 256 digital ionosonde possesses latent capabilities that are often underused. Among these is the ability to produce and record the 8 bit resolution amplitudes and phases versus for all 16 signal channels inside the digital signal processing unit. Modern computer and data storage technologies make it practical to record and store these large 16 channel data sets. The availability of these additional, higher resolution data can enable new analysis techniques and the possibility of new scientific discoveries. This article describes the complexprocess of decoding and comprehending these data,shows some results and provides the FORTRAN source code for the program which performs these tasks.

Digisonde 256 Channels

The term channel, as used to describe Digisonde 256 (D256) data, refers to a set of memory locations inside the Digital Signal Processing section of the ionosonde. For the most common operating modes, these memory locations can be visualized an array of 16x128 elements. There are actually 2 arrays of 16x128 bytes, one for amplitude and one for phase, for each frequency of the ionogram. Each element is 8 bits. The 128 dimension of the array represent 128 range bins. The 16 dimension of the array are the channels. These channels represent specific combinations of Doppler Bin, Receive Antenna Polarization and Receive Array Beam Position. The specific values and combinations of these values, and the range bin values, are referred to as meta-data, or "data about the data".

It is essential to recognize that the Digisonde 256 is inherently and invariably based on Complex Discrete Fourier Transform (CDFT) signal processing. Digitized, time domain in-Phase and Quadrature receiver samples are sent immediately into hardware CDFT circuits. All outputs are frequency domain representations of the time domain signals. The number of Doppler lines computed is independent of, but limited by, the number of samples in the time series. The CDFT process is phase coherent,and independently processes signals versus range. It can also independently process 2, 4 or 8 interlaced signals, where the interlacing is performed from one transmitted pulse to the next. This interlacing allows other portions of the ionosonde, typically the antenna switch, to change operating modes from pulse to pulse. Left and right hand circular polarizaton, or receive antenna array beam position are This produces the effect of simultaneously measuring, for example, Ordinary and eXtraordinary propagation modes.

The 16 channels represent the total number of Doppler lines and interlaced signals. The product of these two values is always 16. For example, if only 2 interlaced signals are being measured, such as O and X, there are 8 Doppler lines available for each of these 2 signals.

The actual data reported by the D256 is the spectral amplitude and spectral phase. The position of the value in the 128x16 array, as a function of system settings, determines the range, polarization, Doppler and direction of arrival. Frequency of the ionosonde is an independent variable, recorded as part of the system settings.

The D256 is quite flexible, thus it is possible to operate the ionosonde in modes where almost everything stated above is over-simplified or false. This article and the code covers the more common modes of operation. The Open Source nature of the source code invites others to contribute and improve on the code.

Relationship to other data formats

The standard format for recording ionogram data from a D256 is the Modified Maximum Method, or MMM. In brief, the MMM process starts with the 16 channel data and reduces it to a 'single channel' by determining the maximum value among the 16 channels available at each range gate. There is some consideration given to adjacent ranges, which is the Modified part of MMM. At each range, the number of the channel selected by the MMM process is retained using 4 bits of data. This value is related to what is called the 'Status' The 4 most significant bits of amplitude from this channel are combined with the 4 bits of channel number to form an 8 bit value for each range gate. Phase data are discarded.

Theoretically there are other MMM modes where differing amounts of the 16 channel data are saved, but these modes are almost never used. Also, there are some modes, specifically 256 range gate modes, where there are effectively only 8 channels, thus only 3 bits are needed for channel number, leaving 5 bits for amplitude.

The MMM is a data selection process primarily designed to save the most important ionogram data while dramatically reducing the size of the resultant files. The MMM process results in about a 30:1 reduction in file size. This comes at the expense of amplitude resolution, Doppler spectra and phase information. Back in the 1980's, in the era of the IBM PC, DOS3.3, 10 MB hard disks and 30MB 9-track tape drives, this data reduction was essential. In 2002, where recordable CD's with 650 MB cost less than $1US and hard disks hold tens of gigabytes, the need for data reduction is diminished.

Obtaining 16 Channel Data

There are approximately 25 Digisonde 256 ionosondes in operation around the world in June of 2002, all of which are capable of recording 16 channel data in some form or another. The specific technique depends on the Digisonde configuration. There are also problems and limitations associated with each approach.

The 16 channel data format produces 4096 bytes of data for each frequency in the ionogram. For an ionogram from 1 to 16 MHz with a linear frequency step of 50 kHz, the resultant data set is 1.2 MB. While large, it is not unmanageable.

Artist4 - The standard Artist4 configuration uses the Artist3 hardware to interface between the Artist4 computer and the Digisonde 256 Processor chassis. Both computers need to properly handle these data. This is the tested configuration.

The Front End (FEND.EXE) program running on the Artist3 computer accepts the data from the D256 Processor and creates 16 channel (.16C) data files on the Artist4 computer via the Local Area Network (LAN) between the two machines.

Versions of FEND.EXE prior to March 2002 have a minor bug where the format of the data files is not precisely that described in the documentation. The Record Type of each 4K block of data is missing. The decoding software can be modified to handle this error, but the resulting program looses robustness and will attempt to process invalid data types, with unstable behavior as a likely outcome.

In general, as of June 2002, the Artist3 and Artist 4 systems are not capable of handling ionograms with 256 range bins. This is independent of the 16 channel data, but does prevent the successful recording and processing of 256 range gate ionograms. This limitation is reflected in the included programs due to the lack of data for code development and testing.

There can also be limitations in memory and RAM disk space which will cause the DOS based Artist3 system to crash. Additional RAM and the adjustment of the RAM disk sizes can reduce this problem.

The Dispacher program (DISPATCH.EXE) on the Artist4 is responsible for transmission and archival of the .16C files. Dispatcher will also attempt to run a program named *md256.exe* to process each 16C ionogram. The .16C files are handled identically to other file formats, in that they can be transmitted by FTP, stored individually, concatenated into 1 file per day archives, retained in a local archive, and written to removable media for centralized archiving. This is all adjustable using configurations in the DISPATCH.UDD file.

The Dispatcher has a normally undocumented feature which uses the time of the ionogram to select whether it is to be processed by the Artist ionogram scaler. This can be useful to prevent unusual ionogram modes from crashing Artist or contaminating the scaled ionogram database with inaccurate scalings.

```
     UNDOCUMENTED FEATURE !!
     Only ionograms started at these times will be processed
     backup4 program does not recognize this entry.

*026 < 00 05 15 20 30 35 45 50  >

                                      ...

     SHORT TERM ARCHIVE SELECTION - PUBLIC SECTOR
     specify file extensions
*043 < ART MMM BEM RSF SBF DFT SAO IOS IHT> no GIFs here, see below

                                      ...

    LONG-TERM ARCHIVE STORAGE OPTIONS -- PUBLIC SECTOR
     choice of data to store
      0 - disable long-term public archiving
     +1 - store SAO one day files
```

```
     +2 - store ART one day files
     +4 - store ionogram one day files
     +8 - store drift one day files
    +32 - store 16 channel day files
   +256 - store GIF individual files from WWW area
*054 <13>

                                        ...

%7 FTP DATA REPORTS

     FTP OUT DIRECTORIES AND TRANSFER SELECTIONS
     Data files to be delivered to remote FTP servers go here
     multiple field: more than one line with same code# possible

     These directories must have SYSTEM\ subdirectory with
         ACCOUNT.FTP file in it. The file contains one string
         with three words: IP# of the server, username and password

     Each code 080 must be accompanied with three more:
     081: file extension list of data to be transferred without compression
     082: file extension list of data to be transferred compressed
     083: put 1 if data is accumulated during communication blackouts
     084: remote directory to change to
     085: minutes to include into reports (looks at filename to match)
     086: LATEST reports: a copy of latest data is reported under "LATEST.???"
name
          Specify extension list. July 98: LATEST.GIF and LATEST.BIT available
*082  < .16C >
                                        ...

     CONTENT SELECTION FOR RMD ONE DAY FILES
     make sure these files appear in the longterm secure archive
      0 - nothing
     +1 - SAO one day files
     +2 - ART one day files
     +4 - ionogram one day files (all kinds combined together)
     +8 - store drift one day files
    +32 - Store 16 Channel one day files
*111 < 15 > default = 15 (1+2+4+8)
                                        ...

     CONTENT SELECTION FOR INDIVIDUAL FILES
     specify extensions for individual file backup
     use empty string to disable storage of individual files
*112 < .16C >
```

Note that the Dispatcher requires the file extension to be .16C, including the period. This is not true for the other file types. As a warning, these .16C files can fill the Artist4 hard disk and cause erratic system behavior and loss of data.

The Processor needs to have the correct parameter settings in order to initiate the saving of 16 channel data. In addition to all the parameters necessary to make a standard, sweep frequency ionogram, the following must be set

P1=6 This acts as a flag to the FEND.EXE program, instructing it to record 16 channel data in the Artist3 filesystem.

P3 = 12 or 13 (C or D) This causes the Processor to send data to the Artist computer and ignore the value of P1 and not try to actually write to a tape drive attached to the Processor. The date on the processor firmware should be at least October 1991. The latest firmware version is <date>

Artist3: The standard Artist3 configuration of a '386 or '486 PC with a 150MB QIC tape cartridge can theoretically record 16 Channel data. This is presently untested due to the lack of a functioning Artist3 testbed system. Care must be taken to dump data from RAM disk to tape often, because filling the RAM disk will crash the system. Processor settings are identical to those used in the Artist4 configuration. Updated Processor firmware and FEND.EXE are recommended.

Processor: The Digisonde 256 Processor/Transceiver chassis can record 16 channel data directly onto 9-track tape. While this method is now archaic and these tape drives large, expensive, fragile and low capacity, the method still works. It has one advantage in that it lacks some of the limitations of the Artist based modes. The disadvantage is, of course, that you have to read these tapes. AFRL and UMLCAR retain this capability, but it is largely unused and vulnerable to failures in old, irreplaceable hardware.

Processor ionogram parameters must include:

P1=6 Write 16 channel amplitude and phase to the tape drive.

Data Decoding Process

The driving philosophy behind this effort is to develop a methodology that can decode the 16 channel data format from its original binary to a text file format as well as determine and include the meta-data along with the observations. The result is the program *dgs16c*, with current version number 0.3x, June 2002. This program was developed and tested under Debian GNU/ Linux with the g77 FORTRAN 77 compiler. It has a few FOTRAN 90 extensions. This program is meant not as a finished application, but as a tool for a user to adapt to meet their own needs.

Binary Data Decoding

The first step of the process is to decode the binary data format. The input file is processed and read in 4K byte blocks. Each block is decoded into its 128 range by 16 channel array of bytes for amplitude and phase. There is a preface which contains all the ionogram parameter values in the Processor for this data block. This is encoded serially into the low order bits of the first 360 amplitude values.

Identification and Quantification of Meta-Data

This section describes the logic required to identify and assign engineering values to the meta-data. By engineering values we mean (virtual) Range in kilometers, Amplitudes in dB, Phase in degrees, Frequency in MHz, Doppler shift in Hz, Azimuth in degrees and Zenith in degrees.

For the quantification of polarization, we look to similar work performed by the Dynasonde community. The Dynasonde polarization is a measured quantity called 'chirality', which tends heavily toward -90 degrees for Ordinary polarized waves and +90 degrees for extraordinary polarization. (J.W. Wright, private communication). We adopt this scheme.

This process is derived from *Galkin (2001)* and is very similar to the process used to obtain this same information from MMM data. For details such as the contents of the various translation tables, reference this document or the source code.

First, each of the 16 channel numbers is converted into a Status value. The Status is the same as what would be printed on the D256 Processor raw data printer and is more independent of ionogram settings than channel number.

Encoded into the Status value is both Doppler and Beam information, where a beam is a combination of antenna polarization and phased array direction. The Doppler is extracted using one table and a variety of ionogram parameters. The Beam information is decoded using a series of antenna translation tables. These tables can be site specific, depending on the version of specific cards in the D256, especially the Antenna Drive (Card 08) and the physical layout of the receive antenna array.

Finally, scale factors to translate 8 bit amplitude values into dB and 8 bit phase values into degrees are determined from the preface settings and applied to the actual data.

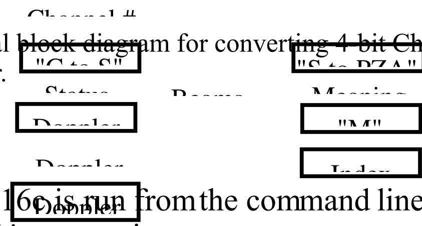

Figure 1 - Functional block diagram for converting 4 bit Channel number CCCC into Polarization, Zenith, Azimuth and Doppler.

Execution

The program dgs16c is run from the command line with several command line switches implemented. It's useage is:

```
dgs16c {-vapsh} {-f #} filename
Decode 16 channel data from a Digisonde 256 16C file
Filename must be provided on the command line.
Command line options:
  -v  Verbose.  Prints the status on stdout
  -f  <freq>  extract data for <frequency>
  -c  Correct phase based on range gate
  -a  Write Amplitude Table
  -p  Write Phase Table
  -s  Write data for gnuplot splot command
  -t  <val> Threshold Detect, <val> above noise
             output format set to -s
```

There are two output formats, tabular and the list or 'splot' format. The table format gives amplitude, phase or both for each channel number. The channels are sorted by the values of each channels' meta-data. The sort order can be changed in the source code.
Lines that start with # are comments. The first two columns are the frequency and range gate values for the data. The next 16 values are the data, in this example amplitude in dB, for each channel.

```
# 2001 089 22:55:05 2.0025 01089225505 22420000 020000 60C4103 02130 12E 908 4555 424 003 223C
#      Frequency  2.0025  2.0025  2.0025  2.0025  2.0025  2.0025  2.0025  2.0025  2.0025  2.0025  2.0025  2.0025  2.0025  2.0025  2.0025  2.0025
#       Polarize  -90.00  -90.00  -90.00  -90.00  -90.00  -90.00  -90.00  -90.00  -90.00  -90.00  -90.00  -90.00  -90.00  -90.00  -90.00  -90.00
#        Doppler  -6.250  -6.250  -4.688  -4.688  -3.125  -3.125  -1.562  -1.562   1.562   1.562   3.125   3.125   4.688   4.688   6.250   6.250
#        Azimuth    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
#         Zenith    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
  2.0025    60.0    64.1    57.0    63.8    57.8    62.2    66.0    57.0    61.9    52.5    58.9    58.9    46.5    56.2    64.9    60.8    61.5
  2.0025    70.0    61.1    58.5    63.4    63.0    58.1    58.9    54.0    63.4    51.8    57.4    56.2    40.9    62.6    59.2    63.4    64.9
  2.0025    80.0    60.8    58.9    66.0    59.2    63.8    58.1    48.8    58.1    60.4    62.2    61.9    56.2    55.1    61.1    48.8    61.1
  2.0025    90.0    60.4    57.8    60.8    57.4    64.9    63.4    60.8    54.0    55.1    63.8    50.2    57.0    55.5    61.1    46.9    60.8
  2.0025   100.0    53.6    57.4    60.8    49.5    56.2    61.9    51.8    59.6    48.8    61.5    51.8    59.6    60.4    58.1    63.4    53.2
  2.0025   110.0    63.4    63.4    64.5    58.9    57.0    59.2    54.0    50.2    60.0    60.0    60.8    60.4    49.9    53.2    58.1    55.1
```

Figure 2 - Sample *dgs16c* output for Amplitude Table (-a) mode.

The splot or -s mode produces the following format output. This format is very useful for plotting with the free *gnuplot* software package. In this format all data and meta-data values are in columns. Each range gate is separated by a blank line. A 'noise' or most porbable amplitude is computed and listed. The data are sorted by Polarization and then Doppler. This is the format used for the plots in this paper.

```
#            Time      Freq     Range  Polarize     Noise   Doppler        Az        Zn  Ampltude     Phase
2001 089 15:01:02    8.5025    60.000   -90.000    48.000    -3.125     0.000     0.000    49.875   248.906
2001 089 15:01:02    8.5025    60.000   -90.000    48.000    -2.344     0.000     0.000    52.500   236.250
2001 089 15:01:02    8.5025    60.000   -90.000    48.000    -1.562     0.000     0.000    50.625   213.750
2001 089 15:01:02    8.5025    60.000   -90.000    48.000    -0.781     0.000     0.000    52.875   210.938
2001 089 15:01:02    8.5025    60.000   -90.000    48.000     0.781     0.000     0.000    55.125   210.938
2001 089 15:01:02    8.5025    60.000   -90.000    48.000     2.344     0.000     0.000    48.750   206.719
2001 089 15:01:02    8.5025    60.000   -90.000    48.000     1.562     0.000     0.000    53.250   202.500
2001 089 15:01:02    8.5025    60.000   -90.000    48.000     3.125     0.000     0.000    46.125   309.375
2001 089 15:01:02    8.5025    60.000    90.000    48.000    -3.125     0.000     0.000    48.750    15.469
2001 089 15:01:02    8.5025    60.000    90.000    48.000    -2.344     0.000     0.000    50.625     0.000
2001 089 15:01:02    8.5025    60.000    90.000    48.000    -1.562     0.000     0.000    52.125   338.906
2001 089 15:01:02    8.5025    60.000    90.000    48.000    -0.781     0.000     0.000    54.750   341.719
2001 089 15:01:02    8.5025    60.000    90.000    48.000     0.781     0.000     0.000    58.500   344.531
2001 089 15:01:02    8.5025    60.000    90.000    48.000     1.562     0.000     0.000    49.500   347.344
2001 089 15:01:02    8.5025    60.000    90.000    48.000     2.344     0.000     0.000    48.375     0.000
2001 089 15:01:02    8.5025    60.000    90.000    48.000     3.125     0.000     0.000    48.375   343.125

2001 089 15:01:02    8.5025    65.000   -90.000    48.000    -3.125     0.000     0.000    49.875   149.062
2001 089 15:01:02    8.5025    65.000   -90.000    48.000    -2.344     0.000     0.000    41.625   135.000
2001 089 15:01:02    8.5025    65.000   -90.000    48.000    -1.562     0.000     0.000    55.500   333.281
2001 089 15:01:02    8.5025    65.000   -90.000    48.000    -0.781     0.000     0.000    64.500   324.844
2001 089 15:01:02    8.5025    65.000   -90.000    48.000     0.781     0.000     0.000    62.625   326.250
2001 089 15:01:02    8.5025    65.000   -90.000    48.000     2.344     0.000     0.000    55.500    74.531
2001 089 15:01:02    8.5025    65.000   -90.000    48.000     1.562     0.000     0.000    54.000    21.094
2001 089 15:01:02    8.5025    65.000   -90.000    48.000     3.125     0.000     0.000    52.500   123.750
2001 089 15:01:02    8.5025    65.000    90.000    48.000    -3.125     0.000     0.000    45.000   322.031
2001 089 15:01:02    8.5025    65.000    90.000    48.000    -2.344     0.000     0.000    40.125    77.344
2001 089 15:01:02    8.5025    65.000    90.000    48.000    -1.562     0.000     0.000    55.125   130.781
2001 089 15:01:02    8.5025    65.000    90.000    48.000    -0.781     0.000     0.000    62.625   129.375
2001 089 15:01:02    8.5025    65.000    90.000    48.000     0.781     0.000     0.000    60.750   130.781
2001 089 15:01:02    8.5025    65.000    90.000    48.000     1.562     0.000     0.000    49.875   156.094
2001 089 15:01:02    8.5025    65.000    90.000    48.000     2.344     0.000     0.000    45.000   182.812
2001 089 15:01:02    8.5025    65.000    90.000    48.000     3.125     0.000     0.000    40.875    88.594
```

Figure 3 - Sample *dgs16c* output for Splot List (-s) mode.

These output formats can be easily changed in the source code, or an external filter program can be run to perform custom data selection or analysis procedures.

Results and Observations

Figure 4 shows an ionogram extracted with dgs16c in the -s format with a threshold of 20dB. The values of Range were plotted versus Frequency, separately for Ordinary (POLARIZE < 0.0) and eXtraordinary (POLARIZE > 0.0). O echoes are red and X echoes are Green. Both O and X signals are overplotted where both are above the noise. There are no amplitude data in this graph. While not terribly impressive, it is a start.

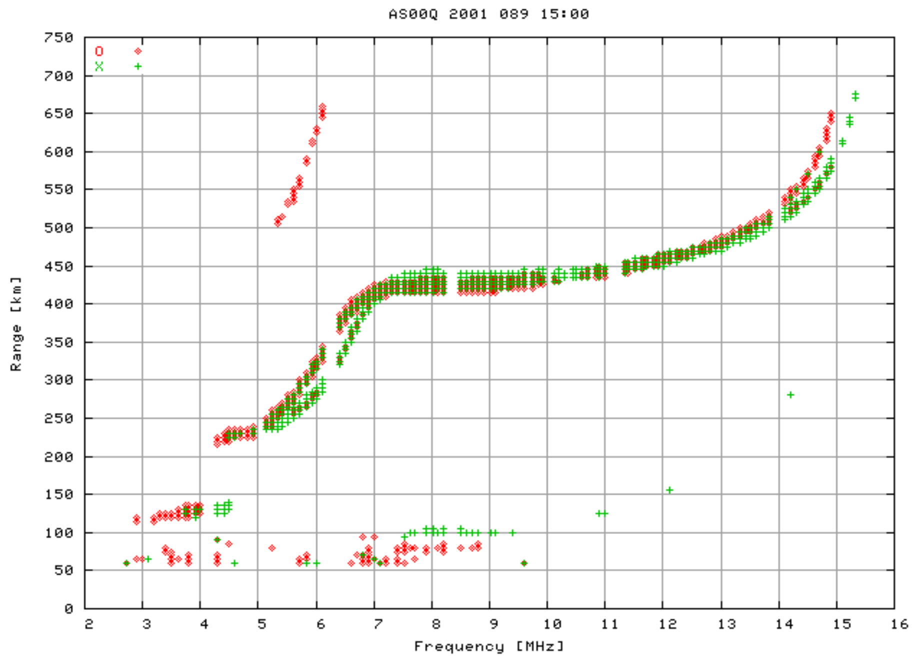

Figure 4 -An ionogram from Ascension Island, 2001 089 15:00Z. Data were extracted from the 16 channel data using the -s option for output and the -t 20 option for threshold selection of data points 20 dB or more above the 'noise'. Note that for many echoes both the O and X channels have strong signal.

If we select a specific Frequency and Range bin from the ionogram in Figure 4, we can plot the Amplitude and Phase of the echo for each of the 16 channels. This is shown in Figure 5, for F=5.7 MHz, R=285 km. In this plot we can see several features. The spectral shape of each signal is qualitatively consistent with a Hanning (cosine squared) weighted Complex Discrete Fourier Transform. (*Bullett, 1993*). The phase of each spectrum is constant across each main spectral lobe. Note that the amplitude and phase of the X signal, in bins 8-15, is different than the O signal in bins 0-7. It is unknown, without further study, if the X signal is present due to cross-pdarization coupling in the receive antennas.

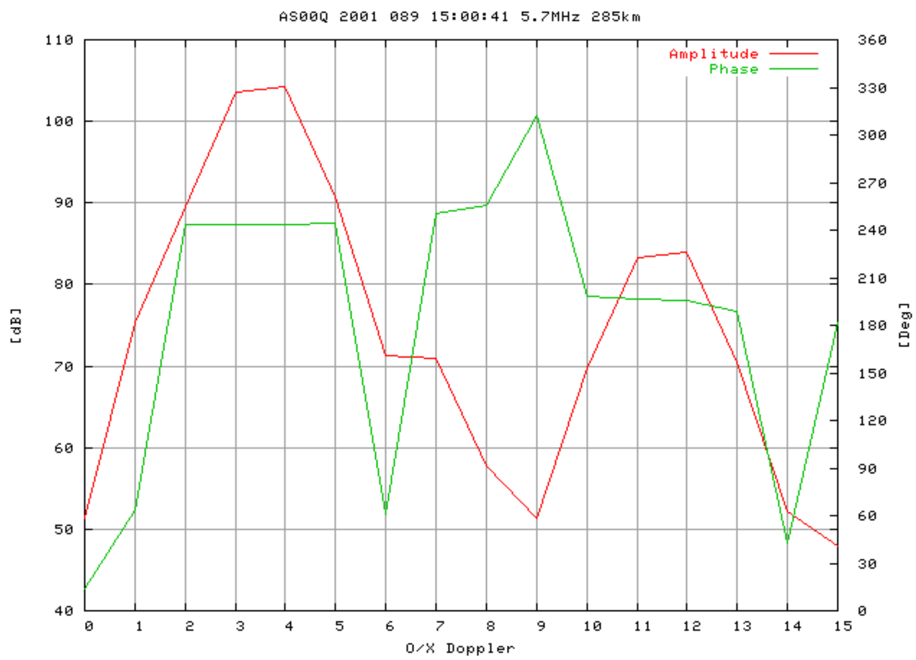

Figure 5 -Spectral amplitude and phase versus relative Doppler shift. Values of O/X-Doppler 0-7 are Ordinary polarization, 8-15 are eXtraordinary.

Several spectra as in Figure 5 can be plotted as a function of range. This produces a waveform-range plot along the horizontal axis. In this type of plot, shown in Figures 6 and 7, the range is constant for each group of 8 data points. These 8 data points are the spectral lines at that range. The range gate increments by 5 km for each set of 8 spectral lines. This plot allows the visualization of 3 dimensional data sets without resorting to 3D plots, which can be difficult to visualize.

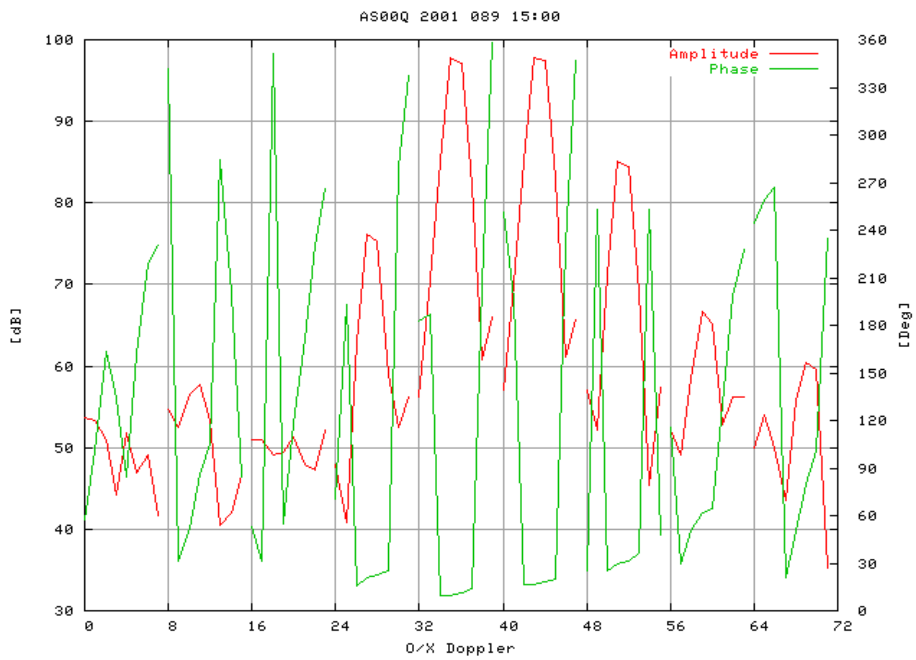

Figure 6 - Echo amplitude and phase versus Doppler-Range for Ordinary polarized signals. Note the consistency in range, spectral shape, and phase.

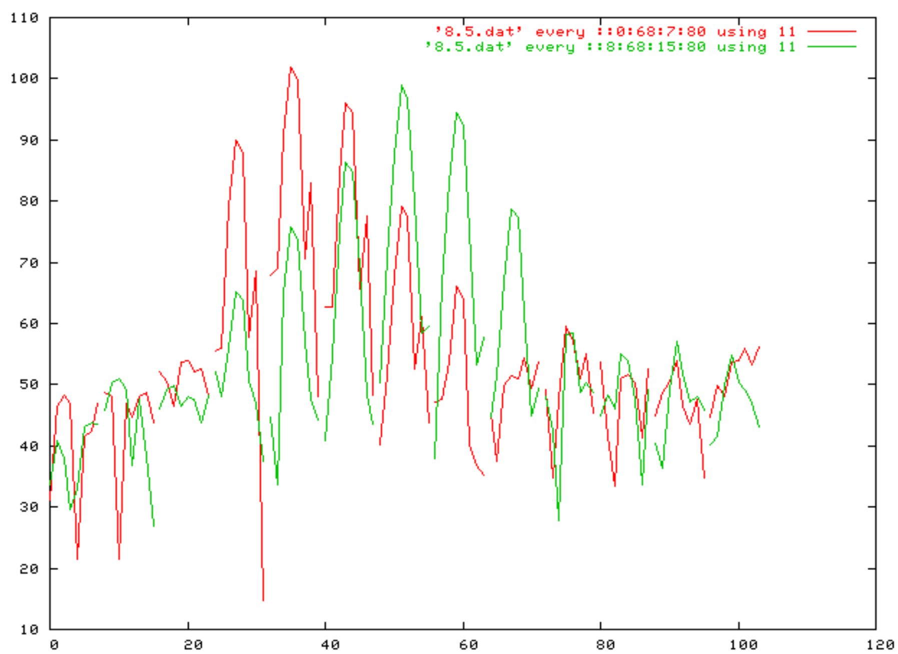

Figure 7 - This figure shows both O and X signal for Ascension 2001 089 15:00. In this portion of the ionogram the O (red) and X (green) signals mingle, and O does come at a shorter range than X. Note the presence of a consistent feature in the O signal in the higher Doppler bins, ~20 dB lower than the main echo.

Future Work

Follow on engineering work to the *dgs16c* program involves enhancing it to handle more ionogram modes, particularly 256 Range Gates and Precision Group Height modes. Improvements to the FEND.EXE program are needed to record these modes. The system specific data in the code needs to be verified and tested, while the ability of the code to accept site specific antenna configuration data needs to be expanded.

Extension of the concept behind *dgs16c*, namely that of an Open Source tool for reading Digisonde data files, needs to be extended to other data types, specifically MMM, Single Block Format (SBF) ionogram files, as well as the Drift mode data files where amplitude and phase are recorded on individual antennas. Calibration data for receivers and antennas should be considered.

Conclusions

There are potential discoveries to be made by looking at the additional and higher resolution data present in the Digisonde 256 16 channel datasets. The *dgs16c* program is a preliminary Open Source tool to facilitate this investigation. We hope that users of this work will follow the spirit of Open Source and contribute their modifications back to the community.

Acknowledgments

This work was support by the Air Force Office of Scientific Research, AFOSR Task

References

Bibil, K., Reinisch, B.W., Kitrosser, D.F., Digisonde 256: General Description of the Compact Digital Ionospheric Sounder. University of Lowell Center for Atmospheric Research; 1981, Revised 1991.

Bullett, Terence W. Mid-latitude ionosphere plasma drift: a comparison between ionosonde and incoherent scatter radar" UMass Lowell Doctoral Thesis; 1993

Galkin, Ivan A. *Decoding of Status Information in MMM Digiosnde Ionograms,* UMass Lowell Center for Atmospheric Research Technical Memorandum; August 2001.

Tang, J., et al., *ARTIST Tape Output Formats*, Technical report GL-TR-09-0190 July 1990

Internet Resources

Debian Linux Operating System http://www.debian.org
gnuplot graphing program http://www.gnuplot.vt.edu/
g77 FORTRAN compiler http://www.gnu.org/software/fortran/fortran.html
Science Applications for Linux http://sal.kachinatech.com/

Appendix A : Subroutine List

The list of files, subroutines and some of the comments at the start of each.

```
==> dgs16c.f <==
C
      PROGRAM DGS16C
C
C     Digisonde 256 16 Channel data decoder
C
C     This program forms the basis of a decoder for 16 channel (16C)
C     data obtained from a UMass Lowell Digisonde 256 ionosonde.
C
C     Terence Bullett
C     Air Force Research Laboratory
C     15 September 2001
C     linux g77
C
C     License:
C     This is in the public domain.
C     This software was paid for by the Taxpayers of the
C     United States of America.
C     Thank one the next time you meet.
C
C     References:
C        GL-TR-09-0190 "ARTIST Tape Output Formats"  July 1990  J. Tang, et al
C
C        "Decoding of Status Information in MMM Digisonde Ionograms"
C                 I. Galkin  September 2001
C
C        Digisonde 256 -- General Description of the Compact Digital
C             Ionospheric Sounder", Bibl, Reinisch, Kitrosser, 1981 Revised
C              aka "Green Book"
C
C        "Digisonde 256 Operators Manual"  Terence Bullett

==> ampv.f <==
C
C====+==================================================================+==
C
      REAL FUNCTION AMPV(EAMPS,MAXRBIN)
C
C     Calculate the Amplitude Most Pobable Value as a proxy for the
C     noise floor of the Range-Doppler data in EAMPS
C     Data are expected in the range of 0 to 120 dB and binned in 2dB bins.
C     The resolution, range and number of bins is hard-coded.  Sorry.

==> channel_map.f <==
C
C====+==================================================================+==
C
      SUBROUTINE CHANNEL_MAP(PREFACE, FRQ,APOL,HDOP,AZ,EL,NBEAM,NDOPP)
C
C     T. Bullett, AFRL, 18 January 2002
C
```

```
C     This subroutine performs the rather complex task of supplying the
C     'meta-data' for the 16 channels of the 16 channel Digisonde 256
C     data.  The meta-data, or data about data, provide information on the
C     context of these different channels.  For the Digisonde 256, the
C     'channel number' can specify Polarization, Doppler and Arrival Angle.
C     It is the precise values of these that this routine computes, based
C     instrument settings.
C     There are lots of caviats here, mostly falling in categoryies:
C       A) That particular setting was not coded or debugged
C       B) Your Digisonde may have different values in its PROMs or
C          was installed differently.
C     The numerical values of polarization are set to +-90.0 to be
C     consistent with the 'chirality' factor of the Dynasonde
C       -90.0  is Ordinary  ; +90.0 is eXtraordinary
C        (J.W. Wright, private communication)
C
C     Output tables are indexed by the channel number(1-16) and contain:
C       FRQ  - Sounding Frequency (MHz)
C       APOL - Rx Antenna Polarization Mode O (-90.0 )or X (+90.0)
C       HDOP - Doppler shift, Hz
C       AZ   - Rx antenna beam azimuth, degrees
C       EL   - Rx antenna beam elevation, degrees
C       PZA  - The PZA or 'Meaning' byte defined by UMLCAR (cf Galkin 2001)
C

==> dop_d256.f <==
C
C  =======================================================================
C
      SUBROUTINE DOP_D256(PREFACE,   NDOPP,DFR)
C
C     Determines the number and spacing (Doppler Frequency Resolutuion) of
C     the Ionogram Doppler bins.
C     Some principles:
C      - The maximum number of unique Doppler lines is 16 [-8..8]
C      - There is never a zero Doppler in a D256.
C      - The Doppler spectrum is always symmetric about zero

==> fixphase.f <==
C
C====+==================================================================+==
C
      SUBROUTINE FIXPHASE(RGT,EPHASE,MAXRBIN)
C
C     The Digisondes all perform complex (In-phase and Quadrature)
C     sampling of the receiver output by measuring the 225 kHz IF
C     signal twice at time separation equal to 90 degrees of phase
C     in the IF.  This is 1.11111 us.
C     A consequence of this is that if successive range gates are not
C     spaced at integer multiples of the IF period (4.444444us) then
C     there will be a phase difference induced by this sampling
C     at the IF.  For dR=1.000 km, this is a 6.6667us dt, or 1.50
C     IF cycles.  For dR=5.000 km, there are 7.500 IF cycles between
C     successive range gates, thus every other range gate is 180 degree
C     phase offset from its neighbors.
```

```
C
C     This routine corrects this, referencing all phases to the first
C     range gate and removing the 360 degree miltiples.
C

==> frq_d256.f <==
C
C====+==============================================================+==
C
      REAL FUNCTION FRQ_D256(PREFACE)
C
C     Function retuns the actual radio frequency used for sounding,
C     including the adjustments for frequency search, if enabled.
C     NOTE: This does NOT exactly work for closely spaced frequency mode
C     (aka Precision Group Height mode)
C

==> indexx.f <==
C
C =============================================================================
C
C     Sort the array ARRIN, Length N, returning the order of the elements
C     indexed in array INDX

==> irtype.f <==
C
C   =======================================================================
C
       INTEGER FUNCTION IRTYPE(IBUF)
C
C      This function takes the Digisonde 256/DPS data block in IBUF and
C      determines the actual data type (Ionogram, Ionogram continuation,
C      Drift, Artist, Raw Ionogram) and returns it as the function value.
C      If the data block is not one of these types (ie unknown) the
C      function returns a zero value for the recordtype.
C
C       NOTE: Due to a bug in the recording of 16 channel data in Artist3
C         circa 2001, where the record type is not encoded into the 4 lowest
C         bits of the first byte of every 8K block, this routine must be
C         specific to 16 channel data and also VERY FRAGILE!
C

==> pref_16c.f <==
C
C =============================================================================
C
      SUBROUTINE PREF_16C(IBUF,  IPREF)
C
C     Extract the preface encoded into the LSB of the amplitude data in the
C     16 channel data blocks.
C     Preface characters 1-31 found in bytes    5-128 for both data types
C     Preface characters 32-57 found in bytes 257-360 for IRTYPE=12
C     Preface characters 32-64 found in bytes 129-232 for IRTYPE=???
```

```
==> prf_d256.f <==
C
C  ========================================================================
C
      REAL FUNCTION PRF_D256(IR)
C
C     Determine the Pulse Repetition Frequency (PRF) of a D256
C     from the R preface parameter

==> pza2az.f <==
C
C====+==================================================================+==
C
      REAL FUNCTION PZA2AZ(PZA)
C
C     Compute the Azimuth Angle from the PZA byte.
C
C     Azimuth is in the low 4 bits of the 0PVZAAAA bits in PZA

==> pza2pol.f <==
C
C====+==================================================================+==
C
      REAL FUNCTION PZA2POL(PZA)
C
C     Calculate the polarization based on the PZA byte
C     The numerical values of polarization are set to +-90.0 to be
C     consistent with the 'chirality' factor of the Dynasonde
C        (J.W. Wright, private communication)
C
C     Polarizaion is in the 64's bit of the 0PVZAAAA bits in PZA
      INTEGER PZA

==> pza2zn.f <==
C====+==================================================================+==
C
      REAL FUNCTION PZA2ZN(PZA)
C
C     Compute the Zenith Angle from the PZA byte.
C
C     Zenith is in the 16's and 32's bit of the 0PVZAAAA bits in PZA
C

==> read4kb.f <==
C
C  =========================================================================
C
      LOGICAL FUNCTION READ4KB(IU,  IBUF)
C
C     Subroutine to read 4096 bytes from opened file unit IU
C     Returns the values in IBUF
C     Return value is TRUE if there is some problem, such as EOF
C     Specialized for Linux g77 compiler which does not easily support
C       binary files
```

```
==> rg_d256.f <==
C
C====+===============================================================+==
C
      SUBROUTINE RG_D256(PREFACE,  NRG, RGT)
C
C     Given the PREFACE, returns the number of range gates
C     and their values
C
C     The range gates are determined by preface parameters H and E
C     as defined in the Digisonde 256 "Green Book"

==> s2pza.f <==
C
C====+===============================================================+==
C
      INTEGER FUNCTION S2PZA(PREFACE,ISTAT)
C
C     Convert Status to PZA byte using the ZT tables, Corrected for L
C

==> sort_spectra.f <==
C
C====+===============================================================+==
C
      SUBROUTINE SORT_SPECTRA(SORT1,SORT2,SORT3,SORT4,SORT5,NSRT, ISEQ)
C
C     This subroutine takes 5 tables of signal characteristics,
C     typically Frequency, Polarization, Zenith, Azimuth and Doppler,
C     and returns the sequence of channel numbers that sorts these
C     keys in increasing order, from min to max value.
C     NSRT is the number (1-4) of the SORT's to perform
C     This is a simple formatting trick.
C
C     ***WARNING***  This code uses a simple multiplication trick to
C     rank the values relative to each other.  This seems weak and
C     prone to failure.
C

==> unpack_16c.f <==
C
C  =====================================================================
C
      LOGICAL FUNCTION UNPACK_16C(IBUF,
     +                    TIME,PREFACE,FREQ,EAMPS,EPHASE,RGT,MAXRBIN)
C
C     This subroutine takes a 4K block of 16 channel amplitude and phase
C     Digisonde 256 data provides the following output.
C        PREFACE - The decoded preface for this ionogram
C        TIME - Time of the observation: YYYY DDD HH:MM:SS
C        FREQ - The precise sounding frequency, in MHz.
C               *NOTE* This will be slightly wrong for PGH modes, but it
C               is still the frequency in the preface.
C        EAMPS(256,16) - Received amplitudes vs range gate and channel # ,
C                        in dB and corrected for all D256 processor settings
```

```
C                        but not for any antenna gains, cable losses, etc.
C        EPHASE(256,16)- Received phase vs range and channel #, in degrees
C        RGT(256) - Range Gate Table assigns a range in km to each range gate.
C        MAXRBIN -  Maximum range bin number (128 or 256)
C
C     Interpetation of the channel number in terms of polarization, Doppler,
C     and Rx antenna beam position (arrival angle) is left to another routine.


==> ztl_init.f <==
C
C  ========================================================================
C
      SUBROUTINE ZTL_INIT()
C
C     Initialize the ZT table
C     This routine sets the values for the ZT table used for
C     determining Rx antenna beams, etc.  It is initialized this way
C     to allow for a custom table to be loaded early in the main program,
C     perhaps from a file.
C
C     The Meaning byte is defined as 0PVZAAAA where
C        P = Polarization (0=Ordinary, 1=eXtraordinary)
C        V = Vertical  (0=Not vertical, 1 = Vertical)
C        Z = Zenith Angles (0 for small(11deg), 1 for Large (22deg))
C        AAAA = Azimuth Value (0-12)
C
C     ToDo: Allow this to optionally read from an UMLCAR format ZT file.
C
C     Source: Green Book table 5.9, P.98, Revised 11Jun86,5Nov87,28Dec88
C
```

Appendix B: Source Code