

**vj2pipe spectral processing utility**

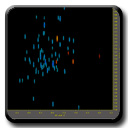
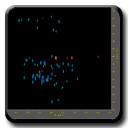
Last Updated 9/12/2014 frank\_delaglio@agilent.com

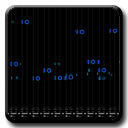
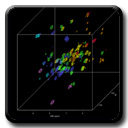
Agilent **vj2pipe** is a system of scripts and software for seamless processing and visualization of data from Agilent and Varian NMR spectrometers. The system builds scripts to process and display data via the widely-used [**NMRPipe Spectral Processing System.**](http://www.nmrscience.com/nmrpipe.html) This version of **vj2pipe** is configured for use with [**VnmrJ BioPack**](http://www.chem.agilent.com/en-US/Products/software/magneticresonancedatasystem/nmr/biopack/pages/default.aspx) for biomolecular data and [**VnmrJ ChemPack**](http://www.chem.agilent.com/Library/primers/Public/VnmrJ%203.1%20Exp%20Guide%205990-7603EN.pdf) for small molecule data. The parameters and software have been enhanced specifically to support the most convenient and effective use of Agilent and Varian NMR data together with NMRPipe. The **vj2pipe** system can be used from within Agilent's [**VnmrJ**](http://www.chem.agilent.com/en-US/Products/software/magneticresonancedatasystem/nmr/vnmrj3.0/pages/default.aspx) software. And, like the tools of NMRPipe, **vj2pipe** can also be used directly from a C-shell command-line.

In addition to creating processing schemes for conventionally sampled data, vj2pipe creates schemes for reconstructing **Non-Uniformly Sampled** (NUS) data. This saves weeks of learning time by automating the creation the complex scripts needed to exploit this data. This makes NUS accessible to everyone for measuring sharper spectra with less measurement time.

**vj2pipe** uses information from the VnmrJ spectral data (the fid file) and parameter file (the procpar file) to create and execute NMRPipe scripts which are then used to process and display the data. If processing parameters such as phase correction have already been selected, the processing and display can be entirely automated. Or, the scripts and data created by **vj2pipe** can be used interactively in the usual NMRPipe-style workflow.

As an initial step, **vj2pipe** reads the VnmrJ data parameters and builds a shell-script which converts the data from VnmrJ format to NMRPipe format. This script also establishes the chemical shift calibration information. Then, depending on the type of experiment, **vj2pipe** creates one or more additional scripts to process and display the spectral data. In the case of 3D experiments, these scripts will process and display the first XY and XZ 2D planes of the 3D data, as well as processing and displaying the entire 3D spectrum as a simple cube plot and in an interactive strip viewer.





**Using vj2pipe from the Command-Line**

The script vj2pipe.com can be used to run **vj2pipe** from a C-shell command-line. By default, **vj2pipe** will read files and parameters from the current directory. So, it is usually used from the directory containing the fid and procpar files. In its simplest form, it can be used without any additional command-line options. When used this way, it will automatically convert and process the data according to parameters in the procpar file, and display the result. For example, the following commands would automatically convert, process, and display the given 3D data:

cd ghn\_coca\_cb\_3D.fid

vj2pipe.com

In addition, **vj2pipe** can be used to list the options and parameters it determines for the current data, without creating or executing any scripts. This makes it possible to review the settings that will be used for the given data, and adjust these as desired via command-line options. The -info option is used to generate the informational display, for example:

cd ghn\_coca\_cb\_3D.fid

vj2pipe.com -info

An example of the informational output produced for a 3D spectrum is given at the bottom of this document. Besides showing the parameter settings extracted from the current data, the informational display shows the name of the NMRPipe-style command-line option which can be used to adjust that parameter.

**vj2pipe** performs the following steps in order:

* If an input data directory has been specified with the -cd option, change to that directory. Input for -procpar and -vjfid will be relative to this directory.
* If an output data directory has been specified with the -pipeDir option, create it if it does not exist, and change to that directory.
* Find the procpar file. If keyword Auto is used, **vj2pipe** will search for    ./procpar    acqfil/procpar   and   ../procpar in that order.
* Find the fid file. If keyword Auto is used, **vj2pipe** will search for    ./fid    acqfil/fid   and   ../fid in that order.
* Read the procpar file and extract all parameters in it.
* Replace or create procpar parameters with command-line values if any are given. This is done via the -pset option. Besides being one way to override values in procpar, this allows vj2pipe to be used with older Varian data that might be missing specific vj2pipe parameters. For example, to set the value used in place of the procpar parameter BPpipetype:

vj2pipe.com -pset BPpipetype HNCA

* If the spectrum is Non-Uniformly Sampled (NUS), **vj2pipe** will search for the sampling schedule ./sampling.sch ../sampling.sch or \*.hdr\_3 in that order, then read and interpret the contents.
* Interpret the procpar parameters to extract information about experiment type, data size, acquisition modes, storage details, axis labels, chemical shift calibration, and processing parameters. These are established as arguments for NMRPipe programs and scripts. Refer to the [Big NMRPipe Reference Page](http://www.nmrscience.com/ref/index.html) and other NMRPipe documentation for more details.
* Replace any of the NMRPipe parameters with command-line values if any are given. This allows control of the details of conversion and processing. For example, to reset the NMRPipe conversion parameter for the X-Axis label of the data:

vj2pipe.com -xLAB HMeth

* Besides the predefined NMRPipe parameters such as aqORD xN xT xLAB etc., additional parameter names can be defined via the -sVar -lVar and -dVar command-line options. This provides a way to make new parameter names available for use in scripts.
* Create a script which converts the Varian-format data to NMRPipe-format. This script uses the usual NMRPipe methods and arguments for format conversion, via the NMRPipe program var2pipe.
* Create one or more scripts for processing the spectral data. These are created from default **script templates** for particular tasks such as performing 3D Fourier processing. The default templates are stored in a directory of the NMRPipe installation ($NMRBASE/vj2pipe/com).
* New scripts are created by replacing text markers in the templates with the specific NMRPipe parameter settings for the given data. The text markers have the same names as the NMRPipe options, but begin and end with the underscore character. For example, the NMRPipe -xP0 parameter specifies the P0 (Zero Order Phase) value to apply when processing the X-Axis of the given data. So, the corresponding text marker for this option would be \_xP0\_ and this text marker will be replaced where ever it appears by the actual value for P0. For example, if a line from an input script template looked like this:

| nmrPipe -fn PS -p0 \_xP0\_ -p1 180.0 -di \   
  
then the text \_xP0\_ would be replaced by the value for P0, for example:   
  
 | nmrPipe -fn PS -p0 43.0 -p1 180.0 -di \

* If requested, create additional scripts for displaying the processed spectral data.
* If additional template script names have been specified via the -subs command-line option, create new versions of these replacing text markers in the same way as for the default scripts.
* Execute the conversion script.
* Execute any processing and display scripts created from the default templates. If any additional options were specified via the vj2pipe –args option, these will be passed as arguments to each script when it is executed. The same set of options will be passed to all scripts, even though all scripts might not use them.

As an example, scripts which perform IST (Iterative Soft Threshold) processing use an option -istIter … so, in order to pass this option, use:

vj2pipe.com –args –istIter 1024

The auxiliary script vj\_clean.com can be used to delete all processed data, and optionally all scripts created by **vj2pipe** in a given data directory. For safety, the script will only clean directories where file timestamp.vj2pipe is found, a file which is created when vj2pipe.com is used. Typical use will be as follows, use the –help argument for a complete list of options:

vj\_clean.com -com

**vj2pipe General Command-Line Options**

|  |  |  |
| --- | --- | --- |
| Option | Default | Description |
| -procpar pName | procpar | Name of VNMRJ procpar Input, or Keyword Auto. |
| -vjfid fName | fid | Name of VNMRJ binary fid Input, or Keyword Auto. |
| -cd cdDir | . | Default Data Directory for Input. |
| -pipeDir | . | Default Data Directory for Output. It will be created if it does not exist. |
| -outPrefix prefix | spec | File Name Prefix for Processed Results and Related Output Files. |
| -msgName mName | stderr | Output File for Status Messages, or Keyword stdout stderr None. |
| -vjDir vName | $NMRBASE/  vj2pipe | Directory of Auxiliary Files and Template Scripts used by vj2pipe. |
| -sVar sList | None | List of Additional User-Defined Single-Value Variables to Create. |
| -lVar lList | None | List of Additional User-Defined Multiple Value (List) Variables to Create. |
| -dVar dList | None | List of Additional User-Defined Dimension-Specific Variables to Create. |
| -subs sList | None | List of Additional User-Defined Template Scripts to Substitute. |
| -h2oM tSlope | -0.009552 | Slope of ppm/Degree K for Water, used in some calibration schemes. |
| -h2oB tInter | 5.011718 | PPM value for water at 273 degrees K, used in some calibration schemes. |
| -coThresh tPPM | 150.0000 | PPM threshold for identifying CO dimensions of 13C experiments. |
| -args argList | None | Optional arguments to pass to conversion and processing scripts when they are executed. Ignored when the –noexec option is used. |
| -fidScript fidComPrefix | vj\_fid | File Name Prefix for Conversion Script. |
| -ft1Script ft1ComPrefix | vj\_ft1 | File Name Prefix for 1D Processing Script. |
| -ft2Script ft2ComPrefix | vj\_ft2 | File Name Prefix for 2D Processing Script. |
| -xyScript xyComPrefix | vj\_xy | File Name Prefix for Processing Script for 2D XY Plane of 3D. |
| -xzScript xzComPrefix | vj\_xz | File Name Prefix for Processing Script for 2D XZ Plane of 3D. |
| -xyzScript xyzComPrefix | vj\_xyz | File Name Prefix for 3D Processing Script. |
| -scrollScript scrComPrefix | vj\_scroll | File Name Prefix for Strip Viewer Display Script for 3D Spectra. |
| -info |  | List All Parameters for the Current Data, then Exit. |
| -clean |  | Delete Previous Results and Scripts Before Proccessing. |
| -cleanOnly |  | Delete Previous Results and Scripts, then Exit. |
| -exec |  | Execute Conversion and Processing Scripts (Default). |
| -noexec |  | Create but Do Not Execute Conversion and Processing Scripts. |
| -partial | celem | Accommodate Partially-Completed Acquisition. |
| -nopartial |  | No Adjustment for Partially-Completed Acquisition (Default). |
| -graph |  | Display Spectral Graphics of the Results (Default). |
| -nograph |  | Do Not Display Spectral Graphics. |
| -drawPlane |  | Use NMRDraw to Display 2D Planes. |
| -nodrawPlane |  | Do Not Use NMRDraw to Display 2D Planes. |
| -drawCube |  | Draw 3D Contour Cube for 3D Data. |
| -nodrawCube |  | Do Not Draw 3D Contour Cube. |
| -drawStrips |  | Draw Strips for 3D Data. |
| -nodrawStrips |  | Draw 3D Strips. |
| -progress |  | Display a Progress Bar During Processing (Default). |
| -noprogress |  | Do Not Use Progress Bar Display. |
| -noskip |  | List Single-Point Dimensions, as for 3D experiments acquired as 2D. |
| -skip |  | Skip Listing of Single-Point Dimensions (Default). |
| -vjshow |  | List VJNMR-Specific Parameters. |
| -novjshow |  | Do Not List VJNMR-Specific Parameters. |
| -vjapod |  | Use VJNMR Window Function Values. *Not yet implemented.* |
| -novjapod |  | Use NMRPipe Window Function Values (Default). |
| -verb |  | Verbose Mode ON (Default). |
| -noverb |  | Verbose Mode OFF. |
| -help |  | List the Command-Line Options and Defaults. |
| -pset pName valList -- |  | Reset a procpar value. Can be used more than once. See below. |

**vj2pipe Data-Specific Command-Line Options**

The following options pertain to the parameters extracted specifically from the current data, and used to build conversion and processing schemes for the data. Options for specific parameters start with prefix -x -y -z etc ...

|  |  |  |  |
| --- | --- | --- | --- |
| Option | Example | Related **procpar** | Description |
| -title | "hn(co)ca" | seqfil BPpipetype | The canonical title of the experiment. |
| -dimension | C13 N15 H1 | dimension | Specified which nuclei are associated with a given dimension |
| -type | HN | tn dn dn2 ... | The general experiment type. Keywords: HN HC HOMO NMR. |
| -ndim | 3 | np ni ni2 ... | Number of dimensions in the data. |
| -nusDim | 3 | SAMPLING, sparse | Number of dimensions Non-Uniformly Sampled (NUS) data will have after it is expanded. |
| -aq2D | States | array | The NMRPipe 2D plane mode. Keywords: States Magnitude. |
| -aqORD | 1 | array | The order of loops for real,imaginary detection in the indirect dimensions. |
| -planeHi | 10% | BPpipedrawinfo | 2D Plane Contour Height, as Value, % or Fraction of Data @. |
| -hi | 2.50% | BPpipedrawinfo | 2D Spectrum Contour Height, as Value, % or Fraction of Data @. |
| -cubeHi | 10.0% | BPpipedrawinfo | 3D Cube Contour Height, as Value, % or Fraction of Data @. |
| -stripHi | 3.00% | BPpipedrawinfo | 3D Strip Contour Height, as Value, % or Fraction of Data @. |
| -projHi | 10.0% | BPpipedrawinfo | Peak detection threshold for analyzing 2D projections for 3D strip locations. |
| -temperature | 298.00 | temp | Temperature, degrees K. Used in some PPM calibration schemes. |
| -xN | 2048 | np ni ni2 ... | Number of total points (real + imagimary) stored for the given dimension. |
| -xT | 1024 | array np ni ni2 ... | Number of valid time-domain points (complex) in the dimension. |
| -xMODE | Complex | array f1coef … | Acquisition Mode for the dimension. Keywords: Complex Rance-Kay. |
| -xCHAN | 1 | dimension refsource... | Channel number the spectrometer associated with the dimension. |
| -xSW | 12019.2000 | sw sw1 sw2 ... | Spectral width, Hz. |
| -xAQTIME | 0.085197 | np ni ... sw sw1 ... | Effective acquisition time, Hz. |
| -xOBS | 599.8896 | sfrq dfrq dfrq2 ... | Observe Frequency at Zero ppm, MHz. |
| -xOBSMID | 599.8896 | sfrq dfrq dfrq2 ... | Observe Frequency at Center of Spectrum, MHz. |
| -xCAR | 4.6300 | obscenterppm h1ref... etc  pipexCAR… etc | PPM position of the center point of the spectrum (point N/2 + 1 of 1 to N). |
| -xCARSET | 0 | h1ref... etc | Indicates which procpar ref parameter (i.e. n15ref2) is set. |
| -xLAB | HN | BPpipetype tn dn ... | Axis label for the dimension. |
| -xNUC | H1 | dimension tn dn dn2 ... | Nucleus associated with the dimension. 1H 13C 15N. |
| -xGAMMA | 1.000000000 | dimension tn dn dn2 ... | Gamma value (gyromagnetic ratio). |
| -xP0 | -160.00 | rp rp1 rp2 ... | Zero-Order Phase, degrees. |
| -xP1 | -71.80 | lp lp1 lp2 ... | First-Order Phase, degrees. |
| -xAPOD | SP |  | NMRPipe Window Function Name.  SP EM GM GMB TM JMOD. |
| -xQ1 | 0.500 |  | Window Function Parameter 1. |
| -xQ2 | 0.950 |  | Window Function Parameter 2. |
| -xQ3 | 2.000 |  | Window Function Parameter 3. |
| -xELB | 0.000 |  | NMRPipe Window Function Exponential Broadening, Hz. |
| -xGLB | 0.000 |  | NMRPipe Window Function Gaussian Broadening, Hz. |
| -xGOFF | 0.000 |  | NMRPipe Window Function Gaussian Center, 0.0 (first point) to 1.0 (last point). |
| -xC1 | 1.000 | lp lp1 lp2 ... | First-Point Time Domain Scale. Usually 0.5 (when P1 phase is zero) or 1.0. |
| -xVJSB | -0.085 | sb sb1 sb2 ... | Original VnmrJ sb window value. |
| -xVJSBS | -0.085 | sbs sbs1 sbs2 ... | Original VnmrJ sbs window value. |
| -xVJLB | 5.000 | lb lb1 lb2 ... | Original VnmrJ lb window value. |
| -xVJGF | 0.022 | gf gf1 gf2 ... | Original VnmrJ gf window value. |
| -xVJGFS | 0.009 | gfs gfs1 gfs2 ... | Original VnmrJ gfs window value. |
| -xSOL | SOL |  | NMRPipe Solvent Filter Function to use. This argument is expandable by use of commas between options. |
| -xZFARG | zf=1,auto | BPpipeprocinfo | NMRPipe Zero-Fill Options. This argument is expandable by use of commas between options. |
| -xFTARG | None | BPpipeprocinfo | NMRPipe Fourier Transform Options. This argument is expandable by use of commas between options. |
| -xEXTX1 | 9.952ppm | sp wp BPpiperegion | Start Coord of Processed Region to Extract. Units pts Hz ppm %. |
| -xEXTXN | 5.423ppm | sp wp BPpiperegion | End Coord of Processed Region to Extract. Units pts Hz ppm %. |
| -xLSTIME | 0 | lsfid lsfid1 lsfid2 ... | Shift to Apply in the Frequency Domain. |
| -xLSFREQ | 0 | lsfrq lsfrq1 lsfrq2 ... | Shift to Apply in the Frequency Domain. |
| -xBASEARG | None | BPpipeprocinfo | NMRPipe Baseline Correction Options. This argument is expandable by use of commas between options. |

**Viewing and Setting vj2pipe Conversion and Processing Parameters**

Using the command vj2pipe.com -info or vj2pipe.com -info -vjshow will generate a list of the default conversion and processing parameters for the data in the current directory. An example of this output is shown below. The values for any of these parameters can be reset at the command-line. For example, to change the default setting for the -title option:

vj2pipe.com -title HNCA

As noted previously, it is also possible to adjust values from the procpar file before they are interpreted to generate NMRPipe parameters. This is done via the -pset option, followed by the name of the procpar parameter the corresponding value or value list, for example:

vj2pipe.com -pset BPpipetype HNCA

The -pset option can be used multiple times to reset more than one parameter. It can also be used with other options for adjusting parameters. In these cases, each use of the -pset option should be concluded by double-dashes -- for example:

vj2pipe.com -pset BPpipetype HNHA -- -pset h1ref3 4.55 -- -xLAB HN -yLAB HA

If no adjustments to procpar parameters are needed, a simple list of conversion and processing parameter settings or adjustments can be used. Commonly, these adjustments will be regions of interest and phase correction values:

vj2pipe.com –xEXT1 11.3ppm –xEXTN 5.2ppm –yP0 45 -xLAB HN -yLAB HA

As noted in the tables above, some options, including the processing arguments –xSOL –xZFARG –xFTARG and –xBASEARG are expandable. This means that when the associated values are used in processing scripts, they will be expanded at commas and equal signs to space-separated flags and argument/value pairs. For example:

-xZFARG zf=1,auto is expanded to –zf 1 –auto

-xSOL POLY,time is expanded to POLY –time

Example output for a 3D CBCANH experiment:

% vj2pipe.com –info –vjshow

-cd . -outPrefix spec -vjPath . -vjfid fid -procpar procpar \

-nusSched Auto -title CBCANH -type HN -ndim 3 -nusDim 0 -celem 8192 \

-arrayDim 8192 -aq2D States -aqORD 1 -temperature 298.00 \

-h2oM -0.009552 -h2oB 5.011718 -coThresh 150.0000 -fidScript vj\_fid \

-ft1Script vj\_ft1 -ft2Script vj\_ft2 -xyScript vj\_xy -xzScript vj\_xz \

-xaScript vj\_xa -xyzScript vj\_xyz -xyzaScript vj\_xyza \

-scrollScript vj\_scroll -tmpScript vj\_tmp -vjAuxDir /u/fd/vj2pipe \

-inName spec.fid -outName spec.dat -auxName aux.dat \

-hi 2.5% -stripHi 3.0% -cubeHi 10% -planeHi 10% -projHi Full \

-noise Auto -tdNoise Auto -ftNoise Auto -istTMult \_default\_ \

-istCMult \_default\_ -istIter \_default\_ -istMaxRes \_default\_ \

-dimension C13 N15 H1 -nusMethods clean irls ist mdd mem ml scrub \

-xN 2048 -yN 128 -zN 64 \

-xT 1024 -yT 64 -zT 32 \

-xMODE Complex -yMODE Complex -zMODE Rance-Kay \

-xCHAN 1 -yCHAN 2 -zCHAN 3 \

-xSW 13020.8000 -ySW 16084.5872 -zSW 2106.6000 \

-xAQTIME 0.078643 -yAQTIME 0.003979 -zAQTIME 0.015190 \

-xOBS 799.5969 -yOBS 201.0666 -zOBS 81.0318 \

-xOBSMID 799.5969 -yOBSMID 201.0666 -zOBSMID 81.0318 \

-xCAR 4.6330 -yCAR 45.9700 -zCAR 119.9800 \

-xCARSET 0 -yCARSET 1 -zCARSET 2 \

-xLAB HN -yLAB CACB -zLAB N \

-xNUC H1 -yNUC C13 -zNUC N15 \

-xGAMMA 1.000000000 -yGAMMA 0.251449530 -zGAMMA 0.101329118 \

-xP0 78.00 -yP0 0.00 -zP0 0.00 \

-xP1 0.00 -yP1 0.00 -zP1 0.00 \

-xP0EXTRA 0.00 -yP0EXTRA 0.00 -zP0EXTRA 0.00 \

-xP1EXTRA 0.00 -yP1EXTRA 0.00 -zP1EXTRA 0.00 \

-xAPOD SP -yAPOD SP -zAPOD SP \

-xQ1 0.500 -yQ1 0.500 -zQ1 0.500 \

-xQ2 0.950 -yQ2 0.950 -zQ2 0.950 \

-xQ3 2.000 -yQ3 1.000 -zQ3 1.000 \

-xELB 0.000 -yELB 0.000 -zELB 0.000 \

-xGLB 0.000 -yGLB 0.000 -zGLB 0.000 \

-xGOFF 0.000 -yGOFF 0.000 -zGOFF 0.000 \

-xC1 0.500 -yC1 0.500 -zC1 0.500 \

-xVJSB -0.079 -yVJSB -0.008 -zVJSB 0.061 \

-xVJSBS -0.079 -yVJSBS -0.008 -zVJSBS -0.061 \

-xVJLB 5.000 -yVJLB 0.318 -zVJLB 1.000 \

-xVJGF 0.040 -yVJGF 0.016 -zVJGF 0.000 \

-xVJGFS 0.000 -yVJGFS 0.000 -zVJGFS 0.000 \

-xDECAY 0.000 -yDECAY 0.000 -zDECAY 0.000 \

-xSOL SOL -ySOL None -zSOL None \

-xZFARG zf=1,auto -yZFARG zf=1,auto -zZFARG zf=1,auto \

-xFTARG None -yFTARG None -zFTARG None \

-xEXTX1 3% -yEXTX1 0% -zEXTX1 0% \

-xEXTXN 47% -yEXTXN 100% -zEXTXN 100% \

-xLSTIME 0 -yLSTIME 0 -zLSTIME 0 \

-xLSFREQ 0 -yLSFREQ 0 -zLSFREQ 0 \

-xBASEARG POLY,auto -yBASEARG NULL -zBASEARG NULL \

-xLPARG None -yLPARG None -zLPARG None \

Example output for a 3D Non-Uniformly Sampled Experiment:

% vj2pipe.com –info –vjshow

-cd . -outPrefix spec -vjPath . -vjfid fid -procpar procpar \

-nusSched sampling.sch -title HNCACB -type HN -ndim 3 \

-nusDim 0 -celem 1248 -arrayDim 1248 -mcFlag 0 -aq2D States \

-aqORD 1 -temperature 298.00 -h2oM -0.009552 -h2oB 5.011718 \

-coThresh 150.0000 -fidScript vj\_fid -ft1Script vj\_ft1 \

-ft2Script vj\_ft2 -xyScript vj\_xy -xzScript vj\_xz -xaScript vj\_xa \

-xyzScript vj\_xyz -xyzaScript vj\_xyza -scrollScript vj\_scroll \

-tmpScript vj\_tmp -vjAuxDir /u/delaglio/vj2pipe -inName spec.fid \

-outName spec.dat -auxName aux.dat -hi 2.5% -stripHi 3.0% \

-cubeHi 10% -planeHi 10% -projHi Full -noise Auto -tdNoise Auto \

-ftNoise Auto -istTMult \_default\_ -istCMult \_default\_ \

-istIter \_default\_ -istMaxRes \_default\_ \

-dimension C13 N15 H1 -nusMethods clean irls ist mdd mem ml scrub \

-xN 2404 -yN 1248 -zN 1 \

-xT 1202 -yT 48 -zT 26 \

-xMODE Complex -yMODE Complex -zMODE Real \

-xCHAN 1 -yCHAN 2 -zCHAN 3 \

-xSW 12019.2308 -ySW 12060.0022 -zSW 1320.0000 \

-xAQTIME 0.100006 -yAQTIME 0.003980 -zAQTIME 0.019697 \

-xOBS 599.4225 -yOBS 150.7307 -zOBS 60.7459 \

-xOBSMID 599.4225 -yOBSMID 150.7307 -zOBSMID 60.7459 \

-xCAR 4.6280 -yCAR 173.9700 -zCAR 127.2100 \

-xCARSET 0 -yCARSET 1 -zCARSET 2 \

-xLAB HN -yLAB CACB -zLAB N \

-xNUC H1 -yNUC C13 -zNUC N15 \

-xGAMMA 1.000000000 -yGAMMA 0.251449530 -zGAMMA 0.101329118 \

-xP0 334.84 -yP0 0.00 -zP0 0.00 \

-xP1 0.00 -yP1 0.00 -zP1 0.00 \

-xP0EXTRA 0.00 -yP0EXTRA 0.00 -zP0EXTRA 0.00 \

-xP1EXTRA 0.00 -yP1EXTRA 0.00 -zP1EXTRA 0.00 \

-xAPOD SP -yAPOD SP -zAPOD SP \

-xQ1 0.500 -yQ1 0.500 -zQ1 0.500 \

-xQ2 0.950 -yQ2 0.950 -zQ2 0.950 \

-xQ3 2.000 -yQ3 1.000 -zQ3 1.000 \

-xELB 0.000 -yELB 0.000 -zELB \_default\_ \

-xGLB 0.000 -yGLB 0.000 -zGLB \_default\_ \

-xGOFF 0.000 -yGOFF 0.000 -zGOFF \_default\_ \

-xC1 0.500 -yC1 0.500 -zC1 0.500 \

-xVJSB -0.100 -yVJSB -0.004 -zVJSB -0.020 \

-xVJSBS -0.090 -yVJSBS -0.061 -zVJSBS -0.029 \

-xVJLB 5.000 -yVJLB 0.318 -zVJLB 1.000 \

-xVJGF 0.046 -yVJGF 0.016 -zVJGF 0.000 \

-xVJGFS 0.000 -yVJGFS 0.000 -zVJGFS 0.000 \

-xDECAY 0.000 -yDECAY 0.000 -zDECAY 0.000 \

-xSOL SOL -ySOL None -zSOL None \

-xZFARG zf=1,auto -yZFARG zf=1,auto -zZFARG zf=1,auto \

-xFTARG None -yFTARG None -zFTARG None \

-xEXTX1 3% -yEXTX1 0% -zEXTX1 0% \

-xEXTXN 47% -yEXTXN 100% -zEXTXN 100% \

-xLSTIME 0 -yLSTIME 0 -zLSTIME 0 \

-xLSFREQ 0 -yLSFREQ 0 -zLSFREQ 0 \

-xBASEARG POLY,auto -yBASEARG NULL -zBASEARG NULL \

-xLPARG None -yLPARG None -zLPARG None \