# VnmrJ Command and Parameter Reference

Varian, Inc. NMR Systems With VnmrJ 2.2C Software Pub. No. 01-999346-00, Rev. B 1207



# VnmrJ Command and Parameter Reference

Varian, Inc. NMR Systems With VnmrJ 2.2C Software Pub. No. 01-999346-00. Rev. B 1207

NOTICE: Varian, Inc. was acquired by Agilent Technologies in May 2010. This document is provided as a courtesy but is no longer kept current and thus will contain historical references to Varian. For more information, go to **www.agilent.com/chem**.



VnmrJ Command and Parameter Reference Varian, Inc. NMR Systems With VnmrJ 2.2C Software Pub. No. 01-999346-00, Rev. B 1207

#### Revision history:

Rev A 0607 – Initial release for VnmrJ 2.2C, removed obsolete material Rev B 1207 – Added new material and updated existing material for VnmrJ 2.2C

Technical writer and editor: Everett Schreiber

Copyright ©2007 by Varian, Inc. 3120 Hansen Way, Palo Alto, California 94304 1-800-356-4437 www.varianinc.com All rights reserved. Printed in the United States.

The information in this document has been carefully checked and is believed to be entirely reliable. However, no responsibility is assumed for inaccuracies. Statements in this document are not intended to create any warranty, expressed or implied. Specifications and performance characteristics of the software described in this manual may be changed at any time without notice. Varian reserves the right to make changes in any products herein to improve reliability, function, or design. Varian does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others. Inclusion in this document does not imply that any particular feature is standard on the instrument.

Varian, Inc. NMR Spectrometer Systems, VnmrJ, VNMR, MAGICAL II, Magnex, AutoLock, AutoShim, AutoPhase, limNET, ASM, and SMS are registered trademarks or trademarks of Varian, Inc. Sun, Solaris, CDE, Suninstall, Ultra, SPARC, SPARCstation, SunCD, and NFS are registered trademarks or trademarks of Sun Microsystems, Inc. and SPARC International. Oxford is a registered trademark of Oxford Instruments LTD. Ethernet is a registered trademark of Xerox Corporation. VxWORKS and VxWORKS POWERED are registered trademarks of WindRiver Inc. Other product names in this document are registered trademarks or trademarks of their respective holders.

## **Contents Overview**

Nota	ational Conventions	
<b>A</b> .		
<b>B</b> .		
<b>C</b> .		
D.		
E.		199
F.		213
G.		245
Η.		273
1		293
J.		303
<b>K</b> .		307
L.		311
M.		341
N.		365
0.		377
<b>P</b> .		
Q,		461
R.		463
S.		499
<i>T</i> .		577
U.		597
W.		619
<b>X</b> .		641
Υ.		651
<b>Z</b> .		653

## **Table of Contents**

## **Notational Conventions**

- 4	

aa	Abort acquisition with error (C)	37
abort	Terminate action of calling macro and all higher macros (C)	37
abortallacqs	Reset acquisition computer in a drastic situation (C)	38
abortoff	Terminate normal functioning of abort in a macro (C)	38
aborton	Restore normal functioning of abort in a macro (C)	38
abs	Find absolute value of a number (C)	38
AC1S-AC11S	Autocalibration macros (M)	39
ACbackup	Make backup copy of current probe file (M)	39
acct	Writes records for operator login and logoff (M)	
ACreport	Print copy of probe file after autocalibration (M)	
acos	Find arc cosine of number (C)	
acosy	Automatic analysis of COSY data (C)	
acosyold	Automatic analysis of COSY data, old algorithm (C)	
acqdisp	Display message on the acquisition status line (C)	
acqi	Interactive acquisition display process (C)	
acqmeter	Open Acqueter window (M)	
Acqmeter	Open Acqueter window (U)	
acqmode	Acquisition mode (P)	
acqstat	Open Acquisition Status window (M)	
Acqstat	Open Acquisition Status window (U)	
acqstatus	Acquisition status (P)	
acquire	Acquire data (M)	
actionid	Current study queue node id (P)	
activestudy	Active study name (P)	
add	Add current FID to add/subtract experiment (C)	
addi	Start interactive add/subtract mode (C)	
addnucleus	Add new nucleus to existing probe file (M)	
addpar	Add selected parameters to current experiment (M)	
addparams	Add parameter to current probe file (M)	
addprobe	Create new probe directory and probe file (M)	
_	Automatic DEPT analysis and spectrum editing (C)	
adept		
aexppl ai	Automatic plot of spectral expansion (M)	
aig	Absolute-intensity group (P)	
alfa alock	Set alfa delay before acquisition (P)	
	Automatic lock control (P)	
ampmode	Independent control of amplifier mode (P)	
amptype	Amplifier type (P)	
analyz	Calculate standard peak height (M)	
analyze	Generalized curve fitting (C)	
ap	Print out "all" parameters (C)	
ap	"All" parameters display control (P)	
apa	Plot parameters automatically (M)	
aph	Automatic phase adjustment of spectra (C)	
aph0	Automatic phase of zero-order term (C)	
aphb	Auto phasing for Bruker data (C)	
aphx	Perform optimized automatic phasing (M)	
appdirs	Starts Applications Directory Editor (M)	
appmode	Application mode (P)	
apptype	Application type (P)	
Apt	Set up parameters for APT experiment (M)	
aptaph	Automatic processing for APT spectra (M)	
array	Easy entry of linearly spaced array values (M)	
arrav	Parameter order and precedence (P)	61

arraydim	Dimension of experiment (P)	
asin	Find arc sine of number (C)	
asize	Make plot resolution along $f_1$ and $f_2$ the same (M)	
assign	Assign transitions to experimental lines (M)	
at	Acquisition time (P)	
atan	Find arc tangent of a number (C)	
atan2	Find arc tangent of two numbers (C)	
atcmd	Call a macro at a specified time (M)	
atext	Append string to current experiment text file (M)	
attval	Calculate pulse width (M)	
atune	ProTune Present (P)	
au	Submit experiment to acquisition and process data (M)	
AuCALch3i	Set up autocalibration with CH3I sample (M)	
AuCALch3i1	Get autocalibration with CH <sub>3</sub> I sample (M)	
AuCALch3oh	Set up autocalibration with Autotest sample (M)	
AuCALch3oh1	Get autocalibration with Autotest sample (M)	
Aucalibz0	Automatic Hz to DAC calibration for Z0 (M)	
AuCdec	Carbon decoupler calibration macro (M)	
AuCgrad	Carbon/proton gradient ratio calibration macro (M)	
AuCobs	Carbon observe calibration macro (M)	
audiofilter	Audio filter board type (P)	
Aufindz0	Automatic adjustment of Z0 (M)	
Augcal	Probe gcal calibration macro (M)	
Augmap	Automated gradient map generation (M)	
Augmapz0	Automatic lock gradient map generation and z0 calibration (M)	68
AuHdec	Proton decoupler calibration (M)	68
AuHobs	Proton observe calibration macro (M)	69
Aumakegmap	Auto lock gradient map generation (M)	69
AuNuc	Get parameters for a given nucleus (M)	69
auto	Prepare for an automation run (C)	69
auto	Automation mode active (P)	69
auto_au	Controlling macro for automation (M)	70
Autobackup	Back up current probe file (M)	
autodept	Automated complete analysis of DEPT data (M)	70
autodir	Automation directory absolute path (P)	71
autogo	Start automation run (C)	71
autolist	Set up and start chained acquisition (M)	71
autoname	Create path for data storage (C)	72
autoname	Prefix for automation data file (P)	
autora	Resume suspended automation run (C)	
autosa	Suspend current automation run (C)	
autoscale	Resume autoscaling after limits set by scalelimits macro (M)	
autostack	Automatic stacking for processing and plotting arrays (M)	
autotest	Open Auto Test Window (C)	
autotime	Displays approximate time for automation (M)	
av	Set abs. value mode in directly detected dimension (C)	
av1	Set abs. value mode in 1st indirectly detected dimension (C)	
av2	Set abs. value mode in 2nd indirectly detected dimension (C)	
averag	Calculate average and standard deviation of input (C)	
awc	Additive weighting const. in directly detected dimension (P)	78
awc1	Additive weighting const. in 1st indirectly detected dimension (P	').78
awc2	Additive weighting const. in 2nd indirectly detected dimension (	P) 78
axis	Provide axis labels and scaling factors (C)	
axis	Axis label for displays and plots (P)	
axisf	Axis label for FID displays and plots (P)	
В		
bandinfo	Shaped pulse information for calibration (M)	81
banner	Display message with large characters (C)	
bc	1D and 2D baseline correction (C)	
boonoff	Turn began off (C)	83

haanan	Turn beener on (C)	92
beepon bigendian	Turn beeper on (C)  Determine system byte order (C)	
binom	Set up parameters for BINOM pulse sequence (M)	84
bioref	Bio-NMR Referencing (P)	84
bootup	Macro executed automatically (M)	
box	Draw a box on a plotter or graphics display (C)	
boxes	Draw boxes selected by the mark command (M)	
bpa	Plot boxed parameters (M)	
br24	Set up parameters for BR24 pulse sequence (M)	
bs	Block size (P)	
C		
C		
c13	Automated carbon acquisition (M)	
c13p	Process 1D carbon spectra (M)	
calcdim	Calculate dimension of experiment (C)	
calfa	Recalculate alfa so that first-order phase is zero (M)	
calibflag	Correct systematic errors in DOSY experiments (P)	
calibrate	Start a dialog for autocalibration routines (M)	
callacq	Utility macro to call Acq command (M)	
capt	Automated carbon and APT acquisition (M)	
Carbon	Set up parameters for 13C experiment (M)	
cat	Display one or more text files in text window (C)	
cattn	Coarse attenuator type (P)	
cd	Change working directory (C)	92
cdc	Cancel drift correction (C)	
cdept	Automated carbon and DEPT acquisition (M)	
cdump	Prints the current graphics screen (M)	
celem	Completed FID elements (P)	
center	Set display limits for center of screen (C)	
centersw	Move cursor to center of spectrum (M)	
centersw1	Move cursor to center of spectrum in 1st indirect dimension (M)	
centersw2	Move cursor to center of spectrum in 2nd indirect dimension (M).	
cexp	Create an experiment (M)	
cf	Current FID (P)	94
cfpmult	Calculate first-point multiplier for 2D experiments (M)	
change	Submit a change sample experiment to acquisition (M)	
checkstring	Find and replace unwanted characters (C)	
chiliConf	Control flag set by ecc_on and ecc_off (P)	
Cigar2j3j	Convert the parameter to a CIGAR2j3j experiment (M)	
cla	Clear all line assignments (M)	
cla	Calculated transition number (P)	
clamp	Calculated transition amplitude (P)	
cleanexp	Remove old files and directories from an experiment (M)	
clear	Clear a window (C)	
cleardosy	Delete temporarily saved data in current sub experiment (M)	
clfreq	Calculated transition frequency (P)	98
clindex	Index of experimental frequency of a transition (P)	
clradd	Clear add/subtract experiment (C)	
color	Select plotting colors from a graphical interface (M)	
combiplate	View a color map for visual analysis of VAST microtiter plate (U)	
combishow	Display regions (red, green, and blue) in CombiPlate window (M)	
compressfid	Compress double-precision FID data (M,U)	
config	Display current configuration and possibly change it (M)	
confirm	Confirm message using the mouse (C)	
Console	System console type (P)	
contact_time	MAS cross-polarization spin-lock contact time (M)	
continueMovie	Continue movie in either forward or backward direction (C)	
convert	Convert data set from a VXR-style system (M,U)	
convertbru	Convert Bruker data (M,U)	
copy	Copy a file (C)	107
COS	Find cosine value of an angle (C)	107

Cosy	Convert the parameter to a COSY experiment (M)	
cosyps	Set up parameters for phase-sensitive COSY pulse sequence (M).	
ср	Copy a file (C)	
ср	Cycle phase (P)	
cpmgt2	Set up parameters for CPMGT2 pulse sequence (M)	
cpos_cvt	Convert data set from a VXR-style system (M,U)	
cptmp	Copy experiment data into experiment subfile (M)	
cpx	Create pbox shape file (M)	
cqexp	Load experiment from protocol (M)	
cqfindz0	Run an experiment to find the value of z0 (M)	
cqgmap	Perform gradient shimming utility functions (M)	
cqinit	Initialize liquids study queue (M)	
cqpars	Create study queue parameters for liquids (M)	
cqplot	Macro to perform generic 2D plot (M)	
cqprotocol	Macro to create protocols (M)	
cqreset	Reset study queue parameters (M)	
cqsavestudy	Macro to save study queue parameters (M)	
cqwtmenu	Macro to set weighting functions from a panel (M)	
cr	Cursor position in directly detected dimension (P)	
crl	Cursor position in 1st indirectly detected dimension (P)	
cr2	Cursor position in 2nd indirectly detected dimension (P)	
crcom	Create user macro without using text editor (M)	
create	Create new parameter in a parameter tree (C)	
createqcomp	Create qcomp parameter (M)	
crf	Current time-domain cursor position (P)	113
crl	Clear reference line in directly detected dimension (M)	114
crl1	Clear reference line in 1st indirectly detected dimension (M)	114
crl2	Clear reference line in 2nd indirectly detected dimension (M)	114
crmode	Current state of the cursors in df, ds, or dconi programs (P)	114
crof2	Recalculate rof2 so that $lp = 0$ (M)	115
cryo_noisetest	Run Cold Probe conditioning experiments (M)	115
cryoclient	Start the CryoBay Monitor program (M, U)	
ct	Completed transients (P)	115
ctext	Clear the text of the current experiment (C)	
curexp	Current experiment directory (P)	116
curscan	Scan currently in progress (P)	
curwin	Current window (P)	116
cutoff	Data truncation limit (P)	116
cyclenoe	Set up parameters for CYCLENOE pulse sequence (M)	117
cylbr24	Set up parameters for cycled BR24 pulse sequence (M)	117
cylmrev	Set up parameters for cycled MREV8 pulse sequence (M)	117
CZ	Clear integral reset points (C)	117
D		
d0	Overhead delay between FIDs (P)	123
d1	First delay (P)	
d2	Incremented delay in 1st indirectly detected dimension (P)	
d2pul	Set up parameters for D2PUL pulse sequence (M)	
d3	Incremented delay for 2nd indirectly detected dimension (P)	
d4	Incremented delay for 3rd indirectly detected dimension (P)	
	Store gradient calibration value in DOSY sequences (P)	
DAC_to_G da	Display acquisition parameter arrays (C)	
	Increment for t1 dependent first-order phase correction (P)	
daslp date	Date (P)	
daxis	Display horizontal LC axis (M)	
	Set up parameters for Dbppste pulse sequence (M)	
Dbppste	Set up parameters for Doppste pulse sequence (M)	
Dbppsteinept	Set up VnmrJ database (U)	
dbsetup		
dbupdate dc	Update the VnmrJ database (U)	
	Apply drift correction to 2D spectra (C)	120
dc2d	Appry unit confection to 2D spectra (C)	120

dcg	Drift correction group (P)	128
dcon	Display noninteractive color intensity map (C)	128
dconi	Interactive 2D data display (C)	129
dconi	Control display selection for the dconi program (P)	
dconn	Display color intensity map without screen erase (C)	
dcrmv	Remove dc offsets from FIDs in special cases (P)	132
ddf	Display data file in current experiment (C)	132
ddff	Display FID file in current experiment (C)	
ddfp	Display phase file in current experiment (C)	
ddif	Synthesize and show DOSY plot (C)	
ddrcr	Direct digital receiver coefficient ratio (P)	
ddrpm	Set ddr precession mode (P)	
ddrtc	Set ddr time constant (P)	
dds	Default display (M)	
dds seqfil	Sequence-specific default display (M)	
debug	Trace order of macro and command execution (C)	
decasynctype	Select the type of decoupler asynchronous mode (P)	
deccwarnings	Control reporting of DECC warnings from PSG (P)	
decomp	Decompose a VXR-style directory (M)	
def osfilt	Default value of osfilt parameter (P)	
defaultdir	Default directory for Files menu system (P)	
delcom	Delete a user macro (M)	
delete	Delete a file, parameter directory, or FID directory (C)	
delexp	Delete an experiment (M)	
deletenucleus	Removes nucleus entry from current probe file (M)	
dels	Delete spectra from $T_1$ or $T_2$ analysis (C)	
delta	Cursor difference in directly detected dimension (P)	
delta1	Cursor difference in 1st indirectly detected dimension (P)	
delta2	Cursor difference in 2nd indirectly detected dimension (P)	
deltaf	Difference of two time-domain cursors (P)	
Dept	Set up parameters for DEPT experiment (M)	
deptgl	Set up parameters for DEPTGL pulse sequence (M)	
deptproc	Process array of DEPT spectra (M)	
destroy	Destroy a parameter (C)	
destroygroup	Destroy parameters of a group in a tree (C)	
df	Display a single FID (C)	141
df2d	Display FIDs of 2D experiment (C)	
dfid	Display a single FID (C)	
dfmode	Current state of display of imaginary part of a FID (P)	
dfrq	Transmitter frequency of first decoupler (P)	
dfrq2	Transmitter frequency of second decoupler (P)	
dfrq3	Transmitter frequency of third decoupler (P)	
dfrq4	Transmitter frequency of fourth decoupler (P)	
dfs	Display stacked FIDs (C)	
dfsa	Display stacked FIDs automatically (C)	144
dfsan	Display stacked FIDs automatically without screen erase (C)	
dfsh	Display stacked FIDs horizontally (C)	
dfshn	Display stacked FIDs horizontally without screen erase (C)	
dfsn	Display stacked FIDs without screen erase (C)	
dfww	Display FIDs in whitewash mode (C)	
dg	Display group of acquisition/processing parameters (C)	
dg	Control dg parameter group display (P)	
	Display group of display parameters (M)	
dg1	Control dg1 parameter group display (P)	
dg1		
dg2	Display group of 3rd and 4th rf channel/3D parameters (M)	
dg2	Control dg2 parameter group display (P)	
dga	Display group of spin simulation parameters (M)	
DgcsteSL	Set up parameters for DgcsteSL pulse sequence (M)	
Dgcstecosy	Set up parameters for Decetebrage pulse sequence (M)	
Dgcstehmqc	Set up parameters for Dgcstehmqc pulse sequence (M)	
dglc	Display group of LC-NMR parameters (M)	
dglc	Control dglc parameter group display (P)	148

dglp	Display group of linear prediction parameters (C)	
dgs	Display group of shims and automation parameters (M)	
dgs	Control dgs parameter group display (P)	
dhp	Decoupler high-power control with class C amplifier (P)	
dialog	Display a dialog box from a macro (C)	
diffparams	Report differences between two parameter sets (U)	
diffshims	Compare two sets of shims (M,U)	
digfilt	Write digitally filtered FIDs to another experiment (M)	
dir	List files in directory (C)	
display	Display parameters and their attributes (C)	
dla	Display spin simulation parameter arrays (M)	
dlalong	Long display of spin simulation parameter arrays (C)	
dli	Display list of integrals (C)	
dlivast	Produce text file and process wells (M)	
dll	Display listed line frequencies and intensities (C)	
dlni	Display list of normalized integrals (M)	
dlp	Decoupler low-power control with class C amplifier (P)	
dm	Decoupler mode for first decoupler (P)	154
dm2	Decoupler mode for second decoupler (P)	
dm3	Decoupler mode for third decoupler (P)	
dm4	Decoupler mode for fourth decoupler (P)  Decoupler modulation frequency for first decoupler (P)	
dmf dmf2	Decoupler modulation frequency for first decoupler (P)	
dmf3	Decoupler modulation frequency for third decoupler (P)	
dmf4	Decoupler modulation frequency for fourth decoupler (P)	
dmfadj	Adjust tip-angle resolution time for first decoupler (M)	
dmf2adj	Adjust tip-angle resolution time for first decoupler (M)	
dmf3adj	Adjust tip-angle resolution time for second decoupler (M)	
dmf4adj	Adjust tip-angle resolution time for fourth decoupler (M)	
dmg	Data display mode in directly detected dimension (P)	
dmg1	Data display mode in 1st indirectly detected dimension (P)	
dmg2	Data display mode in 2nd indirectly detected dimension (P)	
dmgf	Absolute-value display of FID data or spectrum in acqi (P)	
dmm	Decoupler modulation mode for first decoupler (P)	
dmm2	Decoupler modulation mode for second decoupler (P)	
dmm3	Decoupler modulation mode for third decoupler (P)	
dmm4	Decoupler modulation mode for fourth decoupler (P)	
dn	Nucleus for first decoupler (P)	
dn2	Nucleus for second decoupler (P)	
dn3	Nucleus for third decoupler (P)	
dn4	Nucleus for fourth decoupler (P)	
dndfid	Retrieve and process fid data from the locator (M)	
dndjoin	Join a work space from the locator (M)	
dndpar	Retrieve a parameter set from the locator (M)	
dndshims	Retrieve a shimset set from the locator (M)	
dnode	Display list of valid limNET nodes (M,U)	
doautodialog	Start a dialog window using def file (M)	
dodialog	Start a dialog window with dialoglib file (M)	
dof	Frequency offset for first decoupler (P)	
dof2	Frequency offset for second decoupler (P)	
dof3	Frequency offset for third decoupler (P)	. 165
dof4	Frequency offset for fourth decoupler (P)	. 165
Doneshot	Set up parameters for Doneshot pulse sequence (M)	
dopardialog	Start a dialog with dialoglib/experiment def file (M)	
do_pcss	Calculate proton chemical shifts spectrum (C)	
dosy	Process DOSY experiments (M)	
dosy2d	Apptype macro for dosy 2D experiments (M)	
dosyfrq	Larmor frequency of phase encoded nucleus in DOSY (P)	
dosygamma	Gyromagnetic constant of phase encoded nucleus in DOSY (P)	
dosytimecubed	Gyromagnetic constant of phase encoded nucleus in DOSY (P)	. 167
dot1	Set up a $T_1$ experiment (M)	. 167
dotflag	Display FID as connected dots (P)	. 167

downsamp	Downsampling factor applied after digital filtering (P)	168
dp	Double precision (P)	168
dpcon	Display plotted contours (C)	
dpconn	Display plotted contours without screen erase (C)	
dpf	Display peak frequencies over spectrum (C)	
dpir	Display integral amplitudes below spectrum (C)	
dpirn dpiv	Display normalized integral amplitudes below spectrum (M)  Display integral values below spectrum (M)	
dpivn	Display normalized integral values below spectrum (M)	
dpl	Default plot (M)	
dpl seqfil	Sequence-specific default plot (M)	
dplane	Display a 3D plane (M)	
dpr	Default process (M)	
dpr seqfil	Sequence-specific default process (M)	
dprofile	Display pulse excitation profile (M)	
dproj	Display a 3D plane projection (M)	
dps	Display pulse sequence (C)	173
dpwr	Power level for first decoupler with linear amplifier (P)	. 174
dpwr2	Power level for second decoupler with linear amplifier (P)	
dpwr3	Power level for third decoupler with linear amplifier (P)	
dpwr4	Power level for fourth decoupler amplifier (P)	
dpwrf	First decoupler fine power (P)	
dpwrf2	Second decoupler fine power (P)	
dpwrf3	Third decoupler fine power (P)	
dpwrm	First decoupler linear modulator power (P)	
dpwrm2	Second decoupler linear modulator power (P)	
dpwrm3	Third decoupler linear modulator power (P)	
Dqcosy draw	Convert the parameter to a DQCOSY experiment (M)	
draw	Measure linewidth and digital resolution (C)	
dres	Tip-angle resolution for first decoupler (P)	
dres2	Tip-angle resolution for second decoupler (P)	
dres3	Tip-angle resolution for third decoupler (P)	
dres4	Tip-angle resolution for fourth decoupler (P)	
ds	Display a spectrum (C)	
ds2d	Display 2D spectra in whitewash mode (C)	
ds2dn	Display 2D spectra in whitewash mode without screen erase (C)	
dsnarray	Report statistical signal-to-noise for Cold Probes (M)	
dscale	Display scale below spectrum or FID (C)	. 182
dscoef	Digital filter coefficients for downsampling (P)	
dseq	Decoupler sequence for first decoupler (P)	183
dseq2	Decoupler sequence for second decoupler (P)	
dseq3	Decoupler sequence for third decoupler (P)	
dseq4	Decoupler sequence for fourth decoupler (P)	
dsfb	Digital filter bandwidth for downsampling (P)	
dshape	Display pulse shape or modulation pattern (M)	
dshapef	Display last generated pulse shape (M)	
dshapei	Display pulse shape or modulation pattern interactively (M)	
dshim	Display a shim "method" string (M)	
dslsfrq	Bandpass filter offset for downsampling (P)	
dsn	Measure signal-to-noise (C)	
dsnmax dsp	Calculate maximum signal-to-noise (M)	
dsp	Type of DSP for data acquisition (P)	
dsplanes	Display a series of 3D planes (M)	
dsptanes	Type of DSP (P)	
dss	Display stacked spectra (C)	
dssa	Display stacked spectra automatically (C)	
dssan	Display stacked spectra automatically without erasing (C)	
dssh	Display stacked spectra horizontally (C)	
dsshn	Display stacked spectra horizontally without erasing (C)	
dssl	Label a display of stacked spectra (M)	
	<del>-</del>	

dssn	Display stacked spectra without screen erase (C) 1	95
dsvast	Display VAST data in a stacked 1D-NMR matrix format (M) 1	95
dsvast2d	Display VAST data in a pseudo-2D format (M)1	
dsww	Display spectra in whitewash mode (C)	
dtext	Display a text file in graphics window (M)	
dtrig	Delay to wait for another trigger or acquire a spectrum (P) 1	97
dutyc	Duty cycle for homodecoupling (optional) (P) 1	97
_		
E		
e	Eject sample (M)1	99
eaddr	Display Ethernet address (M,U)	
ecc on	Turns on eddy current compensation for Cold Probes (M)	
ecc_off	Turns off eddy current compensation for Cold Probes (M)	
echo	Display strings and parameter values in text window (C)	.00
edit	Edit a file with user-selectable editor (M)	.00
eject	Eject sample (M)	.01
elist	Display directory on remote VXR-style system (M,U)2	.01
email	Email address (P)	
enter	Enter sample information for automation run (M,U)2	.02
enterdialog	Start a dialog window using enterexp file (M)	
eread	Transfer file from remote source (M,U)	
ernst	Calculate the Ernst angle pulse (C)	
errlog	Display recent error messages (C)	
errloglen	Number of lines in error message display (P)2	
ewrite	Transfer file to remote destination (M,U)	
exec	Execute a command (C)	
execpars	Set up the exec parameters (M)	
execplot	Execute plotting macro (P)	
execprep	Execute prepare macro (P)	
execprescan	Execute prescan macro (P)	
execproc	Execute processing macro (P)	
execprocess	Execute processing macro (P)	
execsetup	Execute setup macro (P)	
exists	Checks if parameter, file, or macro exists and file type (C	
exit	Call the vnmrexit command (M)	
exp	Find exponential value of a number (C)	
expactive	Determine if experiment has active acquisition (C)	
expfit	Make least-squares fit to polynomial or exponential curve (U) 2	
expl expladd	Display exponential or polynomial curves (C)	
expladd	Display experiment library (M)	
<del>-</del>	Display current experiment chain and approx. time for each (M)2	
explist explog	Display log file for experiment (M)	
explog	Display rog file for experiment (w) 2  Display experiment time (C) 2	
experime	Display experiment time (c)	12
F		
f	Sat display parameters to full spectrum (C)	1.4
<del>-</del>	Set display parameters to full spectrum (C)	
f19	Automated fluorine acquisition (M) 2 Process 1D fluorine spectra (M) 2	
f19p f1coef	Coefficient to construct F1 interferogram (P)	
f2coef	Coefficient to construct F1 interferogram (P)	
fattn	Fine attenuator (P)2	
fb	Filter bandwidth (P) 2	
fbc	Apply baseline correction for each spectrum in an array (M) 2	
fdm1	Set, write 1D FDM parameters, run FDM (M)	
fiddc3d	3D time-domain dc correction (P)	
fiddle	Perform reference deconvolution (M)2	
fiddled	Perform reference deconvolution (NI)  Perform reference deconvolution subtracting alternate FIDs (C) 2	
fiddleu	Perform reference deconvolution subtracting anternate FIDs (C) 2	
fiddle2d	Perform 2D reference deconvolution (C)	
fiddle2D	Perform 2D reference deconvolution (C)	20
		-

fiddle2dd	2D reference deconvolution subtracting alternate FIDs (C)	220
fiddle2Dd	2D reference deconvolution subtracting alternate FIDs (C)	
fidmax	Find the maximum point in an FID (C)	
fidpar	Add parameters for FID display in current experiment (M)	
fidsave	Save data (M)	
fifolpsize	FIFO loop size (P)	
file	File name of parameter set (P)	
files	Interactively handle files (C)	
filesinfo	Return file information for files display (C)	
filtfile	File of FIR digital filter coefficients (P)	
findxmlmenu	Find an xml menu (M)	
fitspec	Perform spectrum deconvolution (C, U)	
fixgrd	Convert gauss/cm value to DAC (M)	
fixpar	Correct parameter characteristics in experiment (M)	
fixpar3rf	Create parameters for third rf channel (M)	224
fixpar4rf	Create parameters for fourth rf channel (M)	224
fixpar5rf	Create parameters for fifth rf channel (M)	
fixup	Adjust parameter values selected by setup macros (M)	225
fixpsg	Update psg libraries (M)	225
flashc	Convert compressed 2D data to standard 2D format (C)	225
flipflop	Set up parameters for FLIPFLOP pulse sequence (M)	226
Fluorine	Set up parameters for 19F experiment (M)	226
flush	Write out data in memory (C)	226
fn	Fourier number in directly detected dimension (P)	227
fn1	Fourier number in 1st indirectly detected dimension (P)	
fn2	Fourier number in 2nd indirectly detected dimension (P)	227
fn2D	Fourier number to build up 2D DOSY display in freq. domain (I	P) 227
focus	Send keyboard focus to input window (C)	228
foldcc	Fold INADEQUATE data about two-quantum axis (C)	228
foldj	Fold J-resolved 2D spectrum about f <sub>1</sub> =0 axis (C)	228
foldt	Fold COSY-like spectrum along diagonal axis (C)	
fontselect	Open FontSelect window (C)	
format	Format a real number or convert a string for output (C)	
fp	Find peak heights or phases (C)	
fpmult	First point multiplier for np FID data (P)	
fpmult1	First point multiplier for ni interferogram data (P)	
fpmult2	First point multiplier for ni2 interferogram data (P)	
fr	Full recall of a display parameter set (M)	
fread	Read parameters from file and load them into a tree (C)	
fsave	Save parameters from a tree to a file (C)	
fsq	Frequency-shifted quadrature detection (P)	
ft	Fourier transform 1D data (C)	
ft1d	Fourier transform along f <sub>2</sub> dimension (C)	
ft1da	Fourier transform phase-sensitive data (M)	
ft1dac	Combine arrayed 2D FID matrices (M)	
ft2d	Fourier transform 2D data (C)	
ft2da	Fourier transform phase-sensitive data (M)	
ft2dac	Combine arrayed 2D FID matrices (M)	
ft3d	Perform a 3D Fourier transform on a 3D FID data set (M,U)	
full	Set display limits for a full screen (C)	
fullsq	Display largest square 2D display (M)	
fullt	Set display limits for a full screen with room for traces (C)	244
G		
		246
g2pul_ecc	Setup macro for eddy current compensation parameters (M)	
ga	Submit experiment to acquisition and FT the result (M)	
gain	Receiver gain (P)	
gap	Find gap in the current spectrum (M)	
gaussian	Set up unshifted Gaussian window function (M)	
gcal	Gradient calibration constant (P)	
gcoil	Current gradient coil (P)	449

Gcosy	Convert the parameter to a gradient COSY experiment (M)	. 250
gdiff	Diffusion gradient level (P)	
Gdqcosy	Convert the parameter to a gradient DQCOSY experiment (M)	
get1d	Select a 1D experiment for processing (M)	
get2d	Select a 2D experiment for processing (M)	
getdim	Return dimensionality of experiment (M)	251
getfile	Get information about directories and files (C)	
getlimit	get the limits of a variable in a tree (C)	. 252
getll	Get intensity and line frequency of line (C)	
getparam	Retrieve parameter from probe file (M)	
getplane	Extract planes from a 3D spectral data set (M)	253
getreg	Get frequency limits of a specified region (C)	
getsn	Get signal-to-noise estimate of a spectrum (M)	
gettoken	Utility macro to separate a string into tokens (M)	
gettxt	Get text file from VnmrJ data file (C)	
gettype	Get the type of a variable (C)	
getvalue	Get value of parameter in a tree (C)	
gf	Prepare parameters for FID/spectrum display in acqi (M)	
gf	Gaussian function in directly detected dimension (P)	
gf1	Gaussian function in 1st indirectly detected dimension (P)	
gf2	Gaussian function in 2nd indirectly detected dimension (P)	
gflow	Flow encoding gradient level (P)	
gfs	Gaussian shift const. in directly detected dimension (P)	
gfs1	Gaussian shift const. in 1st indirectly detected dimension (P)	
gfs2	Gaussian shift const. in 2nd indirectly detected dimension (P)	
Ghmbc	Convert the parameter to a gradient HMBC experiment (M)	
ghmqc	Set up a PFG HMQC pulse sequence (M)	
Ghmqc	Convert the parameter to a gradient HMQC experiment (M)	
gHMQC15	Set up parameters for <sup>15</sup> N gHMQC experiment (M)	
gHMQC d2	Set up parameters for <sup>15</sup> N gHMQC experiment using dec. 2 (M).	. 259
gHMQC d213	Set up parameters for <sup>13</sup> C gHMQC experiment using dec. 2 (M).	259
ghmqcps	Set up a PFG HMQC phase-sensitive pulse sequence (M)	
ghsqc	Set up a PFG HSQC pulse sequence (M)	
Ghsqc	Convert the parameter to a gradient HSQC experiment (M)	
gHSQC15	Set up parameters for <sup>15</sup> N gHSQC experiment (M)	
gHSQC d2	Set up parameters for <sup>15</sup> N gHSQC experiment using dec. 2 (M)	259
gHSQC d213	Set up parameters for <sup>13</sup> C gHSQC experiment using dec. 2 (M)	259
Ghsqctoxy	Convert parameters for gradient HSQCTOXY experiment (M)	
gilson	Open the Gilson Liquid Handler window (C)	
gin	Return current mouse position and button values (C)	
globalauto	Automation directory name (P)	
glue	Create a pseudo-2D dataset (M)	
gmapshim	Start gradient autoshimming (M)	
gmapshim au	Start acquisition with gradient shimming (M)	
gmapspin	Enable or disable spinning during gradient shimming (P)	
gmapsys	Run gradient autoshimming, set parameters, map shims (M)	
gmapz	Get parameters and files for gmapz pulse sequence (M)	
gmap findtof	Gradient shimming flag to first find tof (P)	
gmap_z1z4	Gradient shimming flag to first shim z1-z4 (P)	
gmax	Maximum gradient strength (P)	
gmqcosy	Set up PFG absolute-value MQF COSY parameter set (M)	
gnoesy	Set up a PFG NOESY parameter set (M)	
go	Submit experiment to acquisition (M)	
go	Pulse sequence setup macro called by go, ga, and au (M)	
gpat-gpat3	Gradient shape (P)	
gplan	Start interactive image planning (C)	
gradientdisable	Disable PFG gradients (P)	
gradientshaping	Activate shaping on the gradient pulses (P)	
gradientsnaping	Gradient step size (P)	
_	Gradients for X, Y, and Z axes (P)	
gradtype	Return the current graphics display status (C)	
graphis	Gray level window adjustment (P)	
grayctr	Gray 10 voi window adjustificit (1 )	. 200

amaua 1	Gray lavel alone (contract) adjustment (D)	260
graysl	Gray level slope (contrast) adjustment (P)	
grecovery grid	Draw a grid on a 2D display (M)	
groupcopy	Copy parameters of group from one tree to another (C)	
gspoil	Spoiler gradient level (P)	
gsspat	Slice-select gradient shape (P)	
gtnnoesy	Set up a PFG TNNOESY parameter set (M)	
gtnroesy	Set up a PFG absolute-value ROESY parameter set (M)	
gtotlimit	Gradient total limit (P)	
gtrim	Trim gradient level (P)	
gxmax,gymax,gzmax	Maximum gradient strength for each axis (P)	270
gzlvl	Pulsed field gradient strength (P)	
gzsize	Number of z-axis shims used by gradient shimming (P)	
gzwin	Spectral width percentage used for gradient shimming (P)	271
Н		
	Automated mustan acquisition (M)	274
h1	Automated proton acquisition (M)	
hlfreq	Process 1D proton spectro (M)	
h1p h2cal	Process 1D proton spectra (M)	
halt	Abort acquisition with no error (C)	
hc	Automated proton and carbon acquisition (M)	
hcapt	Automated proton, carbon, and APT acquisition (M)	
hcchtocsy	Set up parameters for HCCHTOCSY pulse sequence (M)	
hccorr	Automated proton, carbon, and HETCOR acquisition (M)	
hcdept	Automated proton, carbon, and DEPT acquisition (M)	
hcosy	Automated proton and COSY acquisition (M)	
hdmf	Modulation frequency for homonuclear decoupling (P)	
hcmult	Execute protocol actions of apptype hcmult (M)	
hdof	Frequency offset for homodecoupling (P)	
hdpwr	Power level for homodecoupling (P)	
hdpwrf	Homodecoupling fine power (optional) (P)	
hdres	Sets the tip angle resolution (P)	
hdseq	Waveform filename for band selective decoupling (P)	
hdwshim	Hardware shimming (P)	
hdwshimlist	List of shims for hardware shimming (P)	
het2dj	Set up parameters for HET2DJ pulse sequence (M)	
HETCOR	Change parameters for HETCOR experiment (M)	
hetcor	Set up parameters for HETCOR pulse sequence (M)	
hetcorcp1	Set up parameters for solids HETCOR pulse sequence (M)	
hetcorps	Set up parameters for HETCORPS pulse sequence (M)	
hetero2d	Execute protocol actions of apptype hetero2d (M)	
hidecommand	Execute macro instead of command with same name (C)	
hipwrampenable	High Power Amplifier Enable (P)	
Hmbc	Convert the parameter to a HMBC experiment (M)	283
Hmqc	Convert the parameter to a HMQC experiment (M)	
HMQC15	Set up parameters for <sup>15</sup> N HMQC experiment (M)	283
HMQC d2	Set up parameters for <sup>15</sup> N HMQC experiment using dec. 2 (M)	284
HMQC_d213	Set up parameters for <sup>13</sup> C HMQC experiment using dec. 2 (M)	284
hmqcr	Set up parameters for HMQCR pulse sequence (M)	284
Hmqctoxy	Convert the parameter to a HMQCTOXY experiment (M)	
HMQCTOXY15	Set up parameters for <sup>15</sup> N HMQCTOXY experiment (M)	284
HMQCTOXY_d2	Set up parameters for <sup>15</sup> N HMQCTOXY using decoupler 2 (M)	284
HMQCTOXY_d213	Set up parameters for <sup>13</sup> C HMQCTOXY using decoupler 2 (M).	
hmqctoxy3d	Set up parameters for HMQC-TOCSY 3D pulse sequence (M)	
ho	Horizontal offset (P)	
hom2dj	Set up parameters for HOM2DJ pulse sequence (M)	
homo	Homodecoupling control for the observe channel (P)	
HOMODEC	Change parameters for HOMODEC experiment (M)	
homo2d	Execute protocol actions of apptype homo2d (M)	
homorof1	Delay before turning on homo decoupling rf (P)	285

homorof2	Delay after blanking the amp and setting T/R switch to recv (P)	286
homorof3	Delay between setting T/R to receive and gating the recvr on (P)	
hoult	Set parameters alfa and rof2 according to Hoult (M)	
hpa	Plot parameters on special preprinted chart paper (C)	
Hprescan	Proton prescan (P))	. 287
hregions	Select integral regions in proton spectrum (M)	. 287
hs	Homospoil pulses (P)	
Hsqc	Convert the parameter to a HSQC experiment (M)	
HSQC15	Set up parameters for <sup>15</sup> N HSQC experiment (M)	. 288
HSQC_d2	Set up parameters for <sup>15</sup> N HSQC experiment using dec. 2 (M)	. 288
HSQC_d213	Set up parameters for <sup>13</sup> C HSQC experiment using dec. 2 (M)	
hsqcHT	Set up the hsqcHT experiment (M)	
Hsqctoxy	Convert parameters to a HSQCTOXY experiment (M)	
HSQCTOXY15	Set up parameters for <sup>15</sup> N HSQCTOXY experiment (M)	. 288
HSQCTOXY_d2	Set up parameters for <sup>15</sup> N HSQCTOXY using decoupler 2 (M)	. 288
HSQCTOXY_d213	Set up parameters for <sup>13</sup> C HSQCTOXY using decoupler 2 (M)	
hsqctoxySE	Set up parameters for HSQC-TOCSY 3D pulse sequence (M)	
hsrotor	Display rotor speed for solids operation (P)	
hst	Homospoil time (P)	
htbitrev	Hadamard bit reversal flag (P)	
htbw1	Hadamard pulse excitation bandwidth in ni (P)	
htcal1	RF calibration flag for Hadamard waveforms in ni (P)	
htfrq1	Hadamard frequency list in ni (P)	
htofs1	Hadamard offset in ni (P)	
htpwr1	Power level for RF calibration of Hadamard waveforms in ni (P)	
htss1	Stepsize for Hadamard waveforms in ni (P)	
hzmm	Scaling factor for plots (P)	
hztomm	Convert locations from Hz or ppm to plotter units (C)	. 291
1		
I		
i	Insert sample (M)	
ihwinfo	Hardware status of console (U)	
il	Interleave arrayed and 2D experiments (P)	
ilfid	Interleave FIDs during data processing (C)	
imagefile	Display an image file (M)	
imageprint	Plot non interactive gray scale image (M)	
imconi	Display 2D data in interactive grayscale mode (M)	
in	Lock and spin interlock (P)	
inadqt	Set up parameters for INADEQUATE pulse sequence (M)	
index2	Projection or 3D plane index selected (P)	
inept	Set up parameters for INEPT pulse sequence (M)	
initialize_iterate	Set iterate string to contain relevant parameters (M)	
input	Receive input from keyboard (C)	
ins	Integral normalization scale (P)	
ins2	2D volume value (P)	
insref	Fourier number scaled value of an integral (P)	
ins2ref	Fourier number scaled volume of a peak (P)	
insert	Insert sample (M)	
inset	Display an inset spectrum (C)	
integ	Find largest integral in a specified region (C)	
integrate	Automatically integrate 1D spectrum (M)	
intmod	Integral display mode (P)	
intvast	Produces a text file of integral regions (M)	
io	Integral offset (P)	
is	Integral scale (P)	
isadj	Automatic integral scale adjustment (M)	
isadj2	Automatic integral scale adjustment by powers of two (M)	
isreal		
	Utility macro to determine a parameter type (M)	
isstring	Utility macro to determine a parameter type (M)	. 301
isstring iterate	Utility macro to determine a parameter type (M)	. 301

J		
jcurwin	Work space numbers of all viewports (P)	303
jdesign	Start Plot Designer Program (M)	303
jexp	Join existing experiment (C)	
jexp1-jexp9999	Join existing experiment and display new parameters (M)	
jplot	Plot from Plot Designer program (C)	
jplotscale	Scale plot parameters (M)	
jplotunscale	Restore current experiment parameters (M)	
jprint	Prints the selected images to a printer or file (M)	
jumpret	Set up parameters for JUMPRET pulse sequence (M)	
jviewport	Work space numbers of the current viewports (P)	
jviewportlabel	Work space labels for all viewport buttons (P)	
jviewports	Viewport layout (P)  Activate and record activity in current window (M)	
jwin	Activate and record activity in current window (M)	300
K		
killft3d	Terminate any ft3d process started in an experiment (M,U)	
killplot	Stop plot jobs and remove from plot queue (M)	
killprint	Stop print jobs and remove from print queue (M)	
kind	Kinetics analysis, decreasing intensity (M)	
kinds	Kinetics analysis, decreasing intensity, short form (M)	
kini	Kinetics analysis, increasing intensity (M)	
kinis	Kinetics analysis, increasing intensity, short form (M)	309
L		
lastlk	Last lock solvent used (P)	312
lastmenu	Menu to display when Return button is selected (P)	
latch	Frequency synthesizer latching (P)	
1b	Line broadening in directly detected dimension (P)	
1b1	Line broadening in 1st indirectly detected dimension (P)	313
1b2	Line broadening in 2nd indirectly detected dimension (P)	314
lc1d	Pulse sequence for LC-NMR (M)	314
lcpar2d	Create 2D LC-NMR acquisition parameters (M)	
lcpeak	Peak number (P)	
lcplot	Plot LC-NMR data (M)	
lcpsgset	Set up parameters for various LC-NMR pulse sequences (M)	
lcset2d	General setup for 2D LC-NMR experiments (M)	
left	Set display limits to left half of screen (C)	
legrelay	Independent control of magnet leg relay (P)	
length	Determine length of a string (C)	
lf	List files in directory (C)	
liamp	Amplitudes of integral reset points (P)	
lifrq	Frequencies of integral reset points (P)	
liqbear	Liquids Bearing Air Level (P)	
listenoff	Disable receipt of messages from send2Vnmr (M)	
listenon	Enable receipt of messages from send2Vnmr (M)	
lkof	Track changes in lock frequency (P)	
112d	Automatic and interactive 2D peak picking (C)	
112dbackup	Copy current ll2d peak file to another file (M)	
112dmode	Control display of peaks picked by ll2d (P)	
llamp	List of line amplitudes (P)	
llfrq ln	List of line frequencies (P)Find natural logarithm of a number (C)	
load	Load status of displayed shims (P)	
loadcolors	Load colors for graphics window and plotters (M)	
loc	Location of sample in tray (P)	
locaction	Location of sample in tray (F)  Locator action (M)	
lock	Submit an Autolock experiment to acquisition (C)	
lockacqtc	Lock loop time constant during acquisition (P)	
lockfreq	Lock frequency (P)	
TOCKTIEd	Lock frequency (1 /	524

lockgain	Lock gain (P)	325
lockphase	Lock phase (P)	
lockpower	Lock power (P)	
locktc	Lock time constant (P)	
logate	Transmitter local oscillator gate (P)	
lookup	Look up words and lines from a text file (C)	
locprotoexec	Execute a protocol from the locator (M)	
lp	First-order phase in directly detected dimension (P)	
lp1	First-order phase in 1st indirectly detected dimension (P)	
1p2	First-order phase in 2nd indirectly detected dimension (P)	
lpalg	LP algorithm in np dimension (P)	
lpalg1	LP algorithm in ni dimension (P)	
lpalg2	LP algorithm in ni2 dimension (P)	
lpext	LP data extension in np dimension (P)	
lpext1	LP data extension in ni dimension (P)	
lpext2	LP data extension in ni2 dimension (P)	
lpfilt	LP coefficients to calculate in np dimension (P)	
lpfilt1	LP coefficients to calculate in ni dimension (P)	331
lpfilt2	LP coefficients to calculate in ni2 dimension (P)	
lpnupts	LP number of data points in np dimension (P)	331
lpnupts1	LP number of data points in ni dimension (P)	
lpnupts2	LP number of data points in ni2 dimension (P)	332
lpopt	LP algorithm data extension in np dimension (P)	332
lpopt1	LP algorithm data extension in ni dimension (P)	333
lpopt2	LP algorithm data extension in ni2 dimension (P)	
lpprint	LP print output for np dimension (P)	
lpprint1	LP print output for ni dimension (P)	
lpprint2	LP print output for ni2 dimension (P)	
lptrace	LP output spectrum in np dimension (P)	
lptrace1	LP output spectrum in ni dimension (P)	
lptrace2	LP output spectrum in ni2 dimension (P)	
ls	List files in directory (C)	
lsfid	Number of complex points to left-shift the np FID (P)	
lsfid1	Number of complex points to left-shift ni interferogram (P)	
lsfid2	Number of complex points to left-shift ni2 interferogram (P)	
lsfrq	Frequency shift of the fn spectrum (P)	
lsfrq1	Frequency shift of the fn1 spectrum (P)	
lsfrq2	Frequency shift of the fn2 spectrum (P)	
lvl	Zero-order baseline correction (P)	
lvltlt	Control sensitivity of lvl and tlt adjustments (P)	
	201421 501611 Ty 01 111 4110 410 440 440 410 410 410 410	
M		
macro	Macro name (P)	342
macrocat	Display a user macro file in text window (C)	
macrocp	Copy a user macro file (C)	
macrodir	List user macro files (C)	
macroedit	Edit a macro with user-selectable editor (M)	
macrold	Load a macro into memory (C)	
macrorm	Remove a user macro (C)	
macrosyscat	Display a system macro file in text window (C)	
macrosyscp	Copy a system macro to become a user macro (C)	
macrosysdir	List system macros (C)	
macrosysrm	Remove a system macro (C)	
macrovi	Edit a user macro with the vi text editor (M)	
make3dcoef	Make a 3D coefficients file from 2D coefficients (M)	
makedosyparams	Create parameters for DOSY processing (M)	
makefid	Make a FID element using numeric text input (C)	
makeeccglobals	Create global parameters for ECC control (M)	
makeslice	Synthesize 2D projection of 3D DOSY experiment (C)	
man	Display online description of command or macro (M)	
managedb	Update user files (U)	

manualpath	Path to user's manual directory (P)	
manvi	Edit online description of a command or macro (M)	
mapwin	List of experiment numbers (P)	
mark	Determine intensity of spectrum at a point (C)	
masvt	Type of variable temperature system (P)	. 351
maxattench1-4	Maximum limit for attenuator setting for rf channel 1-4 (P)	
maxpen	Maximum number of pens to use (P)	
md	Move display parameters between experiments (C)	
menu	Change status of menu system (C)	
menuvi	Edit a menu with vi text editor (M)	
method	Autoshim method (P)	
mf	Move FIDs between experiments (C)	
mfblk	Copy FID block (C)	
mfclose	Close memory map FID (C)	
mfdata	Move FID data (C)	
mfopen	Memory map open FID file (C)	
mftrace	Move FID trace (C)	
minsw	Reduce spectral width to minimum required (M)	
mkdir	Create new directory (C)	
mlabel	Menu label (P)	
move	Move to an absolute location to start a line (C)	
movedssw	Set downsampling parameters for selected spectral region (M)	
moveossw	Set oversampling parameters for selected spectral region (M)	
movesw	Move spectral window according to cursors (M)	
movetof	Move transmitter offset (M)	
mp	Move parameters between experiments (C)	
mqcosy	Set up parameters for MQCOSY pulse sequence (M)	
mref	Set referencing based on a existing spectrum of the sample (M).	
mrev8	Set up parameters for MREV8 pulse sequence (M)	
mrfb	Set the filter bandwidths for multiple receivers (P)	
mrgain	Set the gain for multiple receivers (P)	
mstat	Display memory usage statistics (C)	
mstring	Menu string (P)	
mtune	Tune probe using swept-tune graphical display (M)	
mv	Move and/or rename a file (C)	
mxconst	Maximum scaling constant (P)	. 363
A.I.		
N		
n1,n2,n3	Name storage for macros (P)	. 365
newmenu	Select a menu without immediate activation (C)	. 366
newshm	Interactively create a shim method with options (M)	
nextpl	Display the next 3D plane (M)	
nfni	Number of increments in 1st indirectly detected dimension (P)	. 367
ni2	Number of increments in 2nd indirectly detected dimension (P)	. 367
ni3	Number of increments in 3rd indirectly detected dimension (P)	
niter	Number of iterations (P)	
nimax	Maximum limit of ni (P)	
nl	Position cursor at the nearest line (C)	
nli	Find integral values (C)	
nlivast	Produces a text file of integral regions without a sum region (M)	. 369
nlivast2	Produces a text file with normalized integral regions (M)	
nlivast3	Produces a text file with normalized integral regions (M)	
nll	Find line frequencies and intensities (C)	
nm	Select normalized intensity mode (C)	
nm2d	Select Automatic 2D normalization (M)	
Noesy	Convert the parameter to a NOESY experiment (M)	
Noesy1d	Convert the parameter set to a Noesy1d experiment (M)	
noise	Measure noise level of FID (C)	
noisemult	Control noise multiplier for automatic 2D processing (M)	
noislm	Limit noise in spectrum (M)	
notebook	Notebook name (P)	
-	· · · · · · · · · · · · · · · · · · ·	

	Number of late maints (D)
np npoint	Number of data points (P)
nrecords	Determine number of lines in a file (M)
nt	Number of transients (P)
ntrig	Number of trigger signals to wait before acquisition (P)
ntype3d	Specify whether $f_1$ or $f_2$ display expected to be N-type (P)
nuctable	Display VNMR style nucleus table for a given H1 frequency (M) 374
numrcvrs	Number of receivers in the system (P)
numreq	Return the number of regions in a spectrum (C)
numrfch	Number of rf channels (P)
	1,0000000000000000000000000000000000000
0	
off	Make a parameter inactive (C)
on	Make a parameter active or test its state (C)
operator	Operator name (P)
operatorlogin	Sets workspace and parameters for the operator (M)
opx	Open shape definition file for Pbox (M)
oscoef	Digital filter coefficients for over sampling (P)
osfb	Digital filter bandwidth for oversampling (P)
osfilt	Oversampling filter for real-time DSP (P)
oslsfrq	Bandpass filter offset for oversampling (P)
overrange	Frequency synthesizer overrange (P)
oversamp	Oversampling factor for acquisition (P)
owner	Operating system account owner (P)
	operating system account owner (1)
P	
p1	Enter pulse width for p1 in degrees (C)
p1	First pulse width (P)
p1pat	Shape of excitation pulse (P)
p2pul	Set up sequence for PFG testing (M)
p31	Automated phosphorus acquisition (M)
p31p	Process 1D phosphorus spectra (M)
pa	Set phase angle mode in directly detected dimension (C)
pa1	Set phase angle mode in 1st indirectly detected dimension (C) 389
pacosy	Plot automatic COSY analysis (C)
pad	Preacquisition delay (P) 390
padept	Perform adept analysis and plot resulting spectra (C)
page	Submit plot and change plotter page (C)
page	Name of page (P)
panellevel	Display level for VnmrJ interface pages (P)
pap	Plot out "all" parameters (C)
par2d	Create 2D acquisition, processing, and display parameters (M) 393
par3d	Create 3D acquisition, processing, and display parameters (M) 393
par3rf	Get display templates for 3rd rf channel parameters (M)
par4d	Create 4D acquisition parameters (M)
paramedit	Edit a parameter and its attributes with user-selected editor (C) 394
paramvi	Edit a parameter and its attributes with vi editor (M)
pards	Create additional parameters used by downsampling (M)
parfidss	Create parameters for time-domain solvent subtraction (M) 395
parfix	Update parameter sets (M)
parlc	Create parameters for LC-NMR experiments (M)
parl12d	Create parameters for 2D peak picking (M)
parlp	Create parameters for linear prediction (M)
parmax	Parameter maximum values (P)
parmin	Parameter minimum values (P)
paros	Create additional parameters used by oversampling (M)
paros	Parameter step size values (P)
parversion	Version of parameter set (P)
path3d	Path to currently displayed 2D planes from a 3D data set (P) 399
pacis	Plot horizontal LC axis (M)
Pbox	Pulse shaping software (U) 400
1202	Table shaping softmare (0)

pbox bw	Define excitation band (M)	)1
pbox_bws	Define excitation band for solvent suppression (notch) pulses (M)40	
pbox_bwb pbox dmf	Extract dmf value from pbox.cal or Pbox shape file (M)	
pbox dres	Extract dres value from pbox.cal or Pbox shape file (M)	
pbox name	Extract name of last shape generated by Pbox from pbox.cal (M). 40	
pbox pw	Extract pulse length from pbox.cal or Pbox shape file (M)	
pbox pwr	Extract power level from Pbox.cal or Pbox shape file (M)	
pbox pwrf	Extract fine power level from pbox.cal or Pbox shape file (M) 40	
pboxget	Extract Pbox calibration data (M)	
pboxpar	Add parameter definition to the Pbox.inp file (M)	)4
pboxrst	Reset temporary Pbox variables (M)	)4
pboxunits	Converts to Pbox default units (M)	
pcon	Plot contours on a plotter (C)	
pcss	Calculate and show proton chemical shifts spectrum (M)	
peak	Find tallest peak in specified region (C)	
peak2d	Return information about maximum in 2D data (C)	
pen	Select a pen or color for drawing (C)40	)6
pexpl	Plot exponential or polynomial curves (C)	)7
pexpladd	Add another diffusion analysis to current plot (M)40	)7
pfgon	Pulsed field gradient amplifiers on/off control (P)40	)7
pfww	Plot FIDs in whitewash mode (C)	
pge	Convert parameter set to PGE pulse sequence (M)	
pge_calib	Calibrate gradient strengths for PGE pulse sequence (M)	
pge_data	Extract data from single element of PGE pulse sequence (M) 40	
pge_output	Output results from PGE pulse sequence (M)	
pge_process	Automated processing of data from PGE pulse sequence (M) 40	
pge_results	Calculate diffusion constant for integral region (M)40	
pge_setup	Set up gradient control parameters for PGE pulse sequence (M)41	
ph	Set phased mode in directly detected dimension (C)	
ph1	Set phased mode in 1st indirectly detected dimension (C)	
ph2	Set phased mode in 2nd indirectly detected dimension (C)	
phase	Change frequency-independent phase rp (M)	
phase	Phase selection (P)	
phase1	Phase of first pulse (P) 41	
phase2	Phase selection for 3D acquisition (P)	
phase3	Phase selection for 4D acquisition (P)	
phasing	Control update region during interactive phasing (P)	
phfid	Zero-order phasing constant for the np FID (P)	
phfid1	Zero-order phasing constant for ni interferogram (P)	
phfid2 Phosphorus	Zero-order phasing constant for ni2 interferogram (P)	.4 .4
	Set up pi/3 shifted sinebell-squared window function (M)	.4+ .4
pi3ssbsq pi4ssbsq	Set up pi/3 shifted sinebell-squared window function (M)	
pin	Pneumatics Router Interlock ((P)	
pintvast	Plots of integral regions (M)	
pir	Plot integral amplitudes below spectrum (C)	
pirn	Plot normalized integral amplitudes below spectrum (M)	
piv	Plot integral values below spectrum (M)	
pivn	Plot normalized integral values below spectrum (M)	6
pl	Plot spectra (C)	
pl2d	Plot 2D spectra in whitewash mode (C)	
plane	Currently displayed 3D plane type (P)	
plapt	Plot APT-type spectra automatically (M)	
plarray	Plotting macro for arrayed 1D spectra (M)	
plate glue	Define a glue order for plotting and display (U)41	
plc	Plot a carbon spectrum (M)	
plcosy	Plot COSY- and NOESY-type spectra automatically (M)	
pldept	Plot DEPT data, edited or unedited (M)	21
plfid	Plot FIDs (C)	
plfit	Plot deconvolution analysis (M)	
plgrid	Plot a grid on a 2D plot (M)	
plh	Plot proton spectrum (M)	22

plhet2dj	Plot heteronuclear J-resolved 2D spectra automatically (M)	
plhom2dj	Plot homonuclear J-resolved 2D spectra automatically (M)	423
plhxcor	Plot X,H-correlation 2D spectrum (M)	424
pll	Plot a line list (M)	425
p112d	Plot results of 2D peak picking (C)	425
plockport	Port number to use to lock out multiple ProTune processes (P)	425
plot	Automatically plot spectra (M)	
plot1d	Plotting macro for simple (non-arrayed) 1D spectra (M)	
plot2D	Plot 2D spectra (M)	
plotfile	Plot to a file (M)	
<del>-</del>		
plothiresprep	High resolution plot output preparation (M)	
plotmanual	Plot manually (M)	
plotlogo	Plots a logo (M)	
plotpreview	Creates temporary plots of the current plot output (M)	
plotside	Plot spectrum on side (M)	
plotter	Plotter device (P)	
plottop	Plot spectrum on top (M)	
plottopside	Plot spectrum on top and side (M)	429
plp	Plot phosphorus spectrum (M)	429
plplanes	Plot a series of 3D planes (M)	
plt2Darg	Plot 2D arguments (P)	
pltext	Plot text file (M)	
pltmod	Plotter display mode (P)	
plvast	Plot VAST data in a stacked 1D-NMR matrix format (M)	
plvast2d	Plot VAST data in a stacked pseudo-2D format (M)	
plww	Plot spectra in whitewash mode (C)	
pmode	Processing mode for 2D data (P)	
poly0	Display mean of the data in regression.inp file (M)	
PP	Decoupler pulse length (P)	
ppa	Plot a parameter list in plain English (M)	
ppcal	Proton decoupler pulse calibration (M)	
ppf	Plot peak frequencies over spectrum (C)	
pph	Print pulse header (M)	435
ppmm	Resolution on printers and plotters (P)	435
pprofile	Plot pulse excitation profile (M)	435
pps	Plot pulse sequence (C)	436
prealfa	Specify a delay for longer ring down (P)	
prep	Run prepare acquisition macro (M)	
Presat	Set up parameters for presat <sup>1</sup> H experiment (M)	
prevpl	Display the previous 3D plane (M)	
prescan	Study queue prescan (P)	
prescan CoilTable	Read or update the CoilTable File (M)	
prescan_colliable prescan tn	Return tn string for a given atomic number (M)	
<del>-</del>		
printer printfile	Printer device (P)	
-		
printformat	Format of saved-to-file image (P)	
printlayout	Layout of printed image (P)	
printoff	Stop sending text to printer and start print operation (C)	
printon	Direct text output to printer (C)	
printregion	Screen region to be printed (P)	
printsize	Size of printed image (P)	
printsend	Defines where image will print (P)	
probe	Probe type (P)	
probeConnect	Specify which nucleus can be acquired on each RF channel (P)	
Probe edit	Edit probe for specific nucleus (U)	
probe edit	Edit probe for specific nucleus (M)	
probe protection	Probe protection control (P)	
proc	Type of processing on np FID (P)	
proc1	Type of processing on ni interferogram (P)	441
proc1d	Processing macro for simple (non-arrayed) 1D spectra (M)	
proc2	Type of processing on ni2 interferogram (P)	
P1002	Process 2D spectra (M)	112

procarray	Process arrayed 1D spectra (M)	. 443
process	Generic automatic processing (M)	
procplot	Automatically process FIDs (M)	
profile	Set up pulse sequence for gradient calibration (M)	
proj	Project 2D data (C)	
Proton	Set up parameters for <sup>1</sup> H experiment (M)	
protune	Macro to start ProTune (M)	
protune	Shell script for start ProTune operation (U)	
protunegui	Macro to start ProTune in graphical user interface (M)	
prune	Prune extra parameters from current tree (C)	
pscale	Plot scale below spectrum or FID (C)	
pseudo	Set default parameters for pseudo-echo weighting (M)	
psg	Display pulse sequence generation errors (M)	
psggen	Compile a user PSG object library (M,U)	
psgset	Set up parameters for various pulse sequences (M)	
psgupdateon	Enable update of acquisition parameters (C)	
psgupdateoff	Prevent update of acquisition parameters (C)	
pshape	Plot pulse shape or modulation pattern (M)	
pshapef	Plot the last created pulse shape (M)	
pshaper	PostScript High Resolution plotting control (P)	
pslabel	Pulse sequence label (P)	
pslabel	PostScript Line Width control (P)	
-	Plot Arrayed Numbers (C)	
pssl ptext	Print out a text file (M)	
•		
ptspec3d	Region-selective 3D processing (P)	
ptsval	PTS frequency synthesizer value (P)	
pulseinfo	Shaped pulse information for calibration (M)	
pulsetool	RF pulse shape analysis (U)	
purge	Remove macro from memory (C)	
puttxt	Put text file into a data file (C)	
putwave	Write a wave into Pbox.inp file (M)	
pw	Enter pulse width pw in degrees (C)	
pw	Pulse width (P)	
pw90	90° pulse width (P)	
pwd	Display current working directory (C)	
pwpat	Shape of refocusing pulse (P)	
pwr	Set power mode in directly detected dimension (C)	
pwr1	Set power mode in 1st indirectly detected dimension (C)	
pwr2	Set power mode in 2nd indirectly detected dimension (C)	
pwsadj	Adjust pulse interval time (M)	
pwxcal	Decoupler pulse calibration (M)	
pxbss	Bloch-Siegert shift correction during Pbox pulse generation (P)	. 457
pxrep	Flag to set the level of Pbox reports (P)	. 457
pxset	Assign Pbox calibration data to experimental parameters (M)	
pxshape	Generates a single-band shape file (M)	. 458
Pxsim	Simulate Bloch profile for a shaped pulse (U)	. 458
Pxspy	Create shape definition using Fourier coefficients (U)	. 459
$\boldsymbol{Q}$		
	Longar doed time for longer ring down (D)	161
qcomp	Longer dead time for longer ring down (P)	
QKexp	Set up quick experiment (M)	
qtune	Tune probe using swept-tune graphical tool (C)	
?	Display the value of an individual parameter (C)	402
R		
/ \		
r	Recall display parameter set (M)	
r(n)	Recall some display parameters (C)	
r1-r7	Real-value storage for macros (P)	
ra	Resume acquisition stopped with sa command (C)	
rcvrwt	Weighting for different receivers (P)	
react	Recover from error conditions during werr processing (M)	466

		4
readallshims	Read all shims from hardware (M)	
readbrutape readfile	Read Bruker data files from 9-track tape (U)	
readlile	Read current values of acquisition hardware (C)	
readlk	Read current lock level (C)	
readparam	Read one of more parameters from a file (C)	
readultra	Read shim coil setting for Ultra•nmr shim system (M)	
real	Create a real variable without a value (C)	
recon all	Reconstruct images from 2D MRI fid data (C)	
record	Record keyboard entries as a macro (M)	
redor1	Set up parameters for REDOR1 pulse sequence (M)	
redosy	Restore 2D DOSY display from sub experiment (M)	. 474
reff1	Reference f1 Indirect Dimension from Observe Dimension (M)	. 474
reff2	Reference f2 Indirect Dimension from Observe Dimension (M)	
reffrq	Reference frequency of reference line (P)	
reffrq1	Reference freq. of reference line in 1st indirect dimension (P)	
reffrq2	Reference freq. of reference line in 2nd indirect dimension (P)	
refpos	Position of reference frequency (P)	
refpos1	Position of reference frequency in 1st indirect dimension (P)	
refpos2	Position of reference frequency in 2nd indirect dimension (P)	
refsource1	Center frequency in 1st indirect dimension (P)	
refsource2	Center frequency in 2nd indirect dimension (P)	.4//
region	Divide spectrum into regions (C)	.4/8
relayh	Set up parameters for RELAYH pulse sequence (M)	
rename	Move and/or rename a file (C)	
reqparcheck	Flag which enables/disables required parameters (P)	
reqparclear	Clears the parameters in required parameter list (M)	
reqparlist	List of required parameters (P)	
reqpartest resetf3	Tests whether required parameters are set (M)	
resetplotter	Reset plotter to system plotter (M)	
resolv	Set resolution enhancement parameters (M)	
restorenuctable	Calculate & store accurate nuctable for current system (M)	
resume	Resume paused acquisition queue (C)	482
return	Terminate execution of a macro (C)	482
rev	System software revision level (P)	
revdate	System software preparation date (P)	
rfband	RF band in use (P)	. 483
rfblk	Reverse FID block (C)	
rfchannel	Independent control of rf channel selection (P)	
rfchtype	Type of rf channel (P)	
rfdata	Reverse FID data (C)	
rfl	Reference peak position in directly detected dimension (P)	. 487
rfl1	Reference peak position in 1st indirectly detected dimension (P).	. 487
rfl2	Reference peak position in 2nd indirectly detected dimension (P)	487
rfp	Reference peak frequency in directly detected dimension (P)	
rfp1	Reference peak freq. in 1st indirectly detected dimension (P)	
rfp2	Reference peak freq. in 2nd indirectly detected dimension (P)	
rftrace	Reverse FID trace (C)	
rftype	Type of rf generation (P)	. 489
rfwg	RF waveform generator (P)	
right	Set display limits to right half of screen (C)	
rights	Determine an operator's specified right (C)	
rinput	Input data for a regression analysis (M)	
rl	Set reference line in directly detected dimension (M)	
rl1	Set reference line in 1st indirectly detected dimension (M)	
rl2	Set reference line in 2nd indirectly detected dimension (M)	
rm rmdir	Delete file (C)	
rmair rmsAddData	Remove directory (C)	
Roesy	Convert the parameter to a ROESY experiment (M)	
Roesyld	Convert the parameter set to a Roesy1d experiment (M)	
	contest are parameter set to a recopy to experiment (11)	/5

	Descrives acting time manading mules (D)	402
rof1 rof2	Receiver gating time preceding pulse (P)	. 493 403
rof3	Receiver gating time following T/R switch (P)	
rotate	Rotate 2D data (C)	
	Rotor synchronization (P)	
rotorsync	Zero-order phase in directly detected dimension (P)	
rp	Zero-order phase in 1st indirectly detected dimension (P)	
rp1 rp2	Zero-order phase in 2nd indirectly detected dimension (P)	
rt	Retrieve FIDs (M)	
rtcmx	Return Spinsight data into current experiment (C)	
rtp	Retrieve parameters (M)	
rts	Retrieve shim coil settings (C)	
rttmp	Retrieve experiment data from experiment subfile (M)	
rtv	Retrieve individual parameters (C)	
rtx	Retrieve parameters based on rtx rules (C)	
2 011	recire to parameters cased on tex rules (e)	. 170
S		
S	Save display parameters as a set (M)	503
s (n)	Save display parameters (C)	
s2pul	Set up parameters for standard two-pulse sequence (M)	
sa	Stop acquisition (C)	
sample	Submit change sample, Autoshim experiment to acquisition (M).	
samplename	Sample name (P)	
save	Save data (M)	
savefile	Base file name for saving files (P)	
saveglobal	Save selected parameters from global tree (P)	
sb	Sinebell constant in directly detected dimension (P)	
sb1	Sinebell constant in 1st indirectly detected dimension (P)	
sb2	Sinebell constant in 2nd indirectly detected dimension (P)	
sbs	Sinebell shift in directly detected dimension (P)	
sbs1	Sinebell shift in 1st indirectly detected dimension (P)	
sbs2	Sinebell shift in 2nd indirectly detected dimension (P)	
sc	Start of chart (P)	
sc2	Start of chart in second direction (P)	
scalelimits	Set limits for scales in regression (M)	
scalesw	Set scaling factor for multipulse experiments (M)	
scalesw	Scale spectral width in directly detected dimension (P)	
scalesw1	Set f <sub>1</sub> scaling factor for 2D multipulse experiments (M)	
scalesw1	Scale spectral width in 1st indirectly detected dimension (P)	. 509
scalesw2	Scale spectral width in 2nd indirectly detected dimension (P)	. 509
sd	Set first decoupler frequency to cursor position (M)	. 509
sd2	Set second decoupler frequency to cursor position (M)	. 509
sd3	Set third decoupler frequency to cursor position (M)	.510
sda	Set first decoupler frequency array (M)	. 510
sd2a	Set second decoupler frequency array (M)	
sd3a	Set third decoupler frequency array (M)	
sdp	Show diffusion projection (M)	
sel1d	Apptype macro for Selective 1D experiments (M)	
select	Select spectrum, FID, trace, or 2D plane without display (C)	
selex	Defines excitation band (M)	
selexcit	Set up PFG selective excitation pulse sequence (M)	. 512
selexHT	Set up a selective Hadamard experiment (M)	
send2vnmr	Send a command to VnmrJ (U)	
seqfil	Pulse sequence name (P)	
seqgen	Initiate compilation of user's pulse sequence (M,U)	
serverport	Returns the VnmrJ network listening port value (C)	
set2D	General setup for 2D experiments (M)	
set2d	General setup for 2D experiments (M)	
set3dproc	Set 3D processing (C)	
setallshims	Set all shims into hardware (M)	
setcolor	Set colors for graphics window and for plotters (C)	. 516

	Cat da aquellar maramatar valvas from mucha fila (M)	517
setdecpars	Set decoupler parameter values from probe file (M)	
setdec2pars	Set decoupler 2 parameter values from probe file (M)	
setdgroup	Set the Dgroup of a parameter in a tree (C)	
setenumeral	Set values of a string parameter in a tree (C)	
setether	Connect or reconnect host computer to Ethernet (U)	518
setexport	Set parameter bits for use with protocols (M)	518
setfrq	Set frequency of rf channels (C)	518
setgauss	Set a Gaussian fraction for lineshape (M)	
setgcal	Set the gradient calibration constant (M)	
setgcoil	Assign sysgcoil configuration parameter (M)	
_	Divide graphics window into rows and columns (C)	
setgrid		
setgroup	Set group of a parameter in a tree (C)	
sethtfrq1	Set a Hadamard frequency list from a line list ((M)	
sethw	Set values for hardware in acquisition system (C)	
setint	Set value of an integral (M)	
setlimit	Set limits of a parameter in a tree (C)	523
setlk	Set up lock parameters (M)	524
setlockfreq	Set lock frequency (M)	
setLP	Set up linear prediction in the direct dimension (M)	
setLP1	Set F1 linear prediction parameters (M)	
setlp0	Set parameters for zero linear phase (M)	
setnoether	Disconnect host computer from Ethernet (U)	
setoffset		
	Calculate offset frequency for given nucleus and ppm (M)	
setparams	Write parameter to current probe file (M)	
setpen	Set maximum number of HP plotter pens (M)	
setplotdev	Return characteristics of a named plotter (C)	
setpower	Set power and pulsewidth for a given γB1 value (M)	
setprotect	Set protection mode of a parameter (C)	
setrc	Set receiver constants (M)	
setref	Set frequency referencing (M)	529
setref1	Set freq. referencing for 1st indirectly detected dimension (M)	530
setref2	Set freq. referencing for 2nd indirect detected dimension (M)	531
setscout	Set up a scout run (M)	531
setssfilter	Set sslsfrq to the frequencies of each suppressed solvents (M)	532
setsw	Set spectral width (M)	
setsw1	Set spectral width in evolution dimension (M)	
setsw2	Set spectral width in 2nd evolution dimension (M)	
setselfrqc	Set selective frequency and width (M)	
setselinv	Set up selective inversion (M)	
settcldefault	Select default display templates for pulse sequence (M)	
settune	Opens the Auto Tune Setup dialog (M)	
settype	Change type of a parameter (C)	
setup	Set up parameters for basic experiments (M)	
setup_dosy	Set up gradient levels for DOSY experiments (M)	
setvalue	Set value of any parameter in a tree (C)	
setwave	Write a wave definition string into Pbox.inp file (M)	
setwin	Activate selected window (C)	536
sf	Start of FID (P)	536
sf1	Start of interferogram in 1st indirectly detected dimension (P)	536
sf2	Start of interferogram in 2nd indirectly detected dimension (P)	536
sfrq	Transmitter frequency of observe nucleus (P)	
sh2pul	Set up for a shaped observe excitation sequence (M)	
shdec	Set up for shaped observe excitation sequence (M)	
shell	Start a UNIX shell (C)	
shelli	Start an interactive UNIX shell (C)	
shim	Submit an Autoshim experiment to acquisition (C)	
shimset	Type of shim set (P)	
showconfig	Show system configuration settings (M)	
showconsole	Show system configuration settings (U)	
showfit	Display numerical results of deconvolution (M)	
showloginbox	Shows operator login dialog (M)	
shownumx	Show x position of number (P)	540

shownumy	Show y position of number (P)	
showoriginal	Restore first 2D spectrum in 3D DOSY experiment (M)	541
showplotter	Show list of currently defined plotters and printers (M)	541
showplotq	Display plot jobs in plot queue (M)	541
showprintq	Display print jobs in print queue (M)	
showprotunegui	show the graphical interface while tuning (P)	
showrfmon	Show RF Monitor Button in Hardware Bar (P)	
showstat	Display information about status of acquisition (M,U)	
sin	Find sine value of an angle (C)	
sine	Find values for a sine window function (M)	
sinebell	Select default parameters for sinebell weighting (M)	
sinesq	Find values for a sine-squared window function (M)	
size		
	Returns the number of elements in an arrayed parameter (O)	
slfreq	Measured line frequencies (P)	
slw	Spin simulation linewidth (P)	
smaxf	Maximum frequency of any transition (P)	
sminf	Minimum frequency of any transition (P)	
smsport	Sample Management System serial port connection (P)	
sn	Signal-to-noise ratio (P)	
solppm	Return ppm and peak width of solvent resonances (M)	
solvent	Lock solvent (P)	546
solvinfo	Retrieve information from solvent table (C)	546
sort	Sort real values of a parameter (M)	546
sp	Start of plot in directly detected dimension (P)	547
sp1	Start of plot in 1st indirectly detected dimension (P)	
sp2	Start of plot in 2nd indirectly detected dimension (P)	
spadd	Add current spectrum to add/subtract experiment (C)	
spcfrq	Display frequencies of rf channels (M)	
specdc3d	3D spectral dc correction (P)	
spin	Submit a spin setup experiment to acquisition (C)	
spin	Sample spin rate (P)	
spincad	Run SpinCAD program (C)	
spingen	Compile SpinCAD pulse sequence (M,U)	
	Set up a slfreq array (M)	
spinll	Open the Spinner Control window (C)	
spinner		
spins	Perform spin simulation calculation (C)	
split	Split difference between two cursors (M)	
spintype	Spinner Type ((P)	
spmax	Take the maximum of two spectra (C)	
spmin	Take minimum of two spectra in add/subtract experiment (C)	
spsm	Enter spin system (M)	
spsub	Subtract current spectrum from add/subtract experiment (C)	
sqcosine	Set up unshifted cosine-squared window function (M)	555
sqdir	Study queue directory (P)	
sqend	End a study queue (M)	556
sqexp	Load experiment from protocol (M)	556
sqfilemenu	Study queue file menu commands (M)	556
sqmode	Study queue mode (P)	556
sqname	Study queue parameter template (P)	
sqpars	Create study queue parameters for imaging (M)	
sqprotocol	Macro to create protocols (M)	
sgreset	Reset study queue parameters for imaging (M)	
sqrt	Return square root of a real number (O)	
sqsavestudy	Macro to save study parameters for imaging (M)	
sqsinebell	Set up unshifted sinebell-squared window function (M)	
sqsinebell	Spinning rate for magic angle spinning (P)	
srace	Read converted data into VnmrJ (C)	
	Calculate exact rof2 value for Cold Probes (M)	
srof2		
SS	Steady-state transients (P)	
ssecho	Set up solid-state echo pulse sequence (M)	
ssecho1	Set up parameters for SSECHO1 pulse sequence (M)	
ssfilter	Full bandwidth of digital filter to yield a filtered FID (P)	560

sslsfrq	Center of solvent-suppressed region of spectrum (P)	560
ssntaps	Number of coefficients in digital filter (P)	560
ssorder	Order of polynomial to fit digitally filtered FID (P)	561
stack	Stacking mode for processing and plotting arrayed spectra (M)	
stackmode	Stacking control for processing arrayed 1D spectra (P)	
startq	Start a chained study queue (M)	
status	Display status of sample changer (C,U)	
std1d	Apptype macro for Standard 1D experiments (M)	
stdshm	Interactively create a method string for autoshimming (M)	
sth	Minimum intensity threshold (P)	
string	Create a string variable (C)	
strtext	Starting point for LP data extension in np dimension (P)	
strtext1	Starting point for LP data extension in ni dimension (P)	
strtext2	Starting point for LP data extension in ni2 dimension (P)	
strtlp	Starting point for LP calculation in np dimension (P)	
strtlp1	Starting point for LP calculation in ni dimension (P)	
strtlp2	Starting point for LP calculation in ni2 dimension (P)	
studyid	Study identification (P)	
studypar	Study parameters (P)	
studystatus	Study status (P)	
studytime	Study time (P)	
su	Submit a setup experiment to acquisition (M)	
sub	Subtract current FID from add/subtract experiment (C)	
substr	Select a substring from a string (C)	
suselfrq	Select peak, continue selective excitation experiment (M)	
svdat	Save data (C)	
svf	Save FIDs in current experiment (M)	
svfdf	Save FID data in FDF format (M)	
svfdir	Directory for non-study data (P)	
Svfname	Create path for data storage (C)	
svfname	Filename parameter template for non-study data ((P)	
svp	Save parameters from current experiment (M)	
svs	Save shim coil settings (C)	
svs	Spin simulation vertical scale (P)	
svtmp	Move experiment data into experiment subfile (M)	
sw	Spectral width in directly detected dimension (P)	
sw1	Spectral width in 1st indirectly detected dimension (P)	
sw2	Spectral width in 2nd indirectly detected dimension (P)	
sw3	Spectral width in 3rd indirectly detected dimension (P)	
sysgcoil	System gradient coil (P)	
system	System type (P)	
systemdir	VnmrJ system directory (P)	5/6
T		
1		
t1	$T_1$ exponential analysis (M)	
t1s	$T_1$ exponential analysis with short output table (M)	
t2	$T_2$ exponential analysis (M)	
t2s	$T_2$ exponential analysis with short output table (M)	
tabc	Convert data in table order to linear order (M)	
tan	Find tangent value of an angle (C)	
tape	Read tapes from VXR-style system (M,U)	
tape	Control tape options of files program (P)	
tchan	RF channel number used for tuning (P)	
tcl	Send Tcl script to Tcl version of dg window (C)	
temp	Open the Temperature Control window (C)	
temp	Sample temperature (P)	
tempcal	Temperature calculation (C)	
tempcalc	Measure approximate sample temperature in Cold Probes (M)	
testacquire	Test acquire mode (P)	
testct	Check ct for resuming signal-to-noise testing (M)	
testsn	Test signal-to-noise of a spectrum (M)	584

toststa	Find which array matches a string M	501
teststr	Find which array matches a string M)	
textis	Return the current text display status (C)	
textvi	Edit text file of current experiment (M)	
th	Threshold (P)	
th2d	Threshold for integrating peaks in 2D spectra (P)	
thadj	Adjust threshold for peak printout (M)	
time	Display experiment time or recalculate number of transients (M)	
tin		
tlt	Temperature interlock (P)	
	Left-shift FID to time-domain cursor (M)	
tmove tmsref	Reference 1D proton or carbon spectrum to TMS (M)	
	Nucleus for observe transmitter (P)	
tn	Set up parameters for TNCOSYPS pulse sequence (M)	
tncosyps		
tndqcosy	Set up parameters for TNDQCOSY pulse sequence (M)	
tnmqcosy	Set up parameters for TNMQCOSY pulse sequence (M)	
tnnoesy	Set up parameters for TNNOESY pulse sequence (M)	
tnroesy	Set up parameters for TNROESY pulse sequence (M)	
tntocsy	Set up parameters for TNTOCSY pulse sequence (M)	
Tocsy	Convert the parameters to a TOCSY experiment (M)	
Tocsyld	Convert the parameter set to a Tocsy1d experiment (M)	
tocsyHT	Set up the tocsyHT experiment (M)	
tof	Frequency offset for observe transmitter (P)	
tpwr	Observe transmitter power level with linear amplifiers (P)	
tpwrf	Observe transmitter fine power (P)	
tpwrm	Observe transmitter linear modulator power (P)	
trace	Mode for <i>n</i> -dimensional data display (P)	
traymax	Sample changer tray slots (P)	
troesy	Set up parameters for TROESY pulse sequence (M)	
trunc	Truncate real numbers (O)	
tshift	Adjust tau2 to current cursor position (M)	
tugain	Receiver gain used in tuning (P)	
tune	Assign a frequency to a channel for probe tuning (C)	
tunehf	Tune both H1 and F19 on an HFX probe (M)	
tunematch	Default match target, in percent of optimum (P)	
tunemethod	Method to use for tuning (P)	
tuneResult	Message indicating how well the tuning succeeded (P)	595
tunesw	Width of the tuning sweep in Hz (P)	
tupwr	Transmitter power used in tuning (P)	595
typeof	Return identifier for argument type (O)	595
U		
ultra8	selects the Ultra 8 shim configuration (M)	597
ultra18	Select 18 shim configuration for Ultra 18 shim power supply (M	.) 597
undospins	Restore spin system as before last iterative run (M)	597
undosy	Restore original 1D NMR data from sub experiment (M)	598
unit	Define conversion units (C)	598
unlock	Remove inactive lock and join experiment (C)	599
updatepars	Update all parameter sets saved in a directory (M)	599
updateprobe	Update probe file (M)	600
updaterev	Update after installing new VnmrJ version (M)	
updtgcoil	Update gradient coil (M)	600
updtparam	Update specified acquisition parameters (C)	
usemark	Use "mark" output as deconvolution starting point (M)	
userdir	VnmrJ user directory (P)	
usergo	Experiment setup macro called by go, ga, and au (M)	
userfixpar	Macro called by fixpar (M)	
17		
V		
vast1d	Set up initial parameters for VAST experiments (M)	
vastget	Selects and displays VAST spectra (M)	604

vastglue	Assemble 1D datasets into a 2D (or pseudo-2D) datasets (M)	
vastglue2	Assemble 1D datasets into a 2D (or pseudo-2D) datasets (M)	
vastgo	Turn off LC stop flow automation, start VAST automation (M)	
vbg	Run VNMR processing in background (U)	
vf	Vertical scale of FID (P)	
vi	Edit text file with vi text editor (M)	
vibradd	Display relative amplitudes of Cold Probe vibrations (M)	
vjhelp	Display VnmrJ help (U)	
vn	Start VNMR directly (U)	
vnmr	Starts VnmrJ (U)	
vnmr2sc	VNMR to SpinCAD pulse sequence translator (M)	
vnmr_accounting	Open Accounting window (U)	
vnmrjcmd()	Commands to invoke the GUI popup (C)	
vnmrexit	Exit from the VNMR system (C)	
vnmrj	Start VnmrJ (U)	
vnmrplot	Plot files (U) Print text files (U)	
vnmrprint	Vertical offset (P)	
VO	Vertical position of spectrum (P)	
vp vpaction	Set initial state for multiple viewports (M)	
vpaction	Current vertical position of FID (P)	
vpfi	Current vertical position of imaginary FID (P)	
vpset3def	Set the viewport state to three default viewports (M)	
vpsetup	Set new viewports (M)	
vpsecup	Vertical scale (P)	
vs2d	Vertical scale (1)  Vertical scale for 2D displays (P)	
vsadj	Automatic vertical scale adjustment (M)	
vsadj2	Automatic vertical scale adjustment by powers of 2 (M)	
vsadjc	Automatic vertical scale adjustment for 13C spectra (M)	
vsadjh	Automatic vertical scale adjustment for <sup>1</sup> H spectra (M)	
vsproj	Vertical scale for projections and traces (P)	
vtairflow	Variable Temperature Air Flow (P)	
vtairlimits	Variable Temperature Air Flow Limits (P)	
vtc	Variable temperature cutoff point (P)	
vtcomplvl	Variable temperature compensation for gradient shimming (P)	
vttype	Variable temperature controller present (P)	
vtwait	Variable temperature wait time (P)	
vxr_unix	Convert VXR-style text files to UNIX format (M, U)	
W		
w	Who is using system (C)	620
walkup	Walkup automation (M)	
waltz	WALTZ decoupling present (P)	
wbs	Specify action when bs transients accumulate (C)	621
wbs	When block size (P)	621
wc	Width of chart (P)	621
wc2	Width of chart in second direction (P)	621
wcmax	Maximum width of chart (P)	
wc2max	Maximum width of chart in second direction (P)	622
wdone	Specify action when experiment is done (C)	
wdone	Specify action when experiment is done (P)	
werr	Specify action when error occurs (C)	
werr	When error (P)	
wet	Flag to turn on or off wet solvent suppression ((P)	
Wet1d	Set up parameters for wet <sup>1</sup> H experiment (M)	
wetdqcosy	Set up parameters for a WETDQCOSY pulse sequence (M)	
wetgcosy	Set up parameters for a WETGCOSY pulse sequence (M)	
wetghmqcps	Set up parameters for a WETGHMQCPS pulse sequence (M)	
wetghsqc	Set up parameters for a WETGHSQC pulse sequence (M)	
wetgmqcosy	Set up parameters for a WETGHSQC pulse sequence (M)	
wetit	Set up and create pulse shapes for Wet1d experiment (M)	624

wetnoesy	Set up parameters for a WETNOESY pulse sequence (M)	625
wethoesy	Number of peaks for wet solvent suppression (P)	
wetpwxcal	Set up parameters for a WETPWXCAL pulse sequence (M)	
wettntocsy	Set up parameters for a WETTWACAL pulse sequence (M)	
wetshape	Shape for pwwet pulses (P)	
wecsnape	Specify action when experiment completes (C)	
wexp	When experiment completes (P)	
wf	Width of FID (P)	
wf1	Width of interferogram in 1st indirectly detected dimension (P)	
wf2	Width of interferogram in 2nd indirectly detected dimension (P)	
wfgtest	Waveform generator test (M)	
wigtest	Weight and Fourier transform 1D data (C)	
wft1d	Weight and Fourier transform f <sub>2</sub> for 2D data (C)	
wft1da	Weight and Fourier transform phase-sensitive data (M)	
wftldac	Combine arrayed 2D FID matrices (M)	
wft2d	Weight and Fourier transform 2D data (C)	
wft2d wft2da	Weight and Fourier transform phase-sensitive data (M)	
wft2dac	Combine arrayed 2D FID matrices (M)	
wftt3	Process f <sub>3</sub> dimension during 3D acquisition (M)	
which		
wnt	Display which command or macro is used (M)	
	When number of transients (P)	
wnt		
wp	Width of plot in directly detected dimension (P)	
wp1	Width of plot in 1st indirectly detected dimension (P)	
wp2	With formattal tout to a desire (C)	
write	Write formatted text to a device (C)	
writefid	Write numeric text file using a FID element (C)	
writeparam	Write one of more parameters to a file (C)	
writespectrum	write a spectrum to a binary file (C)	
wrtp	Command string executed after rtp command (P)	
wsram	Send hardware configuration to acquisition console (C)	
wshim	Conditions when shimming is performed (P)	
wtfile	User-defined weighting in directly detected dimension (P)	
wtfile1	User-defined weighting in 1st indirectly detected dimension (P)	
wtfile2	User-defined weighting in 2nd indirectly detected dimension (P).	
wtgen	Compile user-written weighting functions (M,U)	
wti	Interactive weighting (C)	637
wtia	Interactive weighting for 2D absorptive data (M)	
wtune	Specify when to tune (P)	
wtunedone	What to do after ProTune tuning is done (P)	
wysiwyg	Set plot display or full display (P)	639
V		
X		
<b>x</b> 0	X-zero position of HP pen plotter or Postscript device (P)	642
x1	X1 shim gradient (P)	
x2y2	X2Y2 shim gradient (P)	
x3	X3 shim gradient (P)	
<b>x</b> 4	X4 shim gradient (P)	
xdiag	Threshold for excluding diagonal peaks when peak picking (P)	
xgate	Load time counter (M)	
xm1	Utility macro for study queue experiment manager (M)	
xmaction	Perform study queue action (M)	
xmactionw	Perform study queue action for walkup (M)	
xmaddreg	Add a required protocol before the main protocol (M)	
xmcheckreq	Check required protocol name (M)	
xmconvert	Convert a temporarily stored study into a submitted study (M)	
xmcopy	Copy protocols in a study queue (M)	
xmdelete	Delete nodes in a study queue (M)	644
xmenablepanel	Enable or disable a parameter panel (M)	
xmendg	End a chained study queue (M)	
xmgetatts	Get study queue attributes (M)	
	2 1	-

xmHprescan	Set up and process Proton prescans (M)	6/15
xminit	Initialize an imaging study queue (M)	
xmlockup	Move a study queue node up and lock it (M)	
xmmakenode	Make a new study queue node (M)	
xmmext	Find next prescan or next experiment in study queue (M)	
	Run prescans in study queue (M)	
xmprescan		
xmreact	Recover from error conditions during automation study (M)	
xmreadnode	Read attributes from a study queue node (M)	
xmrtpar	Retrieve parameters from a study queue node (M)	
xmsample	Write enterQ entry for a sample for study queue – liquids (M)	
xmsara	Write enterQ entry for a sample for study queue – imaging (M)	
xmsatfrq	Processing for Presat experiment (M)	
xmselect	Action when study queue node is selected (M)	
xmsetatts	Set an attribute for a study queue node (M)	
xmsetattr	Set an attribute for a study queue node (M)	
xmshowdata	Show data from a study queue node (M)	
xmstartnightq	Start the night queue (M)	
xmsubmit	Submit sample(s) to the study queue (M)	648
xmtime	Update the study queue time (M)	648
xmtune	Check tune parameter during automation (M)	
xmwerr	Recover from acquisition error in study queue (M)	
xmwexp	Processing macro for end of acquisition in study queue (M)	
xmwritenode	Write study queue node attributes (M)	
xmwritesq	Write study queue node order (M)	
xpol	Cross-polarization (P)	
xpolar1	Set up parameters for XPOLAR1 pulse sequence (M)	
хy	XY shim gradient (P)	
xz	XZ shim gradient (P)	
xz2	XZ2 shim gradient (P)	650
<b>Y</b> y0	Y-zero position of HP pen plotter or Postscript device (P)	651
y1	Y1 shim gradient (P)	
y 3	Y3 shim gradient (P)	
- Table	Y4 shim gradient (P)	
y4	YZ shim gradient (P)	
yz yz2	YZ2 shim gradient (P)	
y 2 2	1 ZZ Simii gradient (1 )	052
Z		
Z	Add integral reset point at cursor position (C)	653
<b>z</b> 0	Z0 field position (P)	
z1	Z1 shim gradient (P)	
zlc	Z1C shim gradient (P)	
z2	Z2 shim gradient (P)	
z2c	Z2C shim gradient (P)	
z2x2y2	Z2X2Y2 shim gradient (P)	
z2x3	Z2X3 shim gradient (P)	
z2xy	Z2XY shim gradient (P)	
z2y3	Z2Y3 shim gradient (P)	
z3	Z3 shim gradient (P)	
z3c	Z3C shim gradient (P)	655
z3x	Z3X shim gradient (P)	655
z3x2y2	Z3X2Y2 shim gradient (P)	
z3x3	Z3X3 shim gradient (P)	
z3xy	Z3XY shim gradient (P)	
z3y	Z3Y shim gradient (P)	
z3y3	Z3Y3 shim gradient (P)	
z4	Z4 shim gradient (P)	
z4c		
	Z4C shim gradient (P)	050
z4x	Z4C shim gradient (P)Z4X shim gradient (P)	
z4x z4x2y2		656

z4xy	Z4XY shim gradient (P)	657
z4y	Z4Y shim gradient (P)	
<b>z</b> 5	Z5 shim gradient (P)	657
z5x	Z5X shim gradient (P)	657
z5y	Z5Y shim gradient (P)	
<b>z</b> 6	Z6 shim gradient (P)	
<b>z</b> 7	Z7 shim gradient (P)	
<b>z</b> 8	Z8 shim gradient (P)	
zeroneg	Set all negative intensities of 2D spectra to zero (C)	658
zoom	Adjust display to given width (M)	658
zx2y2	ZX2Y2 shim gradient (P)	
zx3	ZX3 shim gradient (P)	
zxy	ZXY shim gradient (P)	
zy3	ZY3 shim gradient (P)	
Index		661

## **Notational Conventions**

The *VnmrJ Command and Parameter Reference* describes in detail the commands, macros, and parameters in VnmrJ software. Information new to VnmrJ in this version is shown by a change bar (as shown to the left of this paragraph).

#### Title Line Codes

Each entry has a letter in parentheses in the title line that identifies the type of entry:

(C)	VnmrJ command
(M)	VnmrJ macro command (from the maclib directory)
(O)	MAGICAL programming operator
(P)	VnmrJ parameter
(U)	UNIX command (not executable within VnmrJ)
(C,U)(M,U)	Executable from UNIX or VnmrJ (note that syntax is different)

### Applicability

An entry with applicability information applies only to the system or accessory listed. If the entry does not include applicability information, the entry applies to all systems.

#### Command and Macro Syntax

Each command and macro entry includes the syntax used when entering it into the system. The following examples illustrate this syntax:

halt	If no parentheses are shown, enter the command or macro exactly as shown, e.g., enter halt.
<pre>delexp(exp_num)</pre>	If parentheses are shown, enter the command or macro name as shown, but replace arguments with a value, e.g., if exp_num is 5, enter delexp (5).
rttmp(file)	Arguments can be a string (e.g., name of file or solvent), number, variable, or parameter (e.g., pw),. If a string, enclose it with single quote marks, e.g., if file is samp02, enter rttmp('samp02').  If number, variable, or parameter, do not use marks.
rl<(frequency)>	Angle brackets (< and >) indicate optional input, e.g., if frequency not needed or the default value of frequency is acceptable, enter rl, but if frequency has a value such as 10, enter rl (10).

## **Notational Conventions**

<pre>md(<from_exp,>to_exp)</from_exp,></pre>	Arguments can also be optional. Use a comma to separate arguments, e.g., md (2,3). Unless stated otherwise, the order of arguments is often important.
nll<('pos')>	A keyword is frequently used as an argument. In the syntax, keywords are shown in single quotes and are entered exactly as shown, e.g., to use the optional keyword 'pos' for nll, enter nll('pos').
dc2d('f1' 'f2')	A vertical bar indicates an OR condition, e.g., either 'f1' or 'f2' can be an argument to dc2d.
<pre>sin(angle)&lt;:n&gt;</pre>	Some commands return values to a calling macro. This is shown by a colon followed by one or more variables, e.g., if angle is variable $x$ and $n$ is variable $rt$ , then $sin(x)$ : $rt$ returns the value of $sin(x)$ to the calling macro via the variable $rt$ .
z(reset1,reset2,)	Three dots indicate the sequence of arguments continues. Unless a limit is given, you can enter one argument, two, three, or as many as needed.

# Parameter Syntax

Parameter syntax is always in the form parameter\_name=value. If value is a string, enclose it in single quote marks; otherwise, no marks are used, e.g., auto='y', plotter='ThinkJet', spin=5. Note that some parameters are not user-enterable.

## Notational Conventions

Throughout all Varian, Inc. NMR manuals, typewriter-like characters identify commands, parameters, directories, file names, and text displayed on the screen.

Because pressing the Return key is required at the end of almost every command or line of text you type on the keyboard, assume this use of the Return key unless stated otherwise.

## Other Sources of Information

For further information about an entry, refer to the manual listed under "See also." For general coverage on VnmrJ, refer to the following manuals (each manual is also online):

VnmrJ Walkup NMR Spectroscopy User Guide VnmrJ Installation and Administration VnmrJ Imaging NMR

Autocalibration macros (M)

Abort acquisition with error (C) aa Terminate action of calling macro and all higher macros (C) abort

abortallacqs Reset acquisition computer in a drastic situation (C) abortoff Terminate normal functioning of abort in a macro (C) aborton Restore normal functioning of abort in a macro (C)

abs Find absolute value of a number (C)

**ACbackup** Make backup copy of current probe file (M) Writes records for operator login and logoff (M) acct Print copy of probe file after autocalibration (M)

Find arc cosine of number (C) acos

AC1S-AC11S

ACreport

Automatic analysis of COSY data (C) acosy

Automatic analysis of COSY data, old algorithm (C) acosyold acqdisp Display message on the acquisition status line (C) acqi Interactive acquisition display process (C)

Open Acqmeter window (M) acqmeter Acqmeter Open Acqmeter window (U)

acqmode Acquisition mode (P)

Open Acquisition Status window (M) acqstat Open Acquisition Status window (U) Acqstat

acqstatus Acquisition status (P) acquire Acquire data (M)

actionid Current study queue node id (P)

activestudy Active study name (P)

add Add current FID to add/subtract experiment (C)

addi Start interactive add/subtract mode (C) addnucleus Add new nucleus to existing probe file (M)

addpar Add selected parameters to current experiment (M)

addparams Add parameter to current probe file (M) addprobe Create new probe directory and probe file (M) adept Automatic DEPT analysis and spectrum editing (C)

aexppl Automatic plot of spectral expansion (M) ai Select absolute-intensity mode (C) Absolute-intensity group (P) aig alfa Set alfa delay before acquisition (P)

alock Automatic lock control (P)

ampmode Independent control of amplifier mode (P)

amptype Amplifier type (P)

analyz Calculate standard peak height (M) analyze Generalized curve fitting (C) Print out "all" parameters (C) ap "All" parameters display control (P) ap Plot parameters automatically (M) apa

## A

aph Automatic phase adjustment of spectra (C)
aph0 Automatic phase of zero-order term (C)
aphb Auto phasing for Bruker data (C)

aphx Perform optimized automatic phasing (M)
appdirs Starts Applications Directory Editor (M)

appmode Application mode (P)
apptype Application type (P)

Apt Set up parameters for APT experiment (M)

aptaph Automatic processing for APT spectra (M)

array Easy entry of linearly spaced array values (M)

array Parameter order and precedence (P)
arraydim Dimension of experiment (P)
asin Find arc sine of number (C)

at Acquisition time (P)

atan Find arc tangent of a number (C)
atan2 Find arc tangent of two numbers (C)
atcmd Call a macro at a specified time (M)

Append string to current experiment text file (M)

attval Calculate pulse width (M)
atune ProTune Present (P)

au Submit experiment to acquisition and process data (M)

AuCALch3i Set up autocalibration with CH3I sample (M)

AuCALch3i1 Get autocalibration with CH3I sample (M)

AuCALch3oh Set up autocalibration with Autotest sample (M)

AuCALch3oh1 Get autocalibration with Autotest sample (M)

Aucalibz0 Automatic Hz to DAC calibration for Z0 (M)

AuCdec Carbon decoupler calibration macro (M)

AuCgrad Carbon/proton gradient ratio calibration macro (M)

AuCobs Carbon observe calibration macro (M)

audiofilterAudio filter board type (P)Aufindz0Automatic adjustment of Z0 (M)AugcalProbe gcal calibration macro (M)AugmapAutomated gradient map generation (M)

Augmapz 0 Automatic lock gradient map generation and z0 calibration (M)

AuHobs Proton decoupler calibration (M)

AuHobs Proton observe calibration macro (M)

Aumakegmap Auto lock gradient map generation (M)

AuNuc Get parameters for a given nucleus (M)

auto Prepare for an automation run (C)

auto Automation mode active (P)

auto\_au Controlling macro for automation (M)

Autobackup Back up current probe file (M)

autodept Automated complete analysis of DEPT data (M)

autodir Automation directory absolute path (P)

autogo Start automation run (C)

autolist Set up and start chained acquisition (M)

autoname Create path for data storage (C)

autoname Prefix for automation data file (P)

autora Resume suspended automation run (C)

autosa Suspend current automation run (C)

autoscale Resume autoscaling after limits set by scalelimits macro (M)
autostack Automatic stacking for processing and plotting arrays (M)

autotest Open Auto Test Window (C)

autotime Displays approximate time for automation (M)

av Set abs. value mode in directly detected dimension (C)
av1 Set abs. value mode in 1st indirectly detected dimension (C)
av2 Set abs. value mode in 2nd indirectly detected dimension (C)
averag Calculate average and standard deviation of input (C)

awc Additive weighting const. in directly detected dimension (P)

awc1 Additive weighting const. in 1st indirectly detected dimension (P)

awc2 Additive weighting const. in 2nd indirectly detected dimension (P)

axis Provide axis labels and scaling factors (C)

axis Axis label for displays and plots (P)

axisf Axis label for FID displays and plots (P)

## aa Abort acquisition with error (C)

Syntax: aa

Description:

Aborts an experiment that has been submitted to acquisition. If the experiment is active, it is aborted immediately, all data is discarded, and the experiment is interpreted as an error. Any data collected from an earlier block size transfer is retained. If any werr processing is defined, that processing occurs, followed by any queued experiments. The login name, and the FID directory path in file are used as keys to find the proper experiment to abort.

In some circumstances, there is a delay between the time **go** is entered and the acquisition is started. During this time, instructions based on the selected pulse sequence are being generated. This is signified by the letters "PSG" appearing in the upper left corner of the status window. An aa command issued under these circumstances reports that no acquisition is active but it instead stops the instruction generation process and the message "PSG aborted" appears.

See also: NMR Spectroscopy User Guide

Related: file File name of a parameter set (P)

go Submit experiment to acquisition (C)
halt Abort acquisition with no error (C)
werr Specify action when error occurs (C)

werr When error (P)

#### abort Terminate action of calling macro and all higher macros (C)

Syntax: abort

Description: Terminates the action of the calling macro and all higher levels of nested

macros. abort is used only in macros and not entered from the keyboard. It generates an error condition, which is the reason why the calling macro and any

parent (nested) macros above will also be aborted. To exit from the execution of

a macro without generating an error, use return.

See also: VnmrJ User Programming

Related: abortoff Terminate normal functioning of abort in a macro (C)

aborton Restore normal functioning of abort in a macro (C)

return Terminate execution of a macro (C)

abortallacqs Reset acquisition computer in a drastic situation (C)

Syntax: abortallacqs

Description: Reboots the acquisition system from the host computer. Wait at least 30 seconds

before attempting new acquisitions.

See also: NMR Spectroscopy User Guide

abortoff Terminate normal functioning of abort in a macro (C)

Syntax: abortoff

Description: Changes the action of an abort command in a macro. Normally, abort (or

any command aborting with an error condition) terminates the action of the calling macro and all higher levels of nested macros; however if the abortoff command is executed prior to a macro containing the abort command, only the macro containing abort terminates and execution continues to the next macro. The operation of the abortoff command is nullified by the aborton

command. abortoff is used only in macros and not entered from the

keyboard.

See also: VnmrJ User Programming

Related: abort Terminate action of calling macro and all higher macros (C)

 ${\tt aborton} \qquad \qquad {\tt Restore \ normal \ functioning \ of \ abort \ in \ a \ macro \ (C)}$ 

aborton Restore normal functioning of abort in a macro (C)

Syntax: aborton

Description: Nullifies the operation of a abortoff command and restores the normal

functioning of the <a href="mailto:aborton">aborton</a> is used only in macros and not

entered from the keyboard.

See also: VnmrJ User Programming

Related: abortoff Terminate normal functioning of abort in a macro (C)

abs Find absolute value of a number (C)

Syntax: abs(number)<:value>

Description: Finds the absolute value of a number. Absolute value is a nonnegative number

equal in numerical value to the given number (e.g., abs (-6.5) is 6.5).

Arguments: number is the given real number.

value is the return value with the absolute value of the given number. The

default is to display the value in the status window.

Examples: abs(-25)

abs(n):abs val

See also: VnmrJ User Programming

AC1S-AC11S Autocalibration macros (M)

Syntax: ACnS, where n is a number from 1 to 11.

Description: Performs automatic system calibration. When finished with the calibration

routines, the current probe file is updated. If the probe is new to the system (i.e., all values in the probe file are zero), system power levels are determined followed by calibration. If power levels are listed in the current probe file, these values are used. The macro AC1S determines <sup>1</sup>H pw90, AC5S begins <sup>13</sup>C calibration, including decoupler power calibrations. AC10S performs <sup>19</sup>F

calibration, and AC11S performs <sup>31</sup>P calibration.

See also: NMR Spectroscopy User Guide

ACbackup Make backup copy of current probe file (M)

Syntax: ACbackup

Description: Called by the autocalibration macros AC1S-AC11S to back up the probe file

after calibration ends. This macro is not usually called by the user.

See also: NMR Spectroscopy User Guide

Related: AC1S-AC11S Autocalibration macros (M)

acct Writes records for operator login and logoff (M)

Applicability: VnmrJ

Syntax: acct('start'|'done')

Description: acct writes operator login and logoff records to the system adm/tmp/

macrorecords.txt file used by the accounting package.

See also: VnmrJ Installation and Administration manual

Related: operator operator name (P)

operatorlogin Sets work space and parameters for the operator (M)

vnmr accounting Open Accounting window (U)

ACreport Print copy of probe file after autocalibration (M)

Syntax: ACreport

Description: Called by the autocalibration macros AC1S-AC11S to print a copy of the probe

file before beginning a new autocalibration run.

See also: NMR Spectroscopy User Guide

Related: AC1S-AC11S Autocalibration macros (M)

acos Find arc cosine of number (C)

Syntax: acos(value)<:n>

Description: Finds the arc cosine (also called the inverse cosine) of a number.

Arguments: value is a number in the range of  $\pm -1.0$  to +1.0.

n is a return argument giving the arc cosine, in radians, of value. The default

is to display the arc cosine value in the status window.

Examples: acos(.5)

acos(value):acos\_val

See also: VnmrJ User Programming

Related: sin Find sine value of an angle (C)

Δ

acosy Automatic analysis of COSY data (C)

Syntax: acosy

Description: Automatically analyzes a 2D COSY data set with fn=fn1 and sw=sw1. In this

algorithm, a fuzzy pattern recognition technique is used to detect peaks and cluster the cross peaks into groups. Symmetry measures and chemical shifts for all cross peaks are calculated. Connectivities and the correlation table are displayed on the computer screen. This method is less sensitive to the threshold

and rejects most artifacts in the peak list.

See also: NMR Spectroscopy User Guide

Related: acosyold Automatic analysis of COSY data (C)

fn Fourier number in 1st indirectly detected dimension (P)
fn1 Fourier number in directly detected dimension (P)
112d Automatic and interactive 2D peak picking (C)
sw Spectral width in directly detected dimension (P)
sw1 Spectral width in 1st indirectly detected dimension (P)

acosyold Automatic analysis of COSY data, old algorithm (C)

Syntax: acosyold

Description: Analyzes COSY data using an old algorithm.

See also: NMR Spectroscopy User Guide

Related: acosy Automatic analysis of COSY data (C)

fn Fourier number in 1st indirectly detected dimension (P)
fn1 Fourier number in directly detected dimension (P)
112d Automatic and interactive 2D peak picking (C)
sw Spectral width in directly detected dimension (P)
sw1 Spectral width in 1st indirectly detected dimension (P)

acqdisp Display message on the acquisition status line (C)

Syntax: acqdisp(message)

Description: Displays the message specified on the acquisition status line. acqdisp is used

primarily by the acquisition process to update the screen.

Arguments: message is a text string, up to 8 characters long.

See also: NMR Spectroscopy User Guide

acgi Interactive acquisition display process (C)

Syntax: acqi<('par'|'disconnect'|'exit'|'standby')><:\$ret>

Description: Opens the Acquisition window for interactive locking and shimming on the lock

signal, FID, or spectrum. When using a spectrometer, acqi normally

automatically starts. On all systems, if the console has been recently rebooted,

enter su before running acqi.

If acqi is connected to the console and you start an acquisition (su/go/au),

acqi automatically disconnects.

The pulse sequence and parameter set for the FID/spectrum display can be selected by entering gf. Note that if clicking the FID button in acqi causes acqi to "disconnect," the common cause is that gf had not been executed.

The FID display is controlled by the parameters <code>lsfid</code>, <code>phfid</code>, and <code>dmgf</code>. These display parameters are automatically sent to <code>acqi</code> when <code>acqi</code> is first invoked. These parameters may subsequently be changed and sent again to

acqi with the command acqi ('par'). If phfid is not set to "Not Used" for the FID display in acqi, a slide control will be available in acqi for the interactive adjustment of the phfid parameter. The slide will be in the IPA set of adjustments. If the parameter dmgf exists and is set to 'av', the FID display in acqi displays the square root of the sum of the squares of the real and imaginary channels.

The spectrum display is controlled by parameters sp, wp, dmg, rp, lp, rfl, rfp, vs, vp, sw, and fn. These parameters are automatically sent to acqi when acqi is first invoked. These parameters can subsequently be changed and sent again to acqi with the command acqi ('par'). The preparation macro gf also calls acqi ('par'), thereby causing these parameters to be sent to acqi. If fn is greater than 64K, it is lowered to 64K.

A convenient method of setting these parameters is to acquire a spectrum with go, then ft and adjust the display with the ds command options. Once the display is set the way you want, enter gf. The same display should then appear when the spectrum display is selected from acqi. Note that weighting parameters are not used in the *acqi* spectrum display.

The manual NMR Spectroscopy User Guide has a step-by-step description of using acqi.

## Arguments:

'par' causes the current values of parameters lsfid, phfid, dmgf, sp, wp, dmg, rp, lp, rfl, rfp, vs, sw, and fn to be sent to acqi.

'disconnect' causes acqi to be disconnected. Clicking the Close button in acqi is equivalent, and puts acqi in the standby mode. Lock parameters, the spin parameter, and the shim values are sent back to the current experiment when acqi is "disconnected." If the experiment has the load parameter set to 'y', then the shim values are not delivered to the experiment.

'exit' causes an exit from acqi. Clicking the exit button in the Acquisition window is equivalent.

\$ret is a return value with the success or failure of running acqi. The default
is a warning displayed in the status window if acqi fails.

'standby' starts acqi and puts it into the standby mode.

#### Examples: acgi

acqi('par')
acqi('disconnect')
acqi('exit')
acqi:\$ok

See also: NMR Spectroscopy User Guide

Acqstat

## Related:

Display mode in directly detected dimension (P) dmg Absolute-value display of FID data or spectrum in acgi (P) dmqf ds Display a spectrum (C) Fourier number in directly detected dimension (P) Fourier transform 1D data (C) ft gf Prepare parameters for FID/spectrum display in acqi (M) Submit an experiment to acquisition (C) qo load Load status of displayed shims (P) lkof Track changes in lock frequency (P) 1p First-order phase in directly detected dimension (P) lsfid Number of complex points to left-shift the np FID (P) phfid Zero-order phasing constant for np FID (P) Ref. peak position in 1st indirectly detected dimension (P) rfl rfp Ref. peak frequency in directly detected dimension (P) Zero-order phase in directly detected dimension (P) rp

Start of plot in directly detected dimension (P)

Bring up the acquisition status display (U)

spin Sample spin rate (P)

Spectral width in directly detected dimension (P)

Vertical position of the spectrum (P)

Vertical scale (P)

wp Width of plot in directly detected dimension (P)

## acqmeter Open Acqmeter window (M)

Syntax: acqmeter<(remote system)>

Description: Opens the Acqueter window and shows a time line of lock level, temperature

(VT), and/or spinner speed. When first opened, only lock level is displayed. By clicking anywhere in the lock level window with the right mouse button, a menu pops up with choices to close the lock level window, show a temperature (VT) window, show a spinner window, open a properties window, or close the Acqmeter window. Click on the choice desired in the menu with either the left or right mouse button. In the properties window, the host, font, color, and graphical mode can be changed. Continue to click in any Acqmeter window with the right mouse button to open the menu and then open or close windows,

or close the Acqueter window, as desired.

Arguments: remote system is the host name of a remote machine on the same network.

The default is the local machine. To activate the remote feature, the local and remote machines must be on the same Ethernet LAN (local area network) and the local machine must be able to get the Internet address of the remote machine

(usually in the /etc/hosts file).

Examples: acqmeter

acqmeter('nmr500')

See also: NMR Spectroscopy User Guide

Related: acqi Interactive acquisition display (C)

Acqueter Window (U)

## Acqmeter Open Acqmeter window (U)

Syntax: Acqmeter <remote system> <-f file> <&>

Description: Opens the Acqueter window and shows a time line of lock level, temperature

(VT), and/or spinner speed. When first opened, only lock level is displayed. By clicking anywhere in the lock level window with the right mouse button, a menu pops up with choices to close the lock level window, show a temperature (VT) window, show a spinner window, open a properties window, or close the Acqmeter window. Click on the choice desired in the menu with either the left or right mouse button. In the properties window, the host, font, color, and graphical mode can be changed. Continue to click in any Acqmeter window with the right mouse button to open the menu and then open or close windows,

or close the Acqmeter window, as desired.

Arguments: remote system is the host name of a remote machine on the same network.

The default is the local machine. To activate the remote feature, the local and remote machines must be on the same Ethernet LAN (local area network) and the local machine must be able to get the Internet address of the remote machine

(usually in the /etc/hosts file).

-f file is the name of a template file in the directory \$vnmruser/vnmrsys/templates/acqstat used to set the attributes of the Acqmeter window when it opens. This allows customizing the Acqmeter window for different users and experiments. The default name of the file is default.

& (ampersand) character added to the command makes Acqmeter into a background process. For example, if "lab" is the remote machine host name, entering the command Acqmeter lab & displays the acquisition status of the "lab" remote machine as a background process. To activate the remote feature, the local and remote machines must be on the same Ethernet LAN (local area network) and the local machine must be able to get the Internet address of the remote machine (usually in the /etc/hosts file).

Examples: Acqmeter &

Acqmeter nmr400 &

Acqmeter gem300 -f inova500.lisa &

See also: NMR Spectroscopy User Guide

Related: acqi Interactive acquisition display (C)

acqmeter Open Acqmeter window (M)

## acqmode Acquisition mode (P)

Description: A global parameter specifying the normal acquisition mode for acquiring,

locking, fid shimming, and prescan in VnmrJ.

Values: '' (empty string) normal acquisition

'lock' lock acquisition

'fidscan' fid shimming acquisition

'prescan' prescan acquisition

See also: VnmrJ Imaging, User Guide, NMR Spectroscopy User Guide

## acqstat Open Acquisition Status window (M)

Syntax: acqstat<(remote system)>

Description: Opens the Acquisition Status window, which displays acquisition information

such as the current acquisition task, experiment number, spinner status, and temperature status. When the host computer is attached to a spectrometer, this window should open automatically when VnmrJ is started. In the properties window, the host, font, color, and graphical mode can be changed. For a complete description of these windows, refer to the manual *NMR Spectroscopy* 

User Guide.

Arguments: remote system is the host name of a remote machine on the same network.

The default is the local machine. To activate the remote feature, the local and remote machines must be on the same Ethernet LAN (local area network) and the local machine must be able to get the Internet address of the remote machine

(usually in the /etc/hosts file).

Examples: acgstat

acqstat('u500')

See also: NMR Spectroscopy User Guide

Related: Acqstat Open the Acquisition Status window (U)

showstat Display information about status of acquisition (C,U)

### Acgstat Open Acquisition Status window (U)

Syntax: Acgstat <remote system> <-f file> <&>

Description: Opens the Acquisition Status window, which displays acquisition information

such as the current acquisition task, experiment number, spinner status, and temperature status. When the host computer is attached to a spectrometer, this

window should open automatically when VnmrJ is started. In the properties window, the host, font, color, and graphical mode can be changed. For a complete description of these windows, refer to the manual *NMR Spectroscopy User Guide*.

## Arguments:

remote\_system is the host name of a remote machine on the same network. The default is the local machine. To activate the remote feature, the local and remote machines must be on the same Ethernet LAN (local area network) and the local machine must be able to get the Internet address of the remote machine (usually in the /etc/hosts file).

-f file is the name of a template file in the directory \$vnmruser/vnmrsys/templates/acqstat used to set the attributes of the Acquisition Status window when it opens. This allows customizing the Acquisition Status window for different users and experiments. The default name of the file is default.

& (ampersand) character added to the command makes Acqstat into a background process. For example, if "lab" is the remote machine host name, entering the command Acqstat lab & displays the acquisition status of the "lab" remote machine as a background process. To activate the remote feature, the local and remote machines must be on the same Ethernet LAN (local area network) and the local machine must be able to get the Internet address of the remote machine (usually in the /etc/hosts file).

Examples: Acqstat &

Acqstat nmr400 &

Acqstat gem300 -f inova500.lisa &

See also: NMR Spectroscopy User Guide

Related: Acqstat Open the Acquisition Status window (U)

showstat Display information about status of acquisition (C,U)

## acqstatus Acquisition status (P)

## Description:

Whenever wbs, wnt, wexp, or werr processing occurs, the acquisition condition that initiated that processing is available from the parameter acqstatus. This acquisition condition is represented by two numbers, a "done" code and an "error" code. The done code is set in acqstatus[1] and the error code is set in acqstatus[2]. Macros can take different actions depending on the acquisition condition.

The done codes and error codes are listed below and in the file acq\_errors in /vnmr/manual. For example, a werr macro could specify special processing if the maximum number of transients is accumulated. The appropriate test in the macro would be:

```
if (acqstatus[2] = 200) then "do special processing, e.g. dp='y' au" endif
```

#### Done codes:

- 11. FID complete
- 12. Block size complete (error code indicates bs number completed)
- 13. Soft error
- 14. Warning
- 15. Hard error
- 16. Experiment aborted
- 17. Setup completed (error code indicates type of setup completed)
- 101. Experiment complete
- 102. Experiment started

Error codes:

#### Warnings

- 101. Low-noise signal
- 102. High-noise signal
- 103. ADC overflow occurred
- 104. Receiver overflow occurred\*

#### Soft errors

- 200. Maximum transient completed for single-precision data
- 201. Lost lock during experiment (LOCKLOST)
- 300. Spinner errors:
- 301. Sample fails to spin after three attempts at repositioning
- 302. Spinner did not regulate in the allowed time period (RSPINFAIL)\*
- 303. Spinner went out of regulation during the experiment (SPINOUT)\*
- 395. Unknown spinner device specified (SPINUNKNOWN)\*
- 396. Spinner device is not powered up (SPINNOPOWER)\*
- 397. RS-232 cable not connected from console to spinner (SPINRS232)\*
- 398. Spinner does not acknowledge commands (SPINTIMEOUT)\*
- 400. VT (variable temperature) errors:
- 400. VT did not regulate in the given time vttime after being set
- 401. VT went out of regulation during the experiment (VTOUT)
- 402. VT in manual mode after automatic command (see Oxford manual)\*
- 403. VT safety sensor has reached limit (see Oxford manual)\*
- 404. VT cannot turn on cooling gas (see Oxford manual)\*
- 405. VT main sensor on bottom limit (see Oxford manual)\*
- 406. VT main sensor on top limit (see Oxford manual)\*
- 407. VT sc/ss error (see Oxford manual)\*
- 408. VT oc/ss error (see Oxford manual)\*
- 495. Unknown VT device specified (VTUNKNOWN)\*
- 496. VT device not powered up (VTNOPOWER)\*
- 497. RS-232 cable not connected between console and VT (VTRS232)\*
- 498. VT does not acknowledge commands (VTTIMEOUT)
- 500. Sample changer errors:
- 501. Sample changer has no sample to retrieve
- 502. Sample changer arm unable to move up during retrieve
- 503. Sample changer arm unable to move down during retrieve
- 504. Sample changer arm unable to move sideways during retrieve
- 505. Invalid sample number during retrieve
- 506. Invalid temperature during retrieve
- 507. Gripper abort during retrieve
- 508. Sample out of range during automatic retrieve
- 509. Illegal command character during retrieve\*
- 510. Robot arm failed to find home position during retrieve\*
- 511. Sample tray size is not consistent\*
- 512. Sample changer power failure during retrieve\*
- 513. Illegal sample changer command during retrieve\*
- 514. Gripper failed to open during retrieve\*
- 515. Air supply to sample changer failed during retrieve\*
- 525. Tried to insert invalid sample number\*
- 526. Invalid temperature during sample changer insert\*
- 527. Gripper abort during insert\*
- 528. Sample out of range during automatic insert
- 529. Illegal command character during insert\*
- 530. Robot arm failed to find home position during insert\*
- 531. Sample tray size is not consistent\*
- 532. Sample changer power failure during insert\*
- 533. Illegal sample changer command during insert\*
- 534. Gripper failed to open during insert\*
- 535. Air supply to sample changer failed during insert\*
- 593. Failed to remove sample from magnet\*
- 594. Sample failed to spin after automatic insert

595. Sample failed to insert properly

596. Sample changer not turned on

597. Sample changer not connected to RS-232 interface

598. Sample changer not responding\*

600. Shimming errors:

601. Shimming user aborted\*

602. Lost lock while shimming\*

604. Lock saturation while shimming\*

608. A shim coil DAC limit hit while shimming\*

700. Autolock errors:

701. User aborted (ALKABORT)\*

702. Autolock failure in finding resonance of sample (ALKRESFAIL)

703. Autolock failure in lock power adjustment (ALKPOWERFAIL)\*

704. Autolock failure in lock phase adjustment (ALKPHASFAIL)\*

705. Autolock failure, lock lost in final gain adjustment (ALKGAINFAIL)\*

800. Autogain errors.

801. Autogain failure, gain driven to 0, reduce pw (AGAINFAIL)

Hard errors

901. Incorrect PSG version for acquisition

902. Sum-to-memory error, number of points acquired not equal to np

903. FIFO underflow error (a delay too small?)\*

904. Requested number of data points (np) too large for acquisition\*

905. Acquisition bus trap (experiment may be lost)\*

1000. SCSI errors:

1001. Recoverable SCSI read transfer from console\*

1002. Recoverable SCSI write transfer from console\*\*

1003. Unrecoverable SCSI read transfer error\*

1004. Unrecoverable SCSI write transfer error\*

1100. Host disk errors:

1101. Error opening disk file (most likely a UNIX permission problem)\*

1102. Error on closing disk file\*

1103. Error on reading from disk file\*

1104. Error on writing to disk file\*

See also: NMR Spectroscopy User Guide

Related: react Recover from error conditions during werr processing (M)

werr Specify action when error occurs (C)

werr When error (P)

acquire Acquire data (M)

Description: Macro to acquire data. It uses execpars to select the prep and prescan

method, executes them, and then begins acquisition.

See also: NMR Spectroscopy User Guide

Related: execpars Set up the exec parameters (M)

execprescan Execute prescan macro (P)

xmnext Find next prescan or next experiment in study queue (M)
xmwexp Processing macro for end of acquisition in study queue (M)

actionid Current study queue node id (P)

Applicability: Liquids, Imaging

Description: Specifies the currently selected study queue node id.

See also: VnmrJ Imaging, User Guide, NMR Spectroscopy User Guide

Related: xmaction Perform study queue action (M)

xmnext Find next prescan or next experiment in study queue (M)

xmselect Action when study queue node is selected (M)

## activestudy Active study name (P)

Applicability: Liquids, Imaging

Description: A global parameter that specifies the currently active study name. In the Walkup

interface, it specifies the currently active automation run.

Values: 's 20050601' active study name

'auto 2005.06.01' active automation run name

'null' no active study or automation run

See also: VnmrJ Imaging, User Guide and NMR Spectroscopy User Guide

Related: acquire Acquire data (M)

autodir Automation directory absolute pathname (P)

cqinit Initialize liquids study queue (M)

studyid Study identification (P)

xmaction Perform study queue action (M)

xmselect Action when study queue node is selected (M)

## add Add current FID to add/subtract experiment (C)

Syntax: (1) add< (multiplier<, 'new'>) >

(2) add('new')

(3) add('trace',index)

Description: Adds the last displayed or selected FID to the current contents of the add/

subtract experiment (exp5). The parameters <code>lsfid</code> and <code>phfid</code> can be used to shift or phase rotate the selected FID before it is combined with the data in the add/subtract experiment. A multi-FID add/subtract experiment can be created by using the <code>'new'</code> keyword. Individual FIDs in a multi-FID add/subtract experiment can subsequently be added to using the <code>'trace'</code> keyword

followed by the index number of the FID.

Arguments: multiplier is a value that the FID is to be multiplied by before being added

to the add/subtract experiment (exp5). The default is 1.0.

'new' is a keyword to create a new FID element in a add/subtract experiment.

'trace' is a keyword to use the next argument (index) as the number of the FID to add to in an add/subtract experiment. The default is to add to the first FID

in a multi-FID add/subtract experiment.

index is the index number of the FID to be used as a target in a multi-FID add/

subtract experiment.

Examples: add

add(0.75)
add('new')
add('trace',2)

See also: NMR Spectroscopy User Guide

Related: clradd Clear add/subtract experiment (C)

1sfid Number of complex points to left-shift ni interferogram (P)

phfid Zero-order phasing constant for np FID (P)
select Select a spectrum without displaying it (C)

spadd Add current spectrum to add/subtract experiment (C)
sub Subtract current FID from add/subtract experiment (C)

## addi Start interactive add/subtract mode (C)

Syntax: addi

Description:

Starts the interactive add/subtract mode. Before entering addi, start the process with clradd and spadd, then display a second spectrum on the screen. This may involve changing experiments, selecting a second member of an array of spectra, a different trace of a 2D spectrum, or displaying a spin simulated spectrum. The Fourier numbers (fn) must be the same in the two spectra to be manipulated. The width (sw) of the two spectra need not be identical, although adding spectra of different widths will probably not be meaningful. Having selected the second spectrum and ensuring it is in nm mode, enter addi to begin the interactive process.

After addi is invoked, spectrum 1, the spectrum selected by the spadd command, appears in the center of the display. Spectrum 2, the spectrum that was active when addi was entered, appears on the bottom. The sum or difference of these spectra appears on top of the screen. When addi is first entered, this spectrum will be the sum (1+2) by default. The spectra is manipulated using the mouse.

The select button toggles between different modes of control.

- When the label at the screen bottom reads "active: current", all of the parameters (except wp) control spectrum 2, and spectrum 2 can be phased, scaled, or shifted relative to spectrum 1.
- After clicking on select, the label at the screen bottom reads "active: addsub", and now all of the parameters except wp control spectrum 1.
- Clicking select again toggles the label to read "active: result", and now parameter changes affect only the sum or difference spectrum.

Note that wp always controls all spectra, because differential expansions of the two spectra are not supported. Note also that the colors of the labels change to match the colors of the different spectra.

The sum/difference spectrum displayed on the screen while addi is active is strictly a temporary display. Once all manipulations have been performed, and assuming the sum/difference is something you wish to perform further operations with (such as plotting), it must be saved into the add/subtract experiment (exp5) by clicking on save. At this point, spectrum 1, which was in the add/subtract experiment, is overwritten by the sum or difference spectrum, and addi ceases operation. In most cases, you will next want to enter jexp5 ds to display the difference spectrum on the screen, ready for further manipulation (expansion, line listing, etc.) and plotting. If you wish to continue with the add/subtract process by adding in a third spectrum, display that spectrum in the usual way and enter addi again.

See also: NMR Spectroscopy User Guide

Related: clradd Clear add/subtract experiment (C)

jexp Join existing experiment (C)

nm Select normalized intensity mode (C)

spadd Add current spectrum to add/subtract experiment (C)
spmin Take minimum of two spectra in add/subtract experiment (C)
spsub Subtract current spectrum from add/subtract experiment (C)

wp Width of plot in directly detected dimension (P)

## addnucleus Add new nucleus to existing probe file (M)

Applicability: ALL

Description: Entries for nuclei not in the default probe file are appended to the end of the file.

The argument should correspond to a nucleus in the nuctable.

Syntax: addnucleus('nucleus')

Arguments: nucleus — name followed by atomic number, e.g. C13 not 13C.

Examples: addnucleus('Si29')

See also: NMR Spectroscopy User Guide

Related: addprobe Create new probe directory and probe file (M)

deletenucleusRemoves nucleus entry to probe file (M)getparamReceive parameter from probe file (M)

probe Probe type (P)

setparams Write parameter to current probe file (M)

## addpar Add selected parameters to current experiment (M)

Applicability: The '3d', '3rf', '4d', 'fid', and 'image' arguments work on all

systems but are only useful if system has the proper hardware.

Description: Creates selected parameters in the current experiment.

Arguments: If no argument is entered, addpar displays instructions for its use.

'2d', '3d', '3rf', '4d', 'downsamp', 'fid', 'image', 'll2d', 'lp', 'oversamp', and 'ss' are keywords (only one keyword is used at a time) specifying the parameters to be created:

- '2d' specifies creating ni, phase, and sw1, which can be used to acquire a 2D data set (functions the same as macro par2d).
- '3d' specifies creating d3, ni2, phase2, and sw2, which can used to acquire a 3D data set (functions the same as macro par3d).
- '3rf' specifies retrieving the ap and dg2 display templates for third rf channel and 3D parameters (functions the same as macro par3rf).
- '4d' specifies creating the acquisition parameters d4, ni3, phase3, and sw3, which can be used to acquire a 4D data set (functions the same as macro par4d).
- 'downsamp' specifies creating the parameters downsamp, dscoef, dslsfrq, dsfb, and filtfile for digital filtering and downsampling (functions the same as macro pards).
- 'fid' specifies creating FID display parameters axisf, crf, deltaf, dotflag, vpf, and vpfi if the parameter set is older and lacks these parameters (functions the same as macro fidpar).
- 'll2d' specifies creating th2d and xdiag for the ll2d 2D peak picking program (functions the same as macro parll2d).
- 'lp' specifies creating lpalg, lpopt, lpfilt, lpnupts, strtlp, lpext, strtext, lptrace, and lpprint for linear prediction in the acquisition dimension (functions the same as macro parlp). The display template for the dglp macro is also created if necessary.
- 'oversamp' specifies creating parameters def\_osfilt, filtfile, oscoef, osfb, osfilt, oslsfrq, and oversamp for oversampling and digital filtering (functions the same as macro paros).

• 'ss' specifies adding parameters ssorder, ssfilter, ssntaps, and sslsfrq for time-domain solvent subtraction (functions the same as macro parfidss).

dim specifies the dimension when adding linear prediction parameters: 1 for the first implicit dimension or 2 for the second implicit dimension. Default is the acquisition dimension. Therefore, addpar('lp') creates the parameters listed above; addpar('lp',1) creates lpalg1, lpopt1, lpfilt1, lpnupts1, strtlp1, lpext1, strtext1, lptrace1, and lpprint1; and addpar('lp',2) creates lpalg2, lpopt2, lpfilt2, lpnupts2, strtlp2, lpext2, strtext2, lptrace2, and lpprint2. Each separate dimension of a multidimensional data set can have its own unique parameters.

Examples: addpar

addpar('3d')
addpar('lp',1)

See also: NMR Spectroscopy User Guide; VnmrJ Imaging NMR

Related: def osfilt Default value of osfilt (P)

fidpar Add parameters for FID display in current experiment (M)

osfilt Oversampling filter for real-time DSP (P)
par2d Create 2D acquisition parameters (M)
par3d Create 3D acquisition parameters (M)

par3rf Get display templates for 3rd rf channel parameters (M)

par4d Create 4D acquisition parameters (M)

pards Create digital filtering and downsampling parameters (M)
parfidss Set up parameters for time-domain solvent subtraction (M)
paros Create oversampling and digital filtering parameters (M)

parll2d Create parameters for 2D peak picking (M)
parlp Create parameters for linear prediction (M)

## addparams Add parameter to current probe file (M)

Syntax: addparams(param, value, nucleus<, 'tmplt'><, 'system'>)

Description: Adds a new parameter and its value for a specified nucleus to the probe file or

to the probe template.

Arguments: param is the name of the parameter to be added.

value is a string with the value to be written for the parameter.

nucleus is the nucleus to add in the probe file.

'tmplt' is a keyword to add the parameter to the local template. The default

is the probe file.

'system' is a keyword to add the parameter to the system-level template or probe file, provided that you have write permission to that file. The default is to

add the parameter to the local template or probe file.

Examples: addparams('ref pwr', '53', tn)

addparams('ref pwx','00',dn,'tmplt')

addparams('ref\_pwx2','00',dn2,'tmplt','system')

See also: NMR Spectroscopy User Guide

Related: getparam Receive parameter from probe file (M)

setparams Write parameter to current probe file (M)

updateprobe Update probe file (M)

## addprobe Create new probe directory and probe file (M)

Syntax: addprobe(probe\_name<, 'stdar'|'system'><, 'stdpar'>)

Description: Creates a new probe directory and a probe file. Default nuclei included in this

file are <sup>1</sup>H, <sup>19</sup>F, <sup>13</sup>C, and <sup>15</sup>N. The information is saved in the user's directory

vnmrsys/probes.

Arguments: probe name is the name to be given to the probe directory and probe file.

'stdpar' and 'system' are keywords for the second and third arguments:

- If the second argument is 'stdpar', calibration values from the standard parameter sets (stdpar/H1.par, stdpar/C13.par, etc.) will be read and written into the probe file.
- If the second argument is 'system' and the user has write permission into the VnmrJ system probes directory (typically /vnmr/probes), then a system-level probe directory will be made.
- If the second argument is 'system' and the third argument is 'stdpar', then both actions in the preceding bullets will occur.
- The default is the probe file is created with all parameters initialized to zero.

Examples: addprobe('idpfg')

addprobe('idpfg','stdpar')

addprobe('idpfg','system','stdpar')

See also: NMR Spectroscopy User Guide; VnmrJ Walkup

Related: addnucleus Add new nucleus to existing probe file (M)

deletenucleus Removes nucleus entry to probe file (M)
getparam Receive parameter from probe file (M)

probe Probe type (P)

setparams Write parameter to current probe file (M)

## adept Automatic DEPT analysis and spectrum editing (C)

Syntax: adept<(<'noll'><,'coef'><,'theory'>)>

Description: Automatically analyzes a set of four DEPT spectra and edits the spectra so that the spectra is arrayed as follows:

• #4 is CH<sub>3</sub> carbons only

• #3 is CH<sub>2</sub> carbons only

• #2 is CH carbons only

• #1 is all protonated carbons

Because adept modifies the transformed data, it should not be repeated without retransforming the data between calls. adept produces a text file dept. out in the current experiment directory, which contains the result of the

analysis.

Arguments: The following keyword arguments can be supplied in any order:

'noll' causes the line listing to be skipped. If 'noll' is not supplied as an argument, adept first performs a line listing. In that case, the threshold parameter th must be set properly before starting adept.

'coef' causes the combination coefficients to be printed.

'theory' causes theoretical coefficients to be used. The default is optimized coefficients.

## A

Examples: adept

adept('coef')

adept('theory','noll')

See also: NMR Spectroscopy User Guide

Related: autodept Automated complete analysis of DEPT data (M)

Dept Set up parameters for DEPT experiment

deptproc Process DEPT data (M)

padept Perform adept analysis and plot resulting spectra (C)

pldept Plot DEPT data, edited or unedited (M)

th Threshold (P)

## aexppl Automatic plot of spectral expansion (M)

Syntax: aexppl<(expansion\_factor)>

Description: Plots automatically expansions of given regions. Regions have to be defined

first by using the region command or by using the cursors in ds.

Arguments: expansion factor is a spectral expansion factor in units of Hz/mm. The

default is 2 Hz/mm.

Examples: aexppl

aexppl(20)

See also: NMR Spectroscopy User Guide

Related: ds Display a spectrum (C)

region Divide spectrum into regions (C)

## ai Select absolute-intensity mode (C)

Syntax: ai

Description: Selects the absolute-intensity display mode in which the scale is kept constant

from spectrum to spectrum to allow comparison of peak heights from one spectrum to another. The alternative is the normalized-intensity display mode (nm) in which spectra are scaled so that the largest peak in the spectrum is vs mm high. The modes are mutually exclusive—the system is always in either nm

or ai mode. Enter aig? to determine which mode is currently active.

See also: NMR Spectroscopy User Guide

Related: aig Absolute intensity group (P)

nm Select normalized-intensity mode (C)

Vertical scale (P)

## aig Absolute-intensity group (P)

Description: Contains the result of the ai or nm command. aig is not set in the usual way

but can be queried (aig?) to determine which display mode is active.

Values: 'ai' indicates the absolute-intensity display mode is active.

'nm' indicates the normalized-intensity display mode is active.

See also: NMR Spectroscopy User Guide

Related: ai Select absolute intensity mode (C)

dmg Display mode in directly detected dimension (P)

nm Select normalized-intensity mode (C)
? Display individual parameter value (C)

### alfa Set alfa delay before acquisition (P)

Description:

After the final event in the pulse sequence, including any receiver gate times occurring following the final pulse, acquisition occurs after a delay. This delay includes a fixed part, alfa, and a variable part, 1/(beta\*fb).

- On systems with 4-pole Butterworth filters, beta is 2.
- On systems with 8-pole Butterworth (200-kHz) filters, beta is 3.8.
- On systems with 8-pole elliptical filters, beta is 1.29.
- On Systems with 4-pole Bessel filters, beta is 2.3 (only systems with 2-MHz and 5-MHz Analog-to-Digital Converter boards use this filter).

Because the total delay before acquisition is the sum of alfa and 1/ (beta\*fb), it is possible to shorten the delay beyond "normal" values by setting alfa negative (to a maximum of 1/ (beta\*fb)). The macros hoult and calfa frequently result in such negative values of alfa.

To set alfa to a negative number, use either the **setvalue** command to enter a specific value of alfa, or use the **setlimit** command to allow entry of negative values of alfa directly from the keyboard.

Values: 0 to 100,000,000; in μs.

See also: NMR Spectroscopy User Guide

Related: calfa Recalculate alfa so that first-order phase is zero (M)

fb Filter bandwidth (P)

hoult Set parameters alfa and rof2 according to Hoult (M)

rof2 Receiver gating time following pulse (P)
setlimit Set limits of a parameter in a tree (C)
setlp0 Set parameters for zero linear phase (M)
setvalue Set value of any parameter in a tree (C)

## alock Automatic lock control (P)

Description: Governs Autolock control following the insertion of a sample with change or

sample, and following initiation of an acquisition with the go, ga, or au.
Manual adjustment of lock power, gain, and phase is possible using the acqi

command.

Values: Possible values are 'a', 'auto', 'n', 's', 'samp', 'u', or 'y', where:

'a' or 'auto' selects the optimizing Autolock function, which performs a lock capture and an automatic lock power and gain adjustment before data acquisition begins (lock phase is *not* optimized).

'n' leaves the lock in its current state.

's' or 'samp' selects the optimizing Autolock function, which performs a lock capture and an automatic lock power and gain adjustment before data acquisition begins (lock phase is *not* optimized) but only if the sample has just been changed.

'u' turns lock off so that the experiment runs unlocked.

'y' turns on the software Autolock function, which searches for the correct Z0 value only.

See also: NMR Spectroscopy User Guide

Related: acqi Interactive acquisition display process (C)

au Submit experiment to acquisition and process data (C)

change Submit a change sample experiment to acquisition (M)

ga Submit experiment to acquisition and FT the result (C)

gf Prepare parameters for FID/spectrum display in acqi (M)

go Submit experiment to acquisition (C)

lock Submit an Autolock experiment to acquisition (C)

Submit change sample, Autoshim experiment to acquisition (M)

## ampmode Independent control of amplifier mode (P)

Description:

Gives override capability over the default selection of amplifier modes. Unless overridden, the usage of rf channels determines whether the amplifier for a channel is in pulse, CW (continuous wave), or idle mode:

- Observe channel is set to the pulse mode.
- Other used channels are set to the CW mode.
- Any unused channels are set to the idle mode.

The ampmode parameter can be used to override this selection.

ampmode does not normally exist but can be created by the user with the command create('ampmode', 'flag').

Values

List of characters in which the mode of the first amplifier is determined by the first character, the mode of the second amplifier by the second character, and so on. For each amplifier, one of the following characters is used:

- 'c' selects CW mode.
- 'i' selects idle mode.
- 'p' selects pulse mode.
- 'd' selects default behavior.

For example, ampmode='ddp' selects default behavior for the first two amplifiers and forces the third channel amplifier into pulse mode. Additional filtering is usually required when an amplifier in the same band as the observe amplifier is placed in the CW mode.

See also: VnmrJ User Programming

Related: create Create new parameter in a parameter tree (C)

dn Nucleus for the first decoupler (P)
tn Nucleus for observe transmitter (P)

### amptype Amplifier type (P)

Description:

Specifies the type of amplifier on each rf channel of the spectrometer. The value is set in the Spectrometer Configuration window (opened from config) using the label Type of Amplifier.

For each channel, the types are Class C, Linear Full Band, Linear Low Band, Linear Broadband, or, for the fourth channel only, Shared. Selecting Shared means that the amplifier is fully configured for the third channel, and that the fourth channel shares this amplifier with the third channel.

When a type is selected for a channel, a letter (one of the values described below) is added to the value of amptype. For example, a system already set to Linear Full Band on the observe transmitter channel and the first decoupler channel would have amptype='aa'. Selecting the third channel as Linear Low Band would set amptype='aal'. Finally, selecting Shared for the fourth channel would set amptype='aaln'.

Values:

'a' indicates the channel uses a linear full-band amplifier. A full-band amplifier has two outputs: 12 MHz to  $^{31}$ P, and  $^{19}$ F $^{/1}$ H.

- 'b' indicates the system uses a linear broadband amplifier.
- 'c' indicates the system uses a class C amplifier.
- $^{\,\prime}$  1  $^{\,\prime}$  indicates the channel uses a linear low-band amplifier. A low-band amplifier has one output from 12 MHz to  $^{31}P$  only.

'n' indicates the fourth channel shares a linear amplifier with the third.

See also: NMR Spectroscopy User Guide, VnmrJ User Programming

Related: config Display current configuration and possibly change it (M)

## analyz Calculate standard peak height (M)

Syntax: analyz(\$option,\$title)

Description: Macro to calculate average peak height and standard deviation and/or average

phase and standard deviation.

Arguments: \$option ='n' for amplitude and phase, 'a' for amplitude only, and 'p' for phase

only. The \$title option puts a title on the plot.

 $Examples: \quad \hbox{analyz-Does analysis for both amplitude and phase}$ 

analyz ('p') - Does analysis for phase only

analyz('n','Stability') - Does analysis for amplitude and phase and

puts title "Stability" on the plot.

## analyze Generalized curve fitting (C)

Syntax: (curve fitting) analyze('expfit', xarray<, options>)

(regression) analyze('expfit', 'regression'<, options>)

Description: Provides interface to curve fitting program expfit (using the curve fitting

syntax), supplying  $\ensuremath{\mathtt{expfit}}$  with input data in the form of the text file analyze.inp in the current experiment.  $\ensuremath{\mathtt{expfit}}$  can be called from UNIX

with the syntax:

expfit options <analyze.inp >analyze.list

expfit does a least-squares curve fitting to the data supplied in
analyze.inp. Macros are available for the specialized uses of analyze,
such as the 'T1' and 'kinetics' options. These macros avoid the need to
select options and get the correct file format.

In the regression mode (using the regression syntax above), the type of curve fitting, ('poly1', ...) must be selected. The regression section in the manual *NMR Spectroscopy User Guide* gives the input file format and describes the menus that permit choices indirectly through menu buttons.

The text file analyze.inp for the options 'T1', 'T2', 'kinetics', 'contact\_time', and 'regression' contains the following lines (note that (1), (2), (3), etc. do not appear in the file but are used to identify lines in the explanation):

- (1) <text line>
- (2) <text line>
- (3) npeaks npairs <xscale> <yscale>
- (4) <NEXT npairs1>
- (5) peaks
- (6) x y
- (6) x y
- (4) <NEXT npairs2>
- (5) peaks
- (6) x y
- (6) x y

. . .

Line-by-line explanation:

- (1) Optional descriptive text line, for regression only. Omit line otherwise.
- (2) Optional y-axis title, for regression only. Omit line otherwise.

A

- (3) Line containing an integer for the number of peaks (npeaks) followed by another integer for the number of (x, y) pairs per peak (npairs). If regression, the *x*-scale type and *y*-scale type are also listed.
- (4) In the regression mode, a line beginning with the keyword NEXT is inserted at the start of each data set when the number of pairs per peak is variable. In this case, the number of (x, y) pairs for the peak (npair1, npair2, etc.) is also given on the line.
- (5) Peak index.
- (6) Data pairs, one to a line, are listed by peak in the following order:

```
x y (first peak, first pair)
x y (first peak, second pair)
...
x y (second peak, first pair)
x y (second peak, second pair)
```

In the regression mode, the line beginning with NEXT is inserted at the start of the data for each peak when the number of pairs per peak is variable. In this case, the header contains the maximum number of pairs for any peak.

For 'T1', 'T2', 'kinetics', and 'contact\_time', information from the file fp.out and values of the arrayed parameter xarray are used to construct the file; thus, it is necessary to run fp prior to analyze.

For regression, analyze.inp is made by running expl('regression'). If the regression mode is not selected, analyze.inp may be slightly different.

In addition to output to the standard output, which is usually directed to analyze.list, expfit makes a file analyze.out, which is used by expl to display the results of the analysis.

User-supplied analysis programs can be called by analyze in place of expfit. Such programs should read their input from stdin and write the output listing to stdout. No analyze. out file needs to be generated unless display by expl is desired. Use the program expfit as a model.

Arguments:

'expfit' is a required first argument.

xarray is the name of the parameter array holding x-values in 'T1', 'T2',
'kinetics', and 'contact\_time', and is used only with these options.

'regression' sets regression mode and signifies generalized curve fitting with choices 'poly1', 'poly2', 'poly3', and 'exp'.

options are any of the following keywords:

- 'T1' sets  $T_1$  analysis (the default).
- 'T2' sets  $T_2$  analysis.
- 'kinetics' sets kinetics analysis, with decreasing peak height.
- 'increment' sets kinetics analysis, with increasing peak height.
- 'list' makes an extended listing for each peak.
- 'diffusion' sets a special analysis for diffusion experiments.
- 'contact\_time' sets a special analysis for solids cross-polarization spin-lock experiments.
- 'poly1' sets a linear fitting. It is used in regression mode only.
- 'poly2' sets a quadratic fitting. It is used in regression mode only.
- 'poly3' sets a cubic fitting It is used in regression mode only.
- 'exp' sets exponential curve fitting. It is used in regression mode only.

Examples: analyze('expfit','d2','T1','list')

analyze('expfit','pad',kinetics','list')
analyze('expfit','p2','contact\_time','list')
analyze('expfit','regression','poly1','list')

See also: NMR Spectroscopy User Guide

Related: contact time MAS cross-polarization spin-lock contact time (M)

expfit Least squares fit to polynomial or exponential curve (U)

expl Display exponential or polynomial curves (C)
pexpl Plot exponential or polynomial curves (C)
kini Kinetics analysis, increasing intensity (M)

t1  $T_1$  exponential analysis (M) t2  $T_2$  exponential analysis (M)

## ap Print out "all" parameters (C)

Applicability: VnmrJ

Syntax: ap('template name',<'filename'>)

Description: Print a parameter list. The *User Programming* Manual describes the rules for

building a template for the ap commands. The string parameter ap normally controls how the command, ap, displays the parameters. Use command paramvi('ap') to modify the ap parameter. The ap command writes the parameter list to a file if filename is provided as the second argument.

Arguments: template\_name template name must be the first argument.

filename optional, name of file to which the parameters are written.

Examples: ap('ap', 'apout') — writes the parameter list using defined by the ap

parameter to the file apout.

ap('newap')

See also: NMR Spectroscopy User Guide; VnmrJ User Programming

Related: addpar Add selected parameters to the current experiment (M)

"All" parameters display control (P)

dg Display group of acquisition/processing parameters (C) hpa Plot parameters on special preprinted chart paper (C)

pap Plot out "all" parameters (C)

paramvi Edit a variable and its attributes with vi text editor (C)

ppa Plot a parameter list in "English" (M)

## ap "All" parameters display control (P)

Description: Controls the display of the ap and pap commands to print and plot a parameter

list. Use paramvi ('ap') to modify the string value of ap.

See also: NMR Spectroscopy User Guide; VnmrJ User Programming

Related: ap Print out "all" parameters (C)

dg Display group of acquisition/processing parameters (C)

pap Plot out "all" parameters (C)

Edit a variable and its attributes with vi text editor (C)

#### apa Plot parameters automatically (M)

Syntax: apa

Description: Selects automatically the appropriate command on different plotter devices to

plot the parameter list.

See also: VnmrJ User Programming

Related: hpa Plot parameters on special preprinted chart paper (C)

Plot a parameter list in "English" (M)

aph Automatic phase adjustment of spectra (C)

Syntax: aph<:\$ok,\$rp,\$lp>

Description: Automatically calculates the phase parameters 1p and rp required to produce

an absorption mode spectrum and applies these parameters to the current spectrum. Values calculated do *not* depend on the initial values of lp and rp.

Arguments: \$0k is 1 if the phase adjustment succeeds, or 0 if the adjustment fails.

\$rp is the calculated value of rp. If \$rp is requested as a return value, rp is

returned but not applied to the current spectrum.

\$1p is the calculated value of 1p. If \$1p is requested as a return value, 1p is

returned but not applied to the current spectrum.

See also: NMR Spectroscopy User Guide

Related: aph0 Automatic phase of zero-order term (C)

aphx Perform optimized automatic phasing (M)

First-order phase in directly detected dimension (P)

Zero-order phase in directly detected dimension (P)

aph0 Automatic phase of zero-order term (C)

Syntax: aph0<:\$ok,\$rp,\$lp>

Description: Automatically adjusts only the zero-order frequency-independent term rp and

does not rely on the frequency-dependent term lp being previously adjusted. In favorable circumstances, spectra may be obtained in such a way that only rp is expected to change. In these cases, if lp has been determined for one spectrum, then rp only can be computer-adjusted for subsequent spectra by aph0 ("aphzero"). Note that aph0 does not correctly phase an exactly on-resonance peak.

Arguments: \$0k is 1 if the phase adjustment succeeds, or 0 if the adjustment fails.

\$rp is the calculated value of rp.

\$1p is the current value of 1p.

See also: NMR Spectroscopy User Guide

Related: aph Automatic phase adjustment of spectra (C)

aphx Perform optimized automatic phasing (M)

Prirst-order phase in directly detected dimension (P)
Zero-order phase in directly detected dimension (P)

aphb Auto phasing for Bruker data (C)

Syntax: aphb<(threshold)>

Description: Phases Bruker data using the autophasing program.

Arguments: threshold determines if a data point is large enough to qualify it as part of a

peak. If no argument is given, or if the value is equal to or less than 0, the

threshold is calculated from the spectrum.

Examples: aphb

aphb(2)

See also: NMR Spectroscopy User Guide

Related: aph Automatic phase adjustment of spectra (C)

aph0 Automatic phase of zero-order term only (C)

aphx Perform optimized automatic phasing (M)

Syntax: aphx

Description: Optimizes parameters and arguments for the aph command. aphx first

performs an aph then calculates a theoretical value for lp. If lp set by the aph is different from the calculated value by 10 per cent, the calculated value is used

and an aph0 is performed.

See also: NMR Spectroscopy User Guide

Related: aph Automatic phase adjustment of spectra (C)

aph0 Automatic phase of zero-order term only (C)

First order phase along directly detected dimension (P)

## appdirs Starts Applications Directory Editor (M)

Applicability: ALL

Syntax: appdirs

Description: The **appdirs** macro brings up an editor to set the applications directories. The

top section of the editor has rows consisting of a menu and two entry boxes.

Values: Menu selections:

Enabled — enable an application directory.

Disabled — disable an application directory.

Remove(d) — initial setting for other row and the and empty entry boxes. Set an application directory menu to Remove(d) to completely remove it.

#### Fields in each row:

Applications directory path.

A comment can be added to the second entry box.

### Radio-button choices:

Save as private applications directories — sets the applications directories for the current operator only.

Reset to system default applications directories — removes any private applications directories and return to the standard default set.

Save the applications directories for global use — available only to users with write permission for VnmrJ system files. A name must be provided for this choice. This will affect all users the administrator has set that name as their appdirs setting. The Varian default names are Experimental, Walkup, Imaging, and LcNmrMs.

#### **Buttons:**

OK — exit the editor and apply the selections made in the editor.

Cancel — exit the editor and abort the editor session, making no changes to the

applications directories.

See also: VnmrJ Installation and Administration

Related: exists Checks if parameter, file, or macro exists and file type (C)

## A

## appmode Application mode (P)

Description: A global parameter that allows selection of specialized system applications

 $modes, such \ as \ imaging, \ by \ setting \ the \ global \ parameters \ \verb|sysmaclibpath|,$ 

sysmenulipath, and syshelppath.

For example, in /vnmr/maclib is a subdirectory maclib.imaging that contains macros used primarily with imaging applications. Similarly, in / vnmr/menulib is a subdirectory menulib.imaging for imaging-related menus. By separating the imaging macros and menus into subdirectories, access to imaging-specific macros and menus is more convenient. This separation also allows minor modifications to some macros and menus while retaining the names that are in common use or required by other VnmrJ commands.

The value of appmode are set from either the System settings dialog in the

Utilities menu or the VnmrJ Admin interface.

Values: 'standard' sets standard application mode.

'imaging' sets imaging application mode.

'autotest' sets autotest application mode

## apptype Application type (P)

Description: Specifies the application type, the group of pulse sequences to which a pulse

sequence belongs. It is used by the execpars macros to specify the actions executed by the protocol for a pulse sequence. The actions are common to the

group of pulse sequences specified by the apptype.

Values: See the execpars directory in /vnmr.

See also: VnmrJ Imaging, User Guide and NMR Spectroscopy User Guide

Related: cqexp Load experiment from protocol (M)

execpars

execsetup

execprep

execprescan

execpreocess

execute setup macro (P)

Execute prepare macro (P)

execprescan

execpreocess

Execute prescan macro (P)

execpreocess

Execute processing macro (P)

execplot

Execute plotting macro (P)

sqexp Load experiment from protocol (M)

## Apt Set up parameters for APT experiment (M)

Description: Converts a parameter set to the APT (attached proton test) experiment.

See also: NMR Spectroscopy User Guide

Related: aptaph Automatic processing for APT spectra (M)

Capt Automated carbon and APT acquisition (M)

hcapt Automated proton, carbon, and APT acquisition (M)

## aptaph Automatic processing for APT spectra (M)

Syntax: aptaph

Description: Automatically phases APT spectra.
See also: NMR Spectroscopy User Guide

Related: Apt Set up parameters for APT pulse sequence (M)

## array Easy entry of linearly spaced array values (M)

Syntax: array<(parameter<,number steps,start,step size)>

Description: Arrays a parameter to the number of steps, starting value and step size given by

the user. All values of the array will satisfy the limits of the parameter.

If array is typed with none or only some of its arguments, you enter an

interactive mode in which you are asked for the missing values.

Arguments: parameter is the name of the parameter to be arrayed. The default is an

interactive mode in which you are prompted for the parameter. Only numeric

parameters can be arrayed.

number\_steps is the number of values of the parameter. The default is an

interactive mode in which you are prompted for the number of steps.

start is the starting value of the parameter array. The default is an interactive

mode in which you are prompted for the starting value.

step\_size is the magnitude of the difference between elements in the array. The default is an interactive mode in which you are prompted for the step size.

Examples: array

array('pw')

array('tof',40,1400,-50)

See also: NMR Spectroscopy User Guide

## array Parameter order and precedence (P)

Description: Whenever an array of one or more parameters is set up, the string parameter

array tells the system the name of the parameter or parameters that are arrayed and the order and precedence in which the arraying is to take place. The parameter array is automatically updated when acquisition parameters are set. "Diagonal arrays" (those corresponding to using parentheses in the parameter

array) must be entered by hand.

Values: '' (two single quotes with no space between) indicates no parameter is arrayed.

'x' indicates the parameter x is arrayed.

'x, y' indicates the parameters x and y are arrayed, with y taking precedence. That is, the order of the experiments is  $x_1y_1$ ,  $x_1y_2$ ,...  $x_1y_n$ ,  $x_2y_1$ ,  $x_2y_2$ ,...  $x_2y_n$ ,...  $x_my_n$ , with a total of  $m \times n$  experiments being performed.

'y, x' indicates the parameters x and y are arrayed, with x taking precedence. That is, the order of the experiments is  $x_1y_1$ ,  $x_2y_1$ ,...  $x_ny_1$ ,  $x_1y_2$ ,  $x_2y_2$ ,...  $x_my_2$ ,...  $x_my_n$ , with total of  $m \times n$  experiments being performed.

'(x,y)' indicates the parameters x and y are jointly arrayed. The number of elements of the parameters x and y must be identical, and the order of experiments is  $x_1y_1, x_2y_2, \dots x_ny_n$ , with n experiments being performed.

Joint arrays can have up to 10 parameters. Regular multiple arrays can have up to 20 parameters, with each parameter being either a simple parameter or a diagonal array. The total number of elements in all arrays can be  $2^{32}$ –1.

See also: NMR Spectroscopy User Guide

Related: array Easy entry of linearly spaced array values (M)

### arraydim Dimension of experiment (P)

Description: After calculates the dimension of an experiment, the result is put into

the parameter arraydim. If an experiment is arrayed, arraydim is the

product of the size of the arrays.

See also: NMR Spectroscopy User Guide

Related: calcdim Calculate dimension of experiment (C)

celem Completed FID elements (P)

## A

asin Find arc sine of number (C)

Syntax: asin(value)<:n>

Description: Finds the arc sine (also called the inverse sine) of a number.

Arguments: value is a number in the range of  $\pm 1.0$ .

n is a return argument giving the arc sine, in radians, of value. The default is

to display the arc sine value in the status window.

Examples: asin(.5)

asin(val):asin\_val

See also: VnmrJ User Programming

Related: sin Find sine value of an angle (C)

asize Make plot resolution along  $f_1$  and  $f_2$  the same (M)

Syntax: asize

Description: Adjusts the 2D display parameters (sc, wc, sc2, and wc2) so that the

displayed resolution along both  $f_1$  and  $f_2$  is the same. It is not suggested for heteronuclear experiments where the chemical shift spread of one nucleus is

much greater than that of the other.

See also: NMR Spectroscopy User Guide

Related: Start of chart (P)

Start of chart in second direction (P)

WC Width of chart (P)

wc2 Width of chart in second direction (P)

assign Assign transitions to experimental lines (M)

Syntax: (1) assign<('mark')>

(2) assign(transistion number, line number)

Description: Assigns the nearest calculated transition to the lines from a dll or nll listing

after spinll has placed them in slfreq. All lines may not be assigned and transitions must be greater than sth. The next spins ('iterate') determines new parameters to minimize the differences in position of the

assigned pairs.

Arguments: 'mark' makes assign use the lines selected with the mark button in place of

dll. The results of the mark operation are stored in the file mark1d.out,

which is cleared by the command mark ('reset').

transition number is a single calculated transition number that is

assigned to a line from the dll listing.

line\_number is the index of the line from the dll listing. Setting line number=0 removes an assignment from a calculated transition.

Examples: assign

assign('mark')
assign(4,0)

See also: NMR Spectroscopy User Guide

Related: dll Display listed line frequencies and intensities (C)

mark Determine intensity of the spectrum at a point (C)

nll Find line frequencies and intensities (C)

slfreq Measured line frequencies (P)
spinll Set up slfreq array (M)

spins Perform spin simulation calculation (C)
sth Minimum intensity threshold (P)

at Acquisition time (P)

Description: Length of time during which each FID is acquired. Since the sampling rate is

determined by the spectral width sw, the total number of data points to be acquired (2\*sw\*at) is automatically determined and displayed as the parameter np. at can be entered indirectly by using the parameter np.

Values: Number, in seconds. A value that gives a number of data points that is not a

multiple of 2 is readjusted automatically to be a multiple of 2.

See also: NMR Spectroscopy User Guide; VnmrJ User Programming

Related: np Number of data points (P)

Spectral width in directly detected dimension (P)

atan Find arc tangent of a number (C)

Syntax: atan(value)<:n>

Description: Finds the arc tangent (also called the inverse tangent) of a number.

Arguments: value is a number between  $\pi/2$  and  $-\pi/2$ .

n is a return argument giving the arc tangent, in radians, of value. The default

is to display the arc tangent value in the status window.

Examples: atan(.5)

atan(val):atan val

See also: VnmrJ User Programming

Related: sin Find sine value of an angle (C)

atan2 Find arc tangent of two numbers (C)

Syntax: atan2(y,x)<:n>

Description: Finds the arc tangent (also called the inverse tangent) of the quotient of two

numbers.

Arguments: y and x are two numbers, where the quotient y/x is between  $\pi/2$  and  $-\pi/2$  and

x is not equal to zero.

n is a return argument giving the arc tangent, in radians, of y/x. The default is

to display the arc tangent value in the status window.

Examples: atan2(1,2)

atan2(val):atan2 val

See also: VnmrJ User Programming

Related: sin Find sine value of an angle (C)

atcmd Call a macro at a specified time (M)

Description: atcmd<(<'macro'><,'timespec'><,'day'><,'cancel'>)>

Syntax: Calls a macro at the specified time. It only functions on a spectrometer. A

background VnmrJ is started to execute the command. This background VnmrJ is not started in an experiment; therefore, the macro executes a <code>jexp</code> or runs commands or macros that do not need experiment parameters. It will have

access to global and system global parameters.

Arguments: When called with arguments, at cmd updates the database with the supplied information. It does not start the process that calls the macros at the specified times. at cmd with no arguments starts the program that calls the macros at the specified times.

> timespec -- has the format hh:mm < mon tue wed thur fri sat sun> A 24 hour clock is used -- midnight is 0:0, noon is 12:00.

day -- If the optional day field is used, the command will be repeated on that day at the appointed time. The day fields are case insensitive. For monday, wednesday, and friday only a single character is needed. More can be used. For tuesday, thursday, saturday, and sunday, at least two characters must be given.

cancel -- If the cancel argument is given, it will cancel all the commands that match the supplied macro. For example, if you specify cmda to be run at 8:00 on mon and 9:00 on tue, then atcmd('cancel', 'cmda') will cancel both of them. If the macro is '', the cancel option will cancel all atcmd macros.

list -- The list argument lists the timespec for all the atomds that match the supplied macro. If the macro is '', the list option lists all of the atcmd macros and their timespecs. Optional arguments can be returned. The first is the number of atomds. The macro and timespec for each atomd can be returned.

When the command specified by at cmd is executed in background, it will be executed using the environment of the user who requested the atcmd. Also, the background VnmrJ will initially not be joined to a specific experiment.

Examples:

```
atcmd('echo(`good morning`)','8:00 mon tue wed thu
fri')
```

Displays a welcome message every weekday at 8:00 am.

```
atcmd('echo('What are you doing here on a
weekend?`)','8:00 Sat Sun')
```

Questions your intentions on the weekend.

atcmd('startNightQueue','22:00') Runs the macro startNightQueue at 22 hr. (10:00pm).

atcmd('startNightQueue','cancel') Cancels the scheduled startNightQueue cmd

atcmd('','cancel') Cancels all scheduled commands

atcmd('','list') Lists all scheduled commands

#### atext Append string to current experiment text file (M)

Syntax: atext(string)

Description: Adds a line of text to the current experiment text file.

Arguments: string is a single line of text. Examples: atext('T1 Experiment') See also: NMR Spectroscopy User Guide

Related: ctext Clear the text of the current experiment (C)

> Display text or set new text for current experiment (C) text

write Write formatted text to a device (C)

#### attval Calculate pulse width (M)

Syntax: attval (pw,tpwr)

Description: Calculates the pulse width and B<sub>1</sub> field at every transmitter power. A low

transmitter power should be used where the amplifier is not in compression.

Calculation is not valid where amplifier is in compression.

Arguments: pw is the pulse width.

tpwr is the transmitter power.

Examples: attval(7.0,59)

#### atune ProTune Present (P)

Description: Hardware configuration parameter specifying if ProTune is or is not present.

Parameter is set in the System Configuration window.

Arguments: 'y' ProTune is present

'n' ProTune not is present

See also: VnmrJ Installation and Administration
Related: wtune Specify when to tune (P)

Transmitter power used in tuning (P)

## au Submit experiment to acquisition and process data (M)

Syntax: au<(<'nocheck'><,'next'><,'wait'>)>

Description: Performs the experiment described by the current acquisition parameters,

checking the parameters <code>loc</code>, <code>spin</code>, <code>gain</code>, <code>wshim</code>, <code>load</code>, and <code>method</code> to determine the necessity to perform various actions in addition to simple data acquisition. This may involve a single FID or multiple FIDs, as in the case of arrays or 2D experiments. au causes the data to automatically be processed

according to the following parameters:

• wbs specifies what happens after each block.

• wnt specifies what happens after each FID is collected.

• wexp specifies what happens when the entire acquisition is complete (which may involve several complete FIDs in the case of 1D arrays or 2D experiments).

Before starting the experiment, au executes the two user-created macros if they exist. The first is usergo, a macro that allows the user to set up general conditions for the experiment. The second is a macro whose name is formed by go\_followed by the name of the pulse sequence (from seqfil) to be used (e.g., go\_s2pul, go\_dept). This macro allows a user to set up experiment conditions suited to a particular sequence.

conditions suited to a particular sequence.

Arguments: 'nocheck' is a keyword to override checking if there is insufficient free disk

space for the complete 1D or 2D FID data set to be acquired.

'next' is a keyword to put the experiment started with au('next') at the

head of the queue of experiments to be submitted to acquisition.

'wait' is a keyword to stop submission of experiments to acquisition until wexp processing of the experiment, started with au('wait'), is finished.

Examples: au

au('wait')

See also: NMR Spectroscopy User Guide

Related: auto au Controlling macro for automation (M)

change Submit a change sample experiment to acquisition (M) Submit experiment to acquisition and FT the result (M)

gain Receiver gain (P)

go Submit experiment to acquisition (M)

Pulse sequence setup macro called by go, ga, and au (M) go

load Load status of displayed shims (P) loc Location of sample in tray (P)

lock Submit an Autolock experiment to acquisition (C)

method Autoshim method (P)

Submit change sample, Autoshim experiment to acquisition (M) sample

seqfil Pulse sequence name (P)

shim Submit an Autoshim experiment to acquisition (C) spin Submit a spin setup experiment to acquisition (C)

Sample spin rate (P) spin

Submit a setup experiment to acquisition (M) S11

usergo Experiment setup macro called by go, ga, and au (M) Specify action when bs transients accumulate (C) whs wexp Specify action when experiment completes (C) Specify action when nt transients accumulate (C) wnt. wshim Conditions when shimming is performed (P)

#### AuCALch3i Set up autocalibration with CH3I sample (M)

Syntax: AuCALch3i

Description: Retrieves standard proton parameter set and setup for automatic calibration of

> proton (observe and decouple), carbon (observe and decouple), gcal, and C/H gradient ratio. The AuCALch3i macro is the same as the AuCALch3i1

macro.

#### Get autocalibration with CH<sub>3</sub>I sample (M) AuCALch3i1

Syntax: AuCALch3i1

Description: Retrieves standard proton parameter set and setup for automatic calibration of

> proton (observe and decouple), carbon (observe and decouple), gcal, and C/H gradient ratio. The AuCALch3i1 macro is the same as the AuCALch3i

macro.

#### AuCALch3oh Set up autocalibration with Autotest sample (M)

Syntax: AuCALch3oh

Description: Retrieves standard proton parameter set and setup for automatic calibration of

proton (observe), carbon (decouple), qcal and C/H gradient ratio. The

AuCALch30h macro is the same as the AuCALch30h1 macro.

#### AuCALch3oh1 Get autocalibration with Autotest sample (M)

Syntax: AuCALch3oh1

Retrieves standard proton parameter set and setup for automatic calibration of Description:

proton (observe), carbon (decouple), qcal and C/H gradient ratio. The

AuCALch30h1 macro is the same as the AuCALch30h macro.

#### Aucalibz0 Automatic Hz to DAC calibration for Z0 (M)

Applicability: Autocalibration routine

Syntax: Called by Augmapz0 calibration routine.

Description: Called by Augmap 20 calibration routine. Automatically calibrates lock

frequency change per Z0 DAC unit change. The calibrated value is written out

in the probe file as lkhzdac parameter

See also: System Administration.

Related: Augmapz 0 Automatic lock gradient map generation and Z0 calibration (M)

Aufindz 0 Automatic adjustment of Z0 (M)

AuCdec Carbon decoupler calibration macro (M)

Syntax: AuCdec

Description: Used by AuCALch3i and AuCALch3oh autocalibration routines to do carbon

decoupler calibrations. Calibrates high-power pulse widths and dmf.

See also: System Administration

Related: AuCALch3i Get autocalibration with CH3I sample (M)

AuCALch3oh Get autocalibration with Autotest sample (M)

dmf Decoupler modulation frequency for first decoupler (P)

AuCgrad Carbon/proton gradient ratio calibration macro (M)

Syntax: AuCgrad

Description: Used by AuCALch3i1 and AuCALch3oh1 autocalibration routines for C/H

gradient ratio calibrations.

See also: System Administration

Related: AuCALch3i1 Get autocalibration with CH<sub>3</sub>I sample (M)

AuCALch3oh1 Get autocalibration with Autotest sample (M)

AuCobs Carbon observe calibration macro (M)

Syntax: AuCobs

Description: Used by AuCALch3i1 autocalibration routines for carbon observe

calibrations.

See also: System Administration

Related: AuCALch3i1 Get autocalibration with CH3I sample (M)

audiofilter Audio filter board type (P)

Description: Sets the type of audio filter board used where the spectral width (sw) is less than

100 kHz. The filter type is set in the Spectrometer Configuration window

(opened from config) using the label Audio Filter Type.

Values: 'b' indicates the system has a 100-kHz Butterworth filter board (100 kHz

Butterworth choice in the Spectrometer Configuration window.).

'e' indicates the system has a 100-kHz elliptical filter board (100 kHz

Elliptical choice in the Spectrometer Configuration window).

'2' indicates the system has a 200-kHz Butterworth filter board (200 kHz

Butterworth choice in the Spectrometer Configuration window).

'5' indicates the system has a 500-kHz elliptical filter board (500 kHz

Elliptical choice in the Spectrometer Configuration window).

See also: System Administration

Related: config Display current configuration and possibly change it (M)

Spectral width in directly detected dimension (P)

Δ

Aufindz0 Automatic adjustment of Z0 (M)

Syntax: Aufindz0

Description: Finds z0 by doing lock 1D spectrum. The frequency is then used along with the

1khzdac value in the probe file to calculate the z0 value for a given solvent and autolocking is done. This requires previous calibration of the hzdac value

done using the Aucalibz0 macro.

See also: System Administration

Related: Aucalibz0 Automatic Hz to DAC calibration for Z0 (M)

Augcal Probe gcal calibration macro (M)

Syntax: Augcal

Description: Used by AuCALch3i1 and AuCALch3oh1 autocalibration routines for probe

gcal calibrations.

See also: System Administration

Related: AuCALch3i1 Get autocalibration with CH<sub>3</sub>I sample (M)

AuCALch3oh1 Get autocalibration with Autotest sample (M)

gcal Gradient calibration constant (P)

Augmap Automated gradient map generation (M)

Syntax: Augmap

Description: Automatically adjusts gradient level, offset, window, and pulse width to

generate a z1–z4 gradient map using a 2-Hz D<sub>2</sub>O sample. This macro is used by the Aumakegmap auto gradient map generation macro and is applicable only

for a lock gradient map.

See also: System Administration

Related: Aumakegmap Auto lock gradient map generation (M)

Size Number of z-axis shims used by gradient shimming (P)

Augmap z 0 Automatic lock gradient map generation and z0 calibration (M)

Syntax: Augmapz0

Description: Using the 2-Hz D<sub>2</sub>O sample, the augmap z 0 macro automatically creates a lock

gradient map, followed by Hz to DAC calibration of Z0 for the autolocking

procedure.

See also: System Administration

Related: Aucalibz0 Automatic Hz to DAC calibration for Z0 (M)

Aufindz 0 Automatic adjustment of Z0 (M)

AuHdec Proton decoupler calibration (M)

Syntax: AuHdec

Description: Used by AuCALch3i autocalibration routine to do proton decoupler

calibrations. Calibrates high-power pulse widths and dmf.

See also: System Administration

Related: AuCALch3i Get autocalibration with CH3I sample (M)

dmf Decoupler modulation frequency for first decoupler (P)

AuHobs Proton observe calibration macro (M)

Syntax: AuHobs

Description: Used by AuCALch3i and AuCALch3oh autocalibration routines for proton

observe calibrations.

Aumakegmap Auto lock gradient map generation (M)

Syntax: Aumakegmap (<1k or hs or H1>)

Description: Generates z1-z4 lock gradient ('lk' argument), lock homospoil ('hs'

argument), or  $^1H$  gradient map ('H1' argument). If no argument is given, the defaults is 'lk', if gradtype='nnh' to 'hs'. The doped 2-Hz  $D_2O$  should

be used for hs and lk maps. H1 map is typically done on the sample.

Automatically adjusts gradient level, offset, window, and pulse width. The map

name is automatically stored in the probe file.

AuNuc Get parameters for a given nucleus (M)

Syntax: AuNuc(nucleus, solvent)

Description: Retrieves standard parameter set for a given nucleus and adds all required

parameters for Tcl/dg driven parameters. If no parameter set exists in stdpar,

then carbon parameters are retrieved and tn changed.

auto Prepare for an automation run (C)

Applicability: Systems with an automatic sample changer.

Syntax: auto<(automation directory)>

Description: Prepares the automation directory for an automation run. auto aborts if the

spectrometer is already in automation mode.

Arguments: automation\_directory is the name of the automation directory, either an

absolute UNIX path (i.e.the first character is a "/") or a relative path (the first character is not a "/"). The default is the value of the parameter autodir. If for some reason autodir is not defined, you are prompted to provide the location of the automation directory. If not given as an argument, you are prompted for the path. If the automation directory is not present, it is created with full access

for all users. auto aborts if it fails to create this directory.

Examples: auto

auto('/home/vnmr1/autorun 620')

See also: NMR Spectroscopy User Guide, VnmrJ User Programming, VnmrJ Walkup

Related: auto au Controlling macro for automation (M)

autodir Automation directory absolute pathname (P)

autogo Start an automation run (C)

autoname Prefix for automation data file (P)

auto Automation mode active (P)

Applicability: Systems with an automatic sample changer.

Description: A global variable that shows whether or not an automation run is in progress.

Macros typically test this parameter because actions can differ between the automation and non-automation modes. The value of auto is not enterable by the user. An automation experiment is initiated with the autogo command. The auto parameter is only set to 'y' for those macros and commands that are

run as part of an automation experiment.

A

Values: 'y' indicates automation mode is active.

'n' indicates automation mode is inactive

See also: NMR Spectroscopy User Guide, VnmrJ User Programming, VnmrJ Walkup.

Related: auto au Controlling macro for automation (M)

autogo Start an automation run (C)

autora Resume suspended automation run (C)

autosa Suspend current automation run (C)

# auto au Controlling macro for automation (M)

Applicability: Systems with an automatic sample changer.

Syntax: auto au

Description: Reads sampleinfo file (defines an automation experiment) using the

lookup facility, sets the solvent and loc parameters based on the SOLVENT and SAMPLE# fields of sampleinfo, runs exec on the entry in the MACRO field, and writes the experiment text based on the TEXT field. After

that, auto au examines the value of the wexp parameter:

• If wexp is set to 'procplot', then auto\_au calls au.

• If wexp is set to 'autolist', then auto\_au inserts 'auto' as the first argument to autolist and calls au('wait').

• If wexp is set to anything else, auto au does not call au.

If no data is generated from the requested MACRO field, due to an error or some other reason, auto\_au sets the STATUS field to "No Data Requested."

auto\_au is used only during automation and should not be called directly. It provides a starting point for all automation experiments. As such, it is a

convenient point for user customization of automation.

See also: NMR Spectroscopy User Guide, VnmrJ User Programming, VnmrJ Walkup

Related: au Submit experiment to acquisition and process data (M)

auto Prepare for an automation run (C)
autolist Set up and start chained acquisition (M)

exec Execute a VnmrJ command (C)
loc Location of sample in tray (P)

Look up words and lines from a text file (C)

solvent Lock solvent (P)

wexp When experiment completes (P)

# Autobackup Back up current probe file (M)

Syntax: Autobackup

Description: Makes a copy of the probe file before starting the calibrations and prints the

current calibration file. Autobackup is called by the autocalibration routines

AuCALch3i1 and AuCALch3oh1

# autodept Automated complete analysis of DEPT data (M)

Syntax: autodept

Description: Processes DEPT spectra, plots the unedited spectra, edits the spectra, plots the

edited spectra, and prints outs editing information.

See also: NMR Spectroscopy User Guide, VnmrJ Walkup

Related: adept Automatic DEPT analysis and spectrum editing (C)

Dept Set up parameters for DEPT experiment

deptproc Process DEPT data (M)

padept Perform adept analysis and plot resulting spectra (C)

pldept Plot DEPT data, edited or unedited (M)

# autodir Automation directory absolute path (P)

Applicability: Systems with an automatic sample changer or LC-NMR accessory.

Description: When using a sample changer, autodir is a global variable that holds the

absolute path of the currently active automation directory. When VnmrJ is started, autodir is set to the absolute path of the last automation run.

When using the LC-NMR accessory, autodir specifies a directory in which

experiments using a stored queue are saved.

See also: NMR Spectroscopy User Guide

Related: auto Set up an automation directory (C)

autoname Prefix for automation data file (P)
globalauto Automation directory name (P)
walkup Walkup automation (M)

### autogo Start automation run (C)

Applicability: Systems with an automatic sample changer.

Syntax: autogo<(file<,automation directory>)>

Description: Starts an automation run. The autogo parameter cannot be entered while the

spectrometer is in automation mode. You must have an enter queue prepared to start an automation run. The queue is checked to verify that it was prepared using the enter command (autogo aborts if an error in the format is found.) Your automation directory is also checked for the presence of a non-empty enter queue (autogo aborts if the current queue in the automation directory is present and not empty). Finally, autogo checks the automation directory and runs the auto command if this directory is not present or another problem is found. When autogo completes, the system is in automation mode and your

automation run starts.

Arguments: file is the file name of your enter queue. The default is that the system

prompts you for the location of the enter queue.

automation directory is the pathname of the automation directory. The

default is the current value of the parameter autodir.

Examples: autogo

autogo('MySamples')

autogo('MySamples','/home/vnmr1/AutoRun 621')

See also: NMR Spectroscopy User Guide

Related: auto Set up an automation directory (C)

autodir Automation directory absolute path (P)
autoname Prefix for automation data file (P)

enter Enter sample information for automation run (C)

# autolist Set up and start chained acquisition (M)

Syntax: autolist(<options,>experiment1<,experiment2<,...>)

Description: Sets up parameters for chained experiments by executing the experiments given

as arguments and then starting a chained acquisition. Note that the macro au is executed as part of autolist and should not be included in the arguments to

autolist.

Arguments: options is one or more of the following keywords:

- 'auto' is a keyword to add 'wait' to the au call (e.g, au('wait', 'next')).
- 'start' is a keyword to make the first experiment in the list as one that needs to be acquired rather than processed.

experiment1, experiment2, . . . are experiments written as strings (e.g., 'dept' or 'c13'). experiment1 is the current experiment and, when it finishes, the macro procplot is called to process the data. If experiment2 is listed, that experiment is executed and then the macro au('next') is performed. For subsequent experiments, the text, solvent and temp are used from the preceding experiment. Also, the wexp parameter is reset to 'autolist' with the first experiment removed.

Examples: autolist('h1','c13','dept')

autolist('h1','hcosy')

See also: NMR Spectroscopy User Guide

Related: auto\_au Controlling macro for automation (M)

au Submit experiment to acquisition and process data (M)

he Automated proton and carbon acquisition (M)

heapt Automated proton, carbon, and APT acquisition (M)

hecorr Automated proton, carbon, and HETCOR acquisition (M)

hcosy Automated proton and COSY acquisition (M)

procplot Automatically process FIDs (M)

solvent Lock solvent (P)
temp Sample temperature (P)

wexp When experiment completes (P)

### autoname Create path for data storage (C)

Applicability: Automation

Syntax: autoname: \$path

autoname(name template):\$path

autoname(name\_template,sample\_info\_file):\$path

autoname (name\_template,sample\_info\_file,
<'keepspaces'|'replacespaces'>):\$path
autoname(name\_template,sample\_info\_file,

<,'excluded\_suffixes'<,'keepspaces'|'replace</pre>

spaces'>):\$path

Description: The autoname command determines the path for data storage during an

automation run and uses the value of a naming template (the autoname parameter by default) and the contents of a sample info file (default is sampleinfo in the current experiment) to determine this path. The path name is stored in the return argument or displayed on line 3 if no return argument is

present.

The name is prefaced with using the value of the parameter autodir or userdir+'/data/' if autodir is equal to ".

The default excluded suffix is.fid.

Arguments: No argument provided. The command uses the default autoname parameter

and sampleinfo in the current experiment directory for the path to the

sample info file. If the autoname parameter does not exist or is set to ", the default template is \$SAMPLE#: % PEAK#: %.

name\_template (no quotes) is string that contains keywords separated by substitution specifiers to represent the data storage path. Substitution specifiers in this template are either a percent sign (%) or a dollar sign (\$). The keywords are obtained from the sample\_info\_file file, if it exists, using % substitution specifiers or VNMR parameters using \$ substitution specifiers.

A template is passed directly using: autoname('\$owner\$/\$sample\$'):\$path.

Percent sign (%) substitution specifier is used with the autoname command to scan the sample\_info\_file for the text specific by keyword between the first percent sign in the template string and the next percent sign. The text specified by the keyword between the % substitution specifiers is passed to \$path.

The following percent substitutions (% keywords) for time and date are obtained from the system clock, not from the sample info file:

Keyword	Format	Description
%DATE%	YYYYMMDD	4 digit year 2 digit month 2 digit day
%TIME%	HHMMSS	2 digit each for hours, minutes, and seconds
%YR%	YYYY	4-digit year
%YR2%	YY	2-digit year
%MO%	MM	2-digit month
%DAY%	DD	2-digit day
%HR%	НН	2-digit hour
%MIN%	MM	2-digit month
%SEC%	SS	2-digit second

The following are some of the percent substitutions (% keywords) are obtained from the second argument, sample info file.

Keyword	Description
%USER%	user name
%MACRO%	macro name
%SAMPLE%	sample name
%SOLVENT%	solvent name

Version number is specified by Rn where n is an integer from 0 to 9 (default 2), as follows:

n=	Description
0	no revision digits are appended (all names must be uniquely constructed without these revision digits).
1 to 9	revision number is padded with leading zeroes to form an n-digit number. If more places are needed than specified, more zeroes are used.

>9 Rnn is still used as a search string in the sampleinfo

(more than one file. %Rn% must be specified at the end of the digit)

name template string. The revision digits are

always appended except if %R0% is used.

no %Rn% default of %R2% is used

Specify the starting number to be used when constructing the version number by appending a colon: and start number after Rn.

The default starting value is 1. A zero is not allowed.

Dollar sign (\$) substitution specifiers works in manner analogous to the percent substitution specifier, except that the text between the dollar signs is interpreted as the name of a VNMR parameter. The value of this parameter is substituted for the substitution specifier.

Numeric parameters are represented as a string and truncated to an integer value. The template, pw=\$pw\$usec, with vnmr parameter pw having a value of 12.3 produces pw=12usec01 which is appended to fid and passed to \$path. The 01 following usec is added by the %R2% default setting.

sample\_info\_file (no quotes) is the name of a text file to read for the %
substitutions passed to autoname. The file must exist.

Using the keyword 'replacespaces' uses underscores (\_) in place of spaces' in the resulting path name or the keyword 'keepspaces' retains spaces in the resulting path name.

The keyword, 'keepspaces' or 'replacespaces' is an optional argument (includes quotes). The argument is accepted as the third or fourth argument.

Solaris and Linux operating systems default to replacespaces.

A comma separated list of excluded suffixes the new path name will not use or match is specified if the third keyword is not 'keepspaces' or 'replacespaces'.

Examples: Using a \$ substitution specifier:

autoname(pw=\$pw\$usec):\$path

A \$ substitution specifier, pw=\$pw\$usec, is the name\_template and a relative path. The vnmr parameter, pw, has a value of 12.3 and the resulting filename is:pw=12usec01.fid. The path name is prefaced with the value of the parameter autodir if the name template generates a relative pathname.

Examples: Using \$ substitution specifiers and a comma separated list of suffixes:

```
autoname('$seqfil$_$tn$_','/vnmr/
conpar','.img'):$path
```

The \$ substitution specifier is; \$seqfil\$\_\$tn\$\_ the dummy info filename is; '/vnmr/conpar', and the comma separated list of excluded suffixes is .img. The path name is prefaced with seqfil\_tn\_index. Each time a file is written to the directory the command changes the index by one (see %Rn% above). The suffix is both .fid and .img. The file is named

 ${\tt gems\_H1\_03.img}$  if target directory contains  ${\tt gems\_H1\_01.fid}$  and  ${\tt gems\_H1\_02.img}.$ 

See also: NMR Spectroscopy User Guide, VnmrJ User Programming, VnmrJ Walkup

Related: autoname Temple determining the path where is data stored (P)

Svfname Determines the name used to store data (C)

svfname Specifies the filename template (P)

autoname Prefix for automation data file (P)

Applicability: Automation

Description: The autoname temple determines the resulting path where the data is stored

for an entry in the automation run and uses the contents of a sample info file (the name by default is "sampleinfo" in the current experiment) to determine this path. The path name is stored in the return argument and displayed on line 3 if

no return argument is present.

See also: NMR Spectroscopy User Guide, VnmrJ User Programming, VnmrJ Walkup

Related: autoname Determines path for data storage during an automation run (C).

autora Resume suspended automation run (C)

Applicability: Systems with an automatic sample changer.

Syntax: autora

Description: Resumes a previously suspended automation run. No matter what caused the

interruption (including autosa, power failure, or system boot-up), the system examines the condition of the automation file and resumes acquisition for all experiments that have not finished. If autora is executed while an automation

run is in progress, it has no effect.

See also: NMR Spectroscopy User Guide

Related: autosa Suspend current automation run (C)

autosa Suspend current automation run (C)

Applicability: Systems with an automatic sample changer.

Syntax: autosa

Description: Suspends the automation mode at the conclusion of the current experiment and

changes the system to the manual mode. The currently running experiment is

not interrupted.

See also: NMR Spectroscopy User Guide

Related: autora Resume suspended automation run (C)

autoscale Resume autoscaling after limits set by scalelimits macro (M)

Syntax: autoscale

Description: Returns to autoscaling in which the scale limits are determined by the expl

command such that all the data in the expl input file is displayed.

See also: NMR Spectroscopy User Guide

Related: expl Display exponential or polynomial curves (C)

scalelimits Set limits for scales in regression (M)

autostack Automatic stacking for processing and plotting arrays (M)

Syntax: autostack

Description: When processing and plotting arrayed 1D spectra, VnmrJ automatically

determines whether the stacking mode is horizontal, vertical or diagonal from the number of traces and the number of lines in the spectrum. If this automatic

function is not desirable (or makes an undesirable decision), it can be

overridden by placing the stack macro in the experiment startup macro or by calling stack before processing (or reprocessing) a spectrum. autostack switches back to automatic determination of the stack mode by destroying the

stackmode parameter.

A

See also: NMR Spectroscopy User Guide

Related: procarray Process arrayed 1D spectra (M)

plarray Plot arrayed 1D spectra (M)

stack Fix stacking mode for processing / plotting arrayed spectra (M)

stackmode Stacking control for processing (P)

autotest Open Auto Test Window (C)

Syntax: autotest

Description: Opens the Auto Test window. See also: *AutoTest Software* manual.

# autotime Displays approximate time for automation (M)

Syntax: autotime(<automation directory>)

Description: Displays approximate time for each experiment and for each location in an

automation run. If no argument is given, time is calculated for the current

automation run (enterQ).

See also: NMR Spectroscopy User Guide

Related: explist Display approximate time for current experiment chain (M)

# av Set abs. value mode in directly detected dimension (C)

Syntax: av

Description: Selects the absolute-value spectra display mode by setting the parameter dmg

to the string value 'av'. In the *absolute-value display mode*, each real point in the displayed spectrum is calculated as the square root of the sum of the squares of the real and imaginary points comprising each respective complex data point. All information, including noise, is always positive, and the relationship

between signal and noise is linear.

For multidimensional data, av has no effect on data prior to the second Fourier transform. If pmode='full', av acts in concert with commands ph1, av1,

or pwr1 to yield the resultant contour display for the 2D data.

See also: NMR Spectroscopy User Guide

Related: av1 Set abs. value mode in 1st indirectly detected dimension (C)

Set abs. value mode in 2nd indirectly detected dimension (C)

dmg Display mode in directly detected dimension (C)

dmgf Absolute-value display of FID data or spectrum in acqi (P)

ft Fourier transform 1D data (C)

ftld Fourier transform along f<sub>2</sub> dimension (C)

ft2d Fourier transform 2D data (C)

pa Set phase angle mode in directly detected dimension (C)
pal Set phase angle mode in 1st indirectly detected dimension (C)

ph Set phased mode in directly detected dimension (C)ph1 Set phased mode in 1st indirectly detected dimension (C)

pmode Processing mode for 2D data (P)

pwr1 Set power mode in 1st indirectly detected dimension (C)

wft Weigh and Fourier transform 1D data (C)
wftld Weigh and Fourier transform of 2D data (C)
wftld Weigh and Fourier transform 2D data (C)

#### Set abs. value mode in 1st indirectly detected dimension (C) av1

Syntax: av1

Description: Selects the absolute-value spectra display mode along the first indirectly

detected dimension by setting the parameter dmg1 to the value 'av1'. If the

parameter dmg1 does not exist, av1 creates it and set it to 'av1'.

In the absolute-value display mode, each real point in the displayed trace is calculated as the square root of the sum of the squares of the real and imaginary points comprising each respective complex data point. For hypercomplex data, the real-real and imaginary-real points from each respective hypercomplex data point are used in the summation. In this mode, all information, including noise, is always positive; and the relationship between signal and noise is linear.

The av1 command is only needed if mixed-mode display is desired. If the parameter dmg1 does not exist or is set to the null string, the display mode along the first indirectly detected dimension defaults to the display mode of the directly detected dimension (characterized by the parameter dmg). For the contour display of multidimensional data, the result of av1 is the same as for traces provided that pmode='partial' or pmode='' (two single quotes

with no space between).

See also: NMR Spectroscopy User Guide

Related: Set abs. value mode in directly detected dimension (C)

> Data display mode in 1st indirectly detected dimension (P) dmg1

#### Set abs. value mode in 2nd indirectly detected dimension (C) av2

Syntax: av2

Description: Selects absolute-value spectra display mode for the second indirectly detected

> dimension by setting the parameter dmg2 to the value 'av2'. If dmg2 does not exist or is set to the null string, av2 creates dmq2 and set it equal to 'av2'.

> In the *absolute-value display mode*, all information, including noise, is positive; and the relationship between signal and noise is linear. Each real point in the displayed trace is calculated as the square root of the sum of the squares of the real and imaginary points comprising each respective complex data point. For hypercomplex data, the real-real and imaginary-real points from each respective hypercomplex data point are used in the summation.

> The av2 command is only needed if mixed-mode display is desired. If the parameter dmq2 does not exist or is set to the null string, the display mode along the second indirectly detected dimension defaults to the display mode of the directly detected dimension (characterized by the parameter dmg). For the contour display of multidimensional data, the result of av2 is the same as for traces provided that pmode='partial' or pmode='' (two single quotes with no space between).

See also: NMR Spectroscopy User Guide

Related: Set abs. value mode in directly detected dimension (C)

> dmq2 Data display mode in 2nd indirectly detected dimension (P)

#### averag Calculate average and standard deviation of input (C)

Syntax: averag(number1, number2, ...):average, sd,

number arguments, sum numbers, sum squares

Description: Finds average, standard deviation, and other characteristics of a set of numbers.

Arguments: number1, number2, ... is a finite set of numbers.

> average is the average of the numbers. sd is the standard deviation of the numbers.

number\_arguments is the number of number1, number2,... arguments.

sum numbers is the sum of the numbers

sum squares is the sum of squares of the numbers.

Examples: averag(3.4, 4.3, 3.5, 5.4):r1,r2

See also: VnmrJ User Programming

### awc Additive weighting const. in directly detected dimension (P)

Description: Adds the current value of awc to each value of the weighting function along the

directly detected dimension. This dimension is often referred to as the  $f_2$  dimension in 2D data sets, the  $f_3$  dimension in 3D data sets, and so forth. awc is applied *after* the sinebell and exponential function, but *before* the Gaussian function. This allows using gf as a Gaussian apodization even when awc is

non-zero. Typical value of awc is 'n'.

See also: NMR Spectroscopy User Guide

Related: awc1 Additive weighting const. in 1st indirectly detected dimension (P)

awc2 Additive weighting const. in 2nd indirectly detected dim. (P)

Gaussian function in directly detected dimension (P)

# awc1 Additive weighting const. in 1st indirectly detected dimension (P)

Description: Adds the current value of awc1 to each value of the weighting function along

the first indirectly detected dimension This dimension is often referred to as the

f<sub>1</sub> dimension of a multidimensional data set. awc1 is analogous to the parameter awc. The "conventional" parameters (lb, gf, etc.) operate on the detected FIDs, while this "2D" parameter is used during processing of the

interferograms.

See also: NMR Spectroscopy User Guide

Related: awc Additive weighting const. in directly detected dimension (P)

# awc2 Additive weighting const. in 2nd indirectly detected dimension (P)

Description: Adds the current value of awc2 to each value of the weighting function along

the second indirectly detected dimension This dimension is often referred to as the  $f_2$  dimension of a multidimensional data set. awc2 is analogous to the parameter awc. The value of awc2 can be set with wti on the 2D

interferogram data.

See also: NMR Spectroscopy User Guide

Related: awc Additive weighting const. in directly detected dimension (P)

wti Interactive weighting (C)

# axis Provide axis labels and scaling factors (C)

Syntax: axis('fn'|'fn1'|'fn2')

<:\$axis\_label,\$freq\_scaling,\$scaling\_factor>

Description: Displays or returns values of the axis labels and scaling factors to the calling

macro. See the macro rl for an example of using this command.

Arguments: 'fn'|'fn1'|'fn2' is the Fourier number parameter for the axis of interest.

\$axis label is the axis label (e.g., ppm, kHz, cm, or ppm (sc)).

\$freq\_scaling is the divisor needed to convert from units of Hz to the units defined by the axis parameter with any scaling. axis uses the current value

of the axis parameter for that dimension and also checks for axis scaling using the corresponding scalesw, scalesw1, or scalesw2 parameter.

\$scaling\_factor is a second scaling factor, determined solely by the scalesw type of parameter. This last scaling factor is independent of the value of the axis parameter.

Examples: axis('fn')

axis('fn1'):\$lab,\$fr,\$scl

See also: VnmrJ User Programming

Related: axis Axis label for displays and plots (P)

Set reference line (M)

scaleswScale spectral width in directly detected dimension (P)scalesw1Scale spectral width in 1st indirectly detected dimension (P)scalesw2Scale spectral width in 2nd indirectly detected dimension (P)

# axis Axis label for displays and plots (P)

Applicability: Certain arguments work only if system has the proper hardware.

Description: Specifies the units for the axis display and plot.

For 1D experiments, axis uses a single letter that includes 'h' for Hz, 'p' for ppm, and 'k' for kHz (e.g., axis='h').

For 2D experiments, axis uses two letters, with the first letter describing the detected spectral axis  $(f_2)$ , and the second letter describing the indirectly detected axis  $(f_1)$ . Thus axis='ph' is appropriate for a homonuclear 2D-J experiment, with a referenced ppm scale along the spectral axis and an axis in Hz ('h') along the J-axis. axis='pp' is appropriate for COSY or NOESY experiments.

For 3D experiments, axis uses three letters with the first letter describing the detected spectral axis  $(f_3)$ , the second letter describing the first indirectly detected axis  $(f_1)$ , and the third letter specifying the second indirectly detected axis  $(f_2)$ .

The special letter d is used to reference the indirectly detected axis to the parts per million of the decoupler channel, as appropriate for heteronuclear chemical shift correlation experiments, which would typically have axis='pd'. The letter n is used to suppress the axis display on one or both axes (e.g., axis='nn', axis='pn').

For systems with multiple decouplers, the characters '1', '2', and '3' can be used to reference an axis relative to the frequency of that decoupler. Setting axis='p1' is effectively the same as axis='pd'.

Values: '1' sets the axis label for units of ppm relative to the first decoupler.

'2' sets the axis label for units of ppm relative to the second decoupler.

' 3' sets the axis label for units of ppm relative to the third decoupler.

'c' sets the axis label for units of centimeters.

'd' sets the axis label for units of ppm relative to the first decoupler.

'h' sets the axis label for units of hertz.

'k' sets the axis label for units of kilohertz.

'm' sets the axis label for units of millimeters.

'n' sets no axis label display.

'p' sets the axis label for units of ppm relative to the observe transmitter.

'u' sets the axis label for units of micrometers.

See also: NMR Spectroscopy User Guide

Related: axis Provide axis labels and scaling factors (C)

axisf Axis label for FID displays and plots (P)

A

dscale Display scale below spectrum or FID (C)
pscale Plot scale below spectrum or FID (C)

# axisf Axis label for FID displays and plots (P)

Description: Specifies the units for the FID axis display and plot. To create the FID display

parameters axisf, dotflag, vpf, vpfi, crf, and deltaf (if the parameter set is older and lacks these parameters), enter addpar('fid').

Values: 's' sets the axis label for units of seconds.

'm' sets the axis label for units of ms. 'u' sets the axis label for units of  $\mu$ s.

'n' sets no axis label display.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

axis Axis label for displays and plots (P)
dscale Display scale below spectrum or FID (C)
pscale Plot scale below spectrum or FID (C)

# B

bandinfo
Shaped pulse information for calibration (M)
banner
Display message with large characters (C)
bc
1D and 2D baseline correction (C)

beepoff Turn beeper off (C)
beepon Turn beeper on (C)

bigendian Determine system byte order (C)

binom Set up parameters for BINOM pulse sequence (M)

bioref Bio-NMR Referencing (P)

bootup Macro executed automatically (M)

box Draw a box on a plotter or graphics display (C)
boxes Draw boxes selected by the mark command (M)

bpa Plot boxed parameters (M)

br24 Set up parameters for BR24 pulse sequence (M)

bs Block size (P)

# bandinfo Shaped pulse information for calibration (M)

Applicability: Information only useful on systems capable of shaped pulse generation.

Syntax: bandinfo<(shape,width<,ref\_power>)>:duration,power

Description: Displays a table containing the duration and the predicted 90° pulse power

setting for the pulse shape and bandwidth given by the arguments. No parameter settings are changed. The necessary data is contained in the shapeinfo file

in the shapelib subdirectory.

Arguments: If bandinfo is run without arguments, prompts operator for input

shape is the name of the shape. The default is system prompts for a name.

width is the bandwidth, in Hz, desired for the pulse.

ref\_power is value of tpwr to which pw90 is set. The default is 55 dB.

duration is the duration, in  $\mu$ s, of the pulse. power is the predicted 90° pulse power setting.

Examples: bandinfo

bandinfo('sinc',10):pw,tpwr

See also: User Programming

Related: pulseinfo Shaped pulse information for calibration (M)

pw90 90° pulse width (P)

Observe transmitter power level with linear amplifiers (P)

### banner Display message with large characters (C)

Syntax: banner(message<,color>)

Description: Displays text as large-size characters on the graphics windows.

Arguments: message is the text to be displayed. If the text includes a single quotation mark

('), it must be preceded by a backslash (\'). Multiline displays are available by

inserting two backslashes (\\) between lines. Any undefined characters are displayed as a "bug" shape.

color is the color of text on a color display: 'red', 'yellow', 'green', 'cyan', 'blue', 'magenta', and 'white'. The default is 'yellow'.

Examples: banner('banner sample')

banner('Don\'t Touch','blue')

See also: *User Programming* 

# bc 1D and 2D baseline correction (C)

Description: Makes 1D or 2D baseline correction using a spline or a second to twentieth

order polynomial fitting of predefined baseline regions. bc defines every other integral (those integrals that disappear when intmod='partial') as baseline and attempts to correct these points to zero.

### 1D baseline correction

Syntax: bc<(n|'unbc'<,nsubregion<,minpoints<,minregion>>>)>

Description: Performs a 1D baseline correction. The nonintegrated parts of the spectrum (i.e.,

every odd region between integral reset points, or the integral gaps with <code>intmod='partial'</code>) are divided into baseline subregions. The number of baseline subregions in each area are adjusted as possible, so that the subregions are more or less equal in size. Finally, the "center of gravity" (midpoint in *x and* average of the *y* values in the region) for each of the subregions is calculated.

Arguments: n is an integer from 1 to 20 for the baseline correction step. A polynomial of

the (n-1)th order is calculated "through" the "baseline points" using the Chebychev least-squares fitting algorithm, and that polynomial function is subtracted from the spectrum. The coefficients of the polynomial are written into the file cureexp+'/bc.out'. The default is 1(a spline fit).

'unbc' is a keyword to make bc read in the coefficients from the file written by the previous bc operation and reverse that operation. This option is only functional for polynomials with two or more coefficients performing baseline correction operations on 1D spectra or individual 2D traces (i.e., baseline corrections cannot be undone with the default spline correction).

nsubregion defines the number of subregions (minimum 3, maximum 400). By default, the total number of subregions is 20 (if fn<2048), 40 (if fn=2048 or fn=4096), or 80 (if fn>4096).

minpoints sets the minimum number of data points required in an integral gap for bc to regard it as baseline. Use this to exclude small, nonintegrated areas between close signals. The default is fn/1000 (but at least 3).

minregion defines the minimum number of subregions assigned to each baseline area. The default is 1.

Examples: bc

bc(3)

bc('unbc')

bc (1, 200, 8, 2) gives a spline correction using 200 baseline subregions, a gap of 8 data points between two (even) integral regions is regarded as baseline, and each baseline area is split into at least two subregions.

### 2D baseline correction

Syntax: bc(trace\_direction<,num\_coeff><,trace\_start>

<,trace end>)

Description: 2D baseline correction can be performed on three types of 2D data:

- f2 spectra (trace direction='f2') after the first half of a 2D FT (wft1da).
- f2 traces (trace direction='f2') after a full 2D FT (wft2da).
- f1 traces (trace direction='f1') after a full 2D FT (wft2da).

Arguments: trace direction specifies the direction, 'f1' or 'f2', along which the 2D baseline correction is to take place.

> num coeff is the number of coefficients, from 1 to 20, used in the fitting procedure. The default value is 1, which gives a spline fit. A value of 2 gives a linear baseline fit (a + bx), a value of 3 gives a quadratic fit  $(a + bx + cx^2)$ , etc. The maximum value (20) gives a 19th-order polynomial fit with 20 coefficients.

> trace start is the trace number for the spectrum on which the 2D baseline correction is to start. It must lie within the appropriate range or an error results.

> trace end is the trace number for the spectrum on which the 2D baseline correction is to end. It must lie within the appropriate range or an error results.

Examples: bc('f1')

bc('f2',3)

bc('f2',3,10,60)

See also: NMR Spectroscopy User Guide

Related: Calculate spectral drift correction (C)

Fourier number in directly detected dimension (P)

intmod Integral display mode (P) Mode for 2D data display (P) trace

wft1da Weight and Fourier transform phase-sensitive data (M) wft2da Weight and Fourier transform phase-sensitive data (M)

#### beepoff Turn beeper off (C)

Description: Turns off the beeper sound so that the system does not use sound to warn the

user when errors occur. The default is the beeper is turned on.

See also: User Programming

Related: beepon Turn beeper on (C)

#### Turn beeper on (C) beepon

Syntax: beepon

Description: Turns on the beeper sound so that the user hears a sound when errors occur. The

default is the beeper is turned on.

See also: User Programming

Related: beepoff Turn beeper off (C)

#### bigendian Determine system byte order (C)

Syntax: bigendian: \$type

Description: The bigendian command determines the system byte order for storing numbers.

One architecture is Big Endian, used by Sun computers with the "Sparc"

CPU'S. The other architecture is Little Endian, used by most PCs.

Return values to argument \$type:

1 if it is a "Big Endian" system. 0 if it is a "Little Endian" system. This command should rarely be used. Its only current use is when imaging .fdf files are created. The .fdf file headers can specify whether the data is stored as

big or little endian.

Set up parameters for BINOM pulse sequence (M) binom

Description: Sets up a binomial water suppression pulse sequence.

See also: NMR Spectroscopy User Guide

bioref Bio-NMR Referencing (P)

Applicability: All

Syntax: bioref='<y or n>'

Description: Flag, global or local, for Bio-NMR Referencing. Setting the flag

(bioref='y') sets the system to bio-NMR referencing (based on nuctables/nuctabrefBio) rather than standard IUPAC / organic chemistry referencing (based on nuctables/nuctabref). Bio-NMR referencing uses DSS for nuclei such as <sup>13</sup>C and liquid NH<sub>3</sub> for <sup>15</sup>N.

Creating bioref as a local parameter (create ('bioref', 'flag') creates a local flag) permits its use for a specific case. The parameter can be created as a local parameter and saved with a standard parameter set (stdpar/ N15) to enable bio-NMR referencing for a specific nucleus. The local value of

the parameter takes precedence over the global parameter.

create('bioref','flag','global') — creates a global flag. setenumeral('bioref',2,'y','n','global') — sets the

possible values of a string parameter in a parameter tree.

Examples: bioref='y' sets referencing to use nuctables/nuctabrefBio

Related: create Create new parameter in a parameter tree (C)

Macro executed automatically (M) bootup

Syntax: bootup<(foreground)>

Description: Executed automatically when VnmrJ is started up. The bootup macro displays

> a message, looks for a macro login in the user's local maclib directory and executes it (if found), starts Acqstat and acqi (acqi is not run if system is configured as a workstation), and then starts the menu system. This set of actions can be modified on a per user basis by constructing custom bootup or login macros in the user's maclib directory. A custom login macro is preferred because all custom bootup macros are overridden whenever a new

VnmrJ release is installed.

foreground is 0 if VnmrJ is being run in the foreground or nonzero if being Arguments:

run in the background. This argument is passed to the login macro.

See also: User Programming

Related: acqi Interactive acquisition display process (C)

> Acqstat Bring up the acquisition status display (U)

Draw a box on a plotter or graphics display (C) box

Syntax: box(<'keywords',>x1mm,x2mm,y1mm,y2mm

<, 'nolimit'>) <: r1, r2>

Description: Draws a box on a plotter or a graphics display.

'keywords' identifies the output device ('qraphics' | 'plotter'), drawing mode ('xor' | 'normal'), and drawing capability

('newovly'|'ovly'|'ovlyC').

• 'graphics'|'plotter' is a keyword for the output device. The default is 'plotter'. The output selected is passed to subsequent pen, move, or draw commands and remains active until a different output is specified.

- 'xor', 'normal' is a keyword for the drawing mode when using the 'graphics' output device. The default is 'normal'. In the 'xor' mode, if a line is drawn such that one or more points of the line are in common with a previous 'xor' line, the common points are erased. In the normal mode, the common points remain. The mode selected is passed to subsequent pen, move, and draw commands and remains active until a different mode is specified.
- 'newovly', 'ovly' and 'ovlyC' are keywords that specify an interactive drawing capability that is slightly slower than the 'xor' mode but more consistent in color. 'newovly' clears any previous draws, boxes, and writes made with the 'ovly' modes and draws the figure. 'ovly' draws without clearing so that multi-segment figures can be created. 'ovlyC' clears without drawing.

x1mm is the left edge of the box, x2mm is the right edge, y1mm is the bottom, and y2mm is the top. The location of the edges are given in plotter units (mm on most plots) and are scaled in mm for the graphics display. (If units are in Hz or ppm, you can use the hztomm command to convert units.)

'nolimit' allows the box to extend outside the limits determined by the parameters sc, wc, sc2, and wc2.

r1, r2 return the location of the upper left corner of the box.

Examples: box('plotter', 20, 100, 40, 150)

box(25,105,45,155,'nolimit'):r1,r2

See also: NMR Spectroscopy User Guide

Related: gin Return current mouse position and button values (C)

> Convert positions from Hz or ppm to plotter units (C) hztomm

Start of chart (P) SC

sc2 Start of chart in second direction (P)

Width of chart (P) WC

Width of chart in second direction (P) wc2

Maximum width of chart (P) wcmax

#### boxes Draw boxes selected by the mark command (M)

Syntax: boxes<('graphics'|'plotter')>

Description: Draws boxes on a plotter or a graphics display with the location of the edges

> given in Hz. The data to make the boxes is stored in the mark2d.out file produced by the mark command. If there is no data in mark2d.out, a box is drawn from the current cursor positions. The boxes command also numbers

the boxes above the upper left corner.

'graphics' | 'plotter' is a keyword to send output to the graphics Arguments:

display or to the plotter, respectively. The default is 'graphics'.

Examples: boxes

boxes('plotter')

See also: NMR Spectroscopy User Guide

Related: mark Determine intensity of spectrum at a point (C)

bpa Plot boxed parameters (M)

Syntax: bpa:\$sc2 minimum

Description: Plots a box around the entire chart (assuming blank paper) and then plots

"chemist-style" parameters in boxes along the lower edge of the chart. bpa is the same as ppa, but with a different layout. Both ppa and bpa behave somewhat naively if the pulse sequence is more complex, but they were

designed primarily for chemists, not for spectroscopists.

Arguments: sc2 minimum returns the minimum value for sc2 to plot a scale properly. To

use the command pir, vp has to be set to a non-zero value.

See also: NMR Spectroscopy User Guide

Related: apa Plot parameters automatically (M)

pap Plot out "all" parameters (C)

pir Plot integral amplitudes below spectrum (C)
ppa Plot a parameter list in "English" (M)
sc2 Start of chart in second direction (P)
vp Vertical position of spectrum (P)

br24 Set up parameters for BR24 pulse sequence (M)

Applicability: Systems with solids hardware.

Description: Converts a FLIPFLOP, MREV8, or S2PUL parameter set into a BR24 solids

line-narrowing multiple-pulse sequence.

See also: User Guide: Solid-State NMR

Related: cylbr24 Set up parameters for cycled BR24 pulse sequence (M)

flipflop Set up parameters for FLIPFLOP pulse sequence (M)
mrev8 Set up parameters for MREV8 pulse sequence (M)

s2pul Set up standard two-pulse sequence (M)

bs Block size (P)

Description: Directs the acquisition computer, as data are acquired, to periodically store a

block of data on the disk, from where it can be read by the host computer.

CAUTION: If bs='n', block size storage is disabled and data are stored on disk only

at the end of the experiment. If the experiment is aborted prior to

termination, data will be lost.

Values: 1 to 32767 transients, 'n'

See also: NMR Spectroscopy User Guide

Related: wbs Specify action when bs transients accumulate (C)

wbs When block size (P)

# C

C13 Automated carbon acquisition (M)
C13p Process 1D carbon spectra (M)

calcdim Calculate dimension of experiment (C)

calfa Recalculate alfa so that first-order phase is zero (M)
calibflag Correct systematic errors in DOSY experiments (P)
calibrate Start a dialog for autocalibration routines (M)
callacq Utility macro to call Acq command (M)
capt Automated carbon and APT acquisition (M)
Carbon Set up parameters for 13C experiment (M)
cat Display one or more text files in text window (C)

cattn Coarse attenuator type (P)
cd Change working directory (C)
cdc Cancel drift correction (C)

cdeptAutomated carbon and DEPT acquisition (M)cdumpPrints the current graphics screen (M)

Time the current graphics serven

celem Completed FID elements (P)

centerSet display limits for center of screen (C)centerswMove cursor to center of spectrum (M)

centersw1Move cursor to center of spectrum in 1st indirect dimension (M)centersw2Move cursor to center of spectrum in 2nd indirect dimension (M)

Create an experiment (M)

Current FID (P)

cfpmultCalculate first-point multiplier for 2D experiments (M)changeSubmit a change sample experiment to acquisition (M)

checkstringFind and replace unwanted characters (C)chiliConfControl flag set by ecc\_on and ecc\_off (P)

Cigar2j3j Convert the parameter to a CIGAR2j3j experiment (M)

claClear all line assignments (M)claCalculated transition number (P)clampCalculated transition amplitude (P)

cleanexp Remove old files and directories from an experiment (M)

clear a window (C)

cleardosy Delete temporarily saved data in current sub experiment (M)

clfreq Calculated transition frequency (P)

clindex Index of experimental frequency of a transition (P)

clradd Clear add/subtract experiment (C)

color Select plotting colors from a graphical interface (M)

combiplateView a color map for visual analysis of VAST microtiter plate (U)combishowDisplay regions (red, green, and blue) in CombiPlate window (M)

compressfid Compress double-precision FID data (M,U)

config Display current configuration and possibly change it (M)

confirm message using the mouse (C)

Console System console type (P)

contact\_time MAS cross-polarization spin-lock contact time (M)

continue Movie Continue movie in either forward or backward direction (C)

**convert** Convert data set from a VXR-style system (M,U)

convertbru Convert Bruker data (M,U)

Copy a file (C)

Find cosine value of an angle (C)

Cosy Convert the parameter to a COSY experiment (M)

Set up parameters for phase-sensitive COSY pulse sequence (M)

cp Copy a file (C)
cp Cycle phase (P)

 cpmgt2
 Set up parameters for CPMGT2 pulse sequence (M)

 cpos\_cvt
 Convert data set from a VXR-style system (M,U)

 cptmp
 Copy experiment data into experiment subfile (M)

Create pbox shape file (M)

cqexp Load experiment from protocol (M)

cqfindz0 Run an experiment to find the value of z0 (M)
cqgmap Perform gradient shimming utility functions (M)

cqinit Initialize liquids study queue (M)

cqparsCreate study queue parameters for liquids (M)cqplotMacro to perform generic 2D plot (M)cqprotocolMacro to create protocols (M)cqresetReset study queue parameters (M)

cqsavestudy Macro to save study queue parameters (M

cqwtmenu Macro to set weighting functions from a panel (M)

Cr Cursor position in directly detected dimension (P)

Cr1 Cursor position in 1st indirectly detected dimension (P)

Cr2 Cursor position in 2nd indirectly detected dimension (P)

crcom Create user macro without using text editor (M)
create Create new parameter in a parameter tree (C)

createqcomp Create qcomp parameter (M)

Current time-domain cursor position (P)

crl Clear reference line in directly detected dimension (M)
crl1 Clear reference line in 1st indirectly detected dimension (M)
crl2 Clear reference line in 2nd indirectly detected dimension (M)
crmode Current state of the cursors in df, ds, or dconi programs (P)

crof2 Recalculate rof2 so that lp = 0 (M)

 $\begin{tabular}{lll} $\tt cryo\_noisetest & Run Cold Probe conditioning experiments (M) \\ &\tt cryoclient & Start the CryoBay Monitor program (M, U) \\ \end{tabular}$ 

Completed transients (P)

Clear the text of the current experiment (C)

curexpCurrent experiment directory (P)curscanScan currently in progress (P)

curwinCurrent window (P)cutoffData truncation limit (P)

cyclenoe Set up parameters for CYCLENOE pulse sequence (M)
cylbr24 Set up parameters for cycled BR24 pulse sequence (M)

Set up parameters for cycled MREV8 pulse sequence (M)

Cz Clear integral reset points (C)

# c13 Automated carbon acquisition (M)

Syntax: c13<(solvent)>

Description: Prepares parameters for automatically acquiring a standard <sup>13</sup>C spectrum. The

parameter wexp is set to 'procplot' for standard processing. If c13 is used as the command for automation via the enter command, the au is supplied automatically and should not be entered on the MACRO line of the enter program. However, it is possible to customize the standard c13 macro on the MACRO line by following it with additional commands and parameters. For example, c13 nt=1 uses the standard c13 setup but with only one transient.

Arguments: solvent is the name of the solvent. In automation mode the solvent is

supplied by the enter program. The default is 'CDC13'.

Examples: c13

c13('DMSO')

See also: NMR Spectroscopy User Guide

Related: au Submit experiment to acquisition and process data (M)

c13p Process of 1D carbon spectra (M)

enter Enter sample information for automation run (C)

proc1d Processing macro for simple (non-arrayed) 1D spectra (M)

procplot Automatically process FIDs (M)
wexp When experiment completes (P)

# c13p Process 1D carbon spectra (M)

Syntax: c13p

Description: Processes non-arrayed 1D carbon spectra using a set of standard macros. c13p

is called by the procld macro, but can also be used directly. Fully automatic processing (up to a point where a spectrum could be plotted) is provided: Fourier transformation (using pre-set weighting functions), automatic phasing (aphx macro), automatic integration (integrate macro if required only), vertical scale adjustment (vsadjc macro), avoiding excessive noise (noish macro), threshold adjustment (thadj macro), and referencing to the TMS

signal if present (setref macro then tmsref macro).

See also: NMR Spectroscopy User Guide

Related: aphx Perform optimized automatic phasing (M)

c13 Automated carbon acquisition (M)
integrate Automatically integrate 1D spectrum (M)

noislm Limit noise in spectrum (M)

proc1d Processing macro for simple (non-arrayed) 1D spectra (M)

setref Set frequency referencing for proton spectra (M)

thadj Adjust threshold (M)

tmsref Reference spectrum to TMS line (M)
vsadjc Adjust vertical scale for carbon spectra (M)

# calcdim Calculate dimension of experiment (C)

Syntax: calcdim

Description: Calculates the dimension of an experiment and puts the result into the parameter

arraydim. If an experiment is arrayed, arraydim is the product of the size

of the arrays.

See also: NMR Spectroscopy User Guide

Related: arraydim Dimension of experiment (P)

calfa Recalculate alfa so that first-order phase is zero (M)

Syntax: calfa

Description: Based upon the current alfa and lp values, calfa calculates a new value for

alfa so that the first-order phase parameter lp is rendered approximately 0. When digital filtering is active (dsp='r' or dsp='i'), calfa also adjusts rof2 as well as alfa. For calfa to work properly, a trial spectrum must be obtained and phased to pure absorption. This spectrum provides calfa with the current alfa and lp values. calfa pertains to processing 2D data. Unless lp is approximately 0, fpmult will affect both the dc offset and the curvature

of the spectrum.

See also: NMR Spectroscopy User Guide

Related: alfa Set alfa delay before acquisition (P)

cfpmult Calculate first-point multiplier for 2D experiments (M)

crof2 Recalculate rof2 so that lp = 0 (M)
dc Calculate spectral drift correction (C)
dsp Type of DSP for data acquisition (P)
fpmult First-point multiplier for np FID data (P)

hoult Set parameters alfa and rof2 according to Hoult (M)

1p First-order phase in directly detected dimension (P)

rof2 Receiver gating time following pulse (P)

calibflag Correct systematic errors in DOSY experiments (P)

Syntax: calibflag

Description: Corrects systematic errors in DOSY experiments.

Values: 'y' corrects systematic deviations in DOSY analysis.

'n' omits gradient correction in DOSY analysis.

See also: NMR Spectroscopy User Guide

Related: dosy Process DOSY experiments (M)

calibrate Start a dialog for autocalibration routines (M)

Syntax: calibrate

Description: Starts a dialog for autocalibration routines.

callacq Utility macro to call Acq command (M)

Syntax: callacq(arg string)

Description: Utility macro to construct a string to pass to psg via the Acq() command. This

macro should be used only by users with advanced knowledge. A well-constructed argument string is required. The motivation for this macro is to make the 'qo' macro re-entrant, while still synchronizing with VnmrJ.

Arguments: arg\_string is a character string constructed from a macro.

Examples: callacq(\$callback)

Related: go Submit experiment to acquisition (M)

reqparcheck Flag which enables/disables required parameters (P)
reqparclear Clears the parameters in required parameter list (M)

required parameters (P)

required parameters are set (M)

# capt Automated carbon and APT acquisition (M)

Syntax: capt<(solvent)>

Description: Prepares parameters for automatically acquiring a standard <sup>13</sup>C spectrum,

followed by an APT experiment. In non-automation mode, the carbon and APT spectra are acquired in the experiment in which capt is entered. Following

acquisition completes, the commands rttmp('C13') and

rttmp('apt') can be used for further processing of the carbon and APT

spectra, respectively.

Arguments: solvent is name of the solvent used. In automation mode, the enter

program supplies name. In non-automation mode, the default is 'cdcl3'.

Syntax: capt au

capt('dmso')

See also: NMR Spectroscopy User Guide

Related: Apt Prepare parameters for APT experiment (M)

Automated carbon acquisition (M)

enter Enter sample information for automation run (C)

rttmp Retrieve experiment subfile (M)

# Carbon Set up parameters for 13C experiment (M)

Description: Set up parameters for <sup>13</sup>C experiment

# cat Display one or more text files in text window (C)

 $Syntax: \verb| cat(file1<,file2,...>)|\\$ 

Description: Displays the contents of one or more text files on the text window. It pauses after

the window has filled and waits for the user to indicate whether it should display

more or should terminate.

Arguments: file1, file2, . . . are the names of the files to be displayed.

Examples: cat('/vnmr/manual/cat')

cat('/vnmr/manual/cat','/vnmr/manual/cattn')

See also: NMR Spectroscopy User Guide

# cattn Coarse attenuator type (P)

Applicability: Systems with a coarse attenuator.

Description: Identifies the type of coarse attenuator if this attenuator is present on the current

rf channel. The value of cattn is set in the Spectrometer Configuration window (opened by entering config) using the label Coarse Attenuator.

Values: 0 for no coarse attenuator, as in the case with class C amplifiers (Not Present

choice in Spectrometer Configuration window).

79 for standard systems (79 dB choice in Spectrometer Configuration window).

127 for imaging attenuator (63.5 dB SIS choice in Spectrometer Configuration

window).

63 for deuterium decoupler channel.

See also: VnmrJ Installation and Administration

Related: config Display current configuration and possibly change it (M)

fattn Fine attenuator (P)

Observe transmitter power level with linear amplifiers (P)

# cd Change working directory (C)

Syntax: cd<(directory)>

Description: Changes current working directory to another directory.

Arguments: directory is the name of the directory that becomes the new current working

directory. The change is made only if the directory name already exists and the

user has permission to be in the directory. If no argument is included, cd changes the current working directory to the user's home directory.

Examples: cd

cd(userdir+'/exp1')

cd('/home/george/vnmrsys')

See also: NMR Spectroscopy User Guide

Related: pwd Display current working directory (C)

# cdc Cancel drift correction (C)

Syntax: cdc

Description: Turns off the drift correction started by the dc command and resets the spectral

drift correction parameters lvl (level) and tlt (tilt) to zero.

See also: NMR Spectroscopy User Guide

Related: dc Calculate spectral drift correction (C)

dcg Drift correction group (P)

lvl Zero-order baseline correction (P) tlt First-order baseline correction (P)

# cdept Automated carbon and DEPT acquisition (M)

Syntax: cdept<(solvent)>

Description: Prepares parameters for automatically acquiring a standard <sup>13</sup>C spectrum,

followed by a DEPT experiment. In non-automation mode, the carbon and DEPT spectra are acquired in the experiment in which cdept was entered. Following the completion of the acquisition, the rttmp('C13') and rttmp('dept') commands can be used for further processing of the carbon

and DEPT spectra, respectively.

Arguments: solvent is name of the solvent used. In automation mode, the enter program

supplies name. In non-automation mode, the default is 'cdcl3'.

Examples: cdept au

cdept('DMSO')

See also: NMR Spectroscopy User Guide

Related: adept Automatic DEPT analysis and spectrum editing (C)

C13 Automated carbon acquisition (M)

dept Prepare parameters for DEPT experiment (M)

enter Enter sample information for automation run (C)

rttmp Retrieve experiment subfile (M)

cdump Prints the current graphics screen (M)

Syntax: cdump('filename')

Description: cdump takes the current display and sends it to the current printer. If an optional

filename is passed as an argument, the current display will be saved in the print subdirectory of the user's vnmrsys directory. This directory will be created if is does not already exist. If the filename passed to the cdump macro is an absolute pathname, i.e., it starts with a '/' character, that pathname

will be used.

If the current display is saved as a file, the format of the file is specified by the printformat parameter. It can be set to the following values. as for

PostScript formatted output.

japed for Joint Photographic Experts Group JFIF formatted output.

nag for Portable Network Graphics formatted output.

celem Completed FID elements (P)

Description: Indicates the current number of completed FIDs in an experiment. When go or

au is entered, celem is set to 0. As each FID acquisition is completed, celem is updated to reflect this. This parameter is most useful in conjunction with wbs,

wnt, wexp, and werr processing commands.

See also: NMR Spectroscopy User Guide

Related: arraydim Dimension of experiment (P)

au Submit experiment to acquisition and process data (C)

go Submit experiment to acquisition (C)

Number of increments in 1st indirectly detected dimension (P)

wbs Specify action when bs transients accumulate (C)

werr Specify action when error occurs (C)

wexp Specify action when experiment completes (C)
wnt Specify action when nt transients accumulate (C)

center Set display limits for center of screen (C)

Description: Sets parameters sc and wc (horizontal control) and parameters sc2 and wc2

(vertical control) to produce a display (and subsequent plot) in the center portion of the screen (and page). For 2D data, space is left for the scales.

See also: NMR Spectroscopy User Guide

Related: full Set display limits for a full screen (C)

fullt Set display limits for full screen with room for traces (C)

left Set display limits for left half of screen (C)
right Set display limits for right half of screen (C)

Start of chart (P)

Start of chart in second direction (P)

wc Width of chart (P)

wc2 Width of chart in second direction (P)

centersw Move cursor to center of spectrum (M)

Description: Sets cursor position parameter cr in the directly detected dimension for the

center of the spectrum.

See also: NMR Spectroscopy User Guide

Related: centersw1 Move cursor to center of spectrum in 1st indirect dimension (M)

centersw2 Move cursor to center of spectrum in 2nd indirect dimension (M)

Cursor position in directly detected dimension (P)

centersw1 Move cursor to center of spectrum in 1st indirect dimension (M)

Description: Sets cursor position parameter cr1 in the first indirectly detected dimension to

the center of the spectrum.

See also: NMR Spectroscopy User Guide

Related: centersw Move cursor to center of spectrum (M)

Cursor position in 1st indirectly detected dimension (P)

centersw2 Move cursor to center of spectrum in 2nd indirect dimension (M)

Description: Sets cursor position parameter cr2 in the second indirectly detected dimension

to the center of the spectrum.

See also: NMR Spectroscopy User Guide

Related: centersw Move cursor to center of spectrum (M)

Cursor position in 2nd indirectly detected dimension (P)

cexp Create an experiment (M)

Syntax: cexp(<experiment dir,>experiment number)

Description: Creates an experiment as a temporary workspace that can hold a complete 1D,

2D, or 3D data set. Up to 9999 experiments can be created. Experiment 5 is special because it is the add-subtract experiment. cexp creates the appropriate

jexpxxx macro so that the newly created experiment can be joined.

Arguments: experiment dir specifies the path of the directory in which the particular

experiment is to be created. If experiment\_dir is not entered, the default is

the user directory specified by userdir.

experiment number specifies the number, from 1 to 9999, of the

experiment to be created.

Examples: cexp(3)

cexp('/data',2)

See also: NMR Spectroscopy User Guide

Related: delexp Delete an experiment (C)

jexp Join existing experiment (C)

userdir User directory (P)

cf Current FID (P)

Description: Specifies which FID to operate on when working with multi-FID data. All

subsequent operations such as Fourier transformation are applied to the selected

data block.

When an experiment acquires nf number of data segments through explicit acquisition, cf indicates the cfth FID to use. For example, in the COSY-NOESY experiment with nf=2, cf=1 would select the COSY part of the

experiment, and cf=2 would select the NOESY part.

Values: 1 through the value of parameter nf.

See also: NMR Spectroscopy User Guide

Related: nf Number of FIDs (P)

### cfpmult Calculate first-point multiplier for 2D experiments (M)

Description:

Calculates an fpmult value for the dataset, which is then used by wft2da. For 2D experiments, such as NOESY, run cfpmult on the transformed first increment, prior to entering wft2da, to minimize "f2 ridges" in the final 2D spectrum. To do this manually for a 2D dataset, enter fpmult=1.0 wft (1) cdc in the command line and note whether the spectrum (essentially the baseline) moves up or down when dc is typed. Vary the value of fpmult until the dc correction (jump in the baseline) is as small as possible. With care, fpmult can be set to two decimal places. Typical values for fpmult range from 1.00 to 2.00. The default value is 1.0.

This calculation only needs to be performed for cosine-type experiments, such as NOESY, where both the t<sub>2</sub> FID and the t<sub>1</sub> interferogram decay. cfpmult might give incorrect values for first increments of experiments having baseline distortions (e.g., water suppression with 11-echo or 1331); in such cases, manual optimization of fpmult is more suitable.

When processing 2D data, unless the parameter 1p is approximately 0, fpmult affects both the dc offset and the curvature of the spectrum. See the entries for alfa and calfa for more information.

See also: NMR Spectroscopy User Guide

Related: alfa Set alfa delay before acquisition (P)

calfa Recalculate alfa so that first-order phase is zero (M)

crof2 Recalculate rof2 so that lp = 0 (M)

dc Calculate spectral drift correction (C)

fpmult First point multiplier for np FID data (P)

lp First-order phase in directly detected dimension (P) wft2da Weight and Fourier transform phase-sensitive data (M)

### change Submit a change sample experiment to acquisition (M)

Applicability: Systems with automatic sample changer.

Description: Removes the sample currently in the probe and loads the sample currently in

sample location loc. change runs in the acquisition computer and is inoperative if loc is 0 and/or traymax is 'n' or 0. change also sets all

hardware according to the current parameters.

See also: NMR Spectroscopy User Guide

Related: au Submit experiment to acquisition and process data (C)

ga Submit experiment to acquisition and FT the result (C)

Submit experiment to acquisition (C)
Location of sample in tray (P)

lock Submit an autolock experiment to acquisition (C)

Submit change sample, Autoshim experiment to acquisition (M)

shimSubmit an Autoshim experiment to acquisition (C)spinSubmit a spin setup experiment to acquisition (C)suSubmit a setup experiment to acquisition (M)

traymax Sample changer tray size (P)

# checkstring Find and replace unwanted characters (C)

Syntax: checkstring('\$VALUE', variable):variable

Description: checkstring is used panel to check and replace user-entered strings like

samplename, notebook, or page for Unix-unfriendly characters:

""(blank space) , ; : \* ! ? (" ") [" "] {" "} <" "> ~ # \$ & /

Data may be saved to unexpected directories (or not at all) with Save Data Setup (used for automatic saving of NMR data) if operating system special characters

are used within a filename.

An error/warning message is issued and the respective character(s) is/are replaced with an underscore, \_. Multiple consecutive characters are replaced by one single underscore. Example: samplename = 'special type of (new) sample' becomes 'special type of new sample'.

chiliConf Control flag set by ecc\_on and ecc\_off (P)

Applicability: Systems with Varian, Inc. Cold Probes

Description: Control flag set by ecc\_on and ecc\_off macros

 $Values: \quad \hbox{$\mathbb{E} --$ enable PSG control of ECC}$ 

n — disable PSG control of ECC

Related: ecc\_on Turns on eddy current compensation for Cold Probes (M)

ecc\_off Turns off eddy current compensation for Cold Probes (M)

Cigar2j3j Convert the parameter to a CIGAR2j3j experiment (M)

Syntax: Convert the parameter to a CIGAR2j3j experiment.

cla Clear all line assignments (M)

Syntax: cla

Description: Clears the line assignment parameters clindex and slfreq for spin

simulation iteration, which matches simulated spectra to actual data.

See also: NMR Spectroscopy User Guide

Related: assign Assign transitions to experimental lines (M)

dla Display line assignments (M)

clindex Index of experimental frequency of a transition (P)

slfreq Measured line frequencies (P)

cla Calculated transition number (P)

Description: A global arrayed parameter that stores the transition number of calculated

transitions of the spin simulation program when they are above a threshold set by sth. In the iterative mode, the cla value of an assigned transition is associated with an experimental frequency whose index is the clindex value.

See also: NMR Spectroscopy User Guide

Related: clamp Calculated transition amplitude (P)

clfreq Calculated transition frequency (P)

clindex Index of experimental frequency of a transition (P)

sth Minimum intensity threshold (P)

clamp Calculated transition amplitude (P)

Description: A global arrayed parameter that stores the transition amplitude of calculated

transitions of the spin simulation program when they are above a threshold set

by the parameter sth. Enter dla('long') to display clamp.

See also: NMR Spectroscopy User Guide

Related: Cla Calculated transition number (P)

clfreq Calculated transition frequency (P)

clindex Index of experimental frequency of a transition (P)

dla Display line assignments (C) sth Minimum intensity threshold (P)

cleanexp Remove old files and directories from an experiment (M)

Syntax: cleanexp<(file1<,file2<,...>>)>

Description: Removes experiment subfiles from chained experiments that exist in an

experiment directory. cleanexp only cleans the currently active experiment.

Arguments: file1, file2, ... are specific experiment subfiles to be removed. If no

argument is given, all files in curexp/subexp are removed.

Examples: cleanexp

cleanexp('H1','relayh')

See also: NMR Spectroscopy User Guide

Related: curexp Current experiment directory (P)

hccorr Automated proton, carbon, and HETCOR acquisition (M)

hcosy Automated proton and COSY acquisition (M)

clear Clear a window (C)

Syntax: clear<(window number)>

Description: Clears one of the four windows on the GraphOn terminal (status, input,

graphics, text) or one of the two windows on the Sun (text and graphics).

Arguments: window number is the number (1 to 4) of the window to be cleared:

• 1 clears the status window (GraphOn only)

• 2 clears the graphics window

• 3 clears the input window (GraphOn only)

• 4 clears the text window (the default value).

Examples: clear

clear(2)

See also: User Programming

cleardosy Delete temporarily saved data in current sub experiment (M)

Syntax: cleardosy

Description: Deletes any copies of DOSY data temporarily saved in the current sub

experiment.

See also: NMR Spectroscopy User Guide

Related: dosy Process DOSY experiments (M)

clfreq Calculated transition frequency (P)

Description: A global arrayed parameter that stores the transition frequency of calculated

transitions of the spin simulation program when they are above a threshold set

by the parameter sth. Enter dla to display clfreq.

See also: NMR Spectroscopy User Guide

Related: cla Calculated transition number (P)

clamp Calculated transition amplitude (P)

clindex Index of experimental frequency of a transition (P)

dla Display line assignments (M) sth Minimum intensity threshold (P)

clindex Index of experimental frequency of a transition (P)

Description: A global arrayed parameter where each value contains the index of an

experimental frequency assigned to the associated calculated transition for use in iterative spin simulation. Use assign to make the assignments. A value of

zero indicates no assignment.

See also: NMR Spectroscopy User Guide

Related: assign Assign transitions to experimental lines (M)

cla Clear line assignments (M)
cla Calculated transition number (P)
dla Display line assignments (M)

clradd Clear add/subtract experiment (C)

Description: Deletes the add/subtract experiment (exp5).

See also: NMR Spectroscopy User Guide

Related: add Add current FID to add/subtract experiment (C)

sub Subtract current FID from add/subtract experiment (C)

color Select plotting colors from a graphical interface (M)

Description: Displays a window with color palettes for selecting colors for plotting the

background of the display screen, spectrum, integral, FID, etc.

See also: NMR Spectroscopy User Guide

Related: pl Plot spectra (C)

setcolor Set colors for graphics window and for plotters (C)

combiplate View a color map for visual analysis of VAST microtiter plate (U)

Syntax: (From UNIX) combiplate

Description: Opens the CombiPlate window, which provides a map of microtiter plate,

allowing data to be viewed from individual sample wells. The window enables

viewing integral region intensities by colors and color densities.

See also: NMR Spectroscopy User Guide

Related: combishow Display regions as red, green, and blue in CombiPlate window (M)

dlivast Produce text file and process last wells (M)

combishow Display regions (red, green, and blue) in CombiPlate window (M)

Syntax: combishow(r,g,b)

Description: Displays integral regions shown on the spectrum as red (r), green (q), and blue

(b) in the CombiPlate window. CombiPlate reads the regions automatically. 1, 2, or 3 integral regions can be designated. At least one integral region must be specified. Combishow displays spectra associated with individual wells.

See also: NMR Spectroscopy User Guide

Related: combiplate View a color map for visual analysis of VAST microtiter plate (U)

> dlivast Produce text file and process last wells (M)

#### compressfid Compress double-precision FID data (M,U)

Syntax: compressfid(<inFIDdir,>outFIDdir)

(From UNIX) compressfid -i inFIDdir -o outFIDdir -f (From UNIX) compressfid -e exp number -o outFIDdir -f

Description: Compresses double-precision FID data to single-precision and updates the

> parameter dp in the file procpar. compressfid can be run through a macro interface in VnmrJ or directly at the UNIX level. In entering FID directory

names, leave off the .fid directory extension.

Arguments: inFIDdir is the double-precision FID directory to be compressed. If

inFIDdir is not entered, the default FID directory is curexp/acqfil.

outFIDdir is the FID directory to receive the output.

exp number is the number of the experiment that contains the FID data.

-i specifies that the next argument is the input FID directory.

-o specifies that the next argument is the output FID directory.

-e specifies that the next argument is the number of the experiment that contains the FID data. The -e and the -i options are mutually exclusive.

-f specifies that any existing directory with the name outFIDdir.fid is to be overwritten. Note that the macro interface always overwrites any preexisting directory with the name specified by outFIDdir.fid.

compressfid('/vnmr/fidlib/fidld', Examples:

'testfid1d')compressfid('testfid1d') (From UNIX) compressfid -e 5 -o testfid1d -f (From UNIX) compressfid -i /vnmr/fidlib/fidld -o

testfid1d -f

See also: NMR Spectroscopy User Guide

Related: Double precision (P)

#### config Display current configuration and possibly change it (M)

Syntax: config <('display')>

Displays the current system configuration parameters in a window (called the Description:

Spectrometer Configuration window). The values of the configuration

parameters can be changed if config is entered from the console without any arguments and the user has write access to the directories /vnmr and /vnmr/ conpar. If so, the user can interactively make changes to the choices in the

window.

If the user does not meet the conditions above, or if the VnmrJ administrator enters the command config('display'), instead of the interactive mode, the user is restricted to the display mode, where system information is listed in the Process tab -> Text page.

If config is entered without any arguments, or if Utilities->System Settings is selected, the program checks if the user is logged in as the administrator. If so, it runs in interactive mode; if not, it runs in display mode. By entering config('display'), vnmrl can run in the display mode instead of interactively.

In the interactive mode, a separate panel displays the options with the current choice appearing to the right. Position the mouse over the choice to be modified, then use the left button to cycle through each choice or use the right button to display a menu of all possible choices.

The Use Console Data button sets parameter values in the Spectrometer Configuration window using information captured during console startup.

This button makes config capture from the system all values shown in the Spectrometer Configuration window except Sample Changer, Sample Changer Serial Port, Rotor Synchronization, Frequency Overrange, and Upper Limit of decoupler power. For the Gradients entry, config recognizes the Performa I and Performa II modules but not other gradients. For the VT Controller entry, if VT is found, config does not change the value set, and if VT is not found, config changes the value to Not Present.

The EXIT, and SAVE button writes a new conpar configuration file before leaving. The QUIT, no SAVE button terminates the session with no modifications to the conpar file, but remember that the parameters are always set. These two buttons require use of the left button on the mouse. In the display mode, the current choices are displayed in the text window.

To send output to the printer, enter the sequence of commands printon config('display') printoff.

Commands for working with parameters (such as create, destroy, exists and setvalue) have an option to select which parameter tree the parameter is in. The systemglobal tree is the internal name for /vnmr/conpar, and it can be used to search for, modify, or create a parameter in conpar. But note that any changes made, either directly (e.g., by typing vttype=0) or by using create and similar commands, only affect parameters in memory. To permanently change parameters:

- For parameters in config, enter the change in the Spectrometer Configuration window and then quit using the Exit & Save button.
- For other parameters, after creating or changing the parameter, enter fsave('/vnmr/conpar','systemglobal').

Both methods, usually restricted to vnmr1 only, overwrite conpar.

The Spectrometer Configuration labels listed below can be changed in the interactive mode. For each label, the choices available and a short description of the label is provided. Shown in parentheses is the associated parameter, which you should refer to for further information.

- System Type: Spectrometer or Data Station. Sets the basic type of system (system).
- Console: or Imager. Sets the type of system console (Console). When go, au, or ga is entered, the value set is copied to the current experiment as the console parameter (lowercase c).
- Proton Frequency: 085, 100, 200, 300, 400, 500, 600, 700, 750, 800, 900, 3T, and 4T. Sets the resonant frequency, in MHz or tesla, of <sup>1</sup>H as determined by magnet field strength (h1freq).
- Sample Changer: None, Carousel, SMS 50 Sample, SMS 100 Sample, VAST, NMS, LC-NMR, 768 AS. Sets the type of sample changer. Set to none if a sample changer is not present or is to be disabled (traymax).
- Sample Changer Comm Port: Not Used, Port A, Port B, Ethernet. Sets the serial port used to connect the sample changer. Select Not Used if no sample changer is present (smsport).

- Shimset: Varian 13 Shims, Varian 14 Shims, Oxford 15 Shims, Oxford 18 Shims, Varian 18 Shims, Varian 20 Shims, Varian 23 Shims, Varian 26 Shims, Varian 28 Shims, Varian 29 Shims, Varian 35 Shims, Varian 40 Shims, Ultra 18 Shims, Ultra 39 Shims, and Whole Body Shims. Sets type of shim sets on system (shimset).
- Audio Filter Type: 100 kHz Elliptical, 100 kHz Butterworth 200 kHz Butterworth, 500 kHz Elliptical. If the spectral width (sw) is less than 100 kHz, sets type of audio filters used (audiofilter).
- VT Controller: Not Present, Present. Sets whether a variable temperature controller is present or not on the system (vttype).
- Maximum DMF: 9900, 32700, 2.0e6. Sets maximum frequency, in Hz, for decoupler modulation (parmax [11]).
- Max. Spectral Width: 100 kHz, 200 kHz, 500 kHz, 2 MHz, 5 MHz. Sets maximum spectral width available to a system (parmax [5]).
- AP Interface Type: Type 1, Type 2, Type 3, N/A. Sets type of AP bus interface board in the system.
- Fifo Loop Size: 63, 1024, 2048. Sets size of FIFO loop, which depends on the type of controller board in the system.
- Rotor Synchronization: Not Present, Present. Sets whether system supports the solids rotor synchronization module (rotorsync).
- Lock Frequency: (frequency entered directly). Sets lock frequency of the system. To observe NMR signals, the lock frequency value must be set correctly (lockfreq).
- IF Frequency: 10.5 MHz, 20.0 MHz.
- Number of RF Channels: 1, 2, 3, 4, 5. Selects which rf channel is listed in the Configure panel that appears in the lower section of the Spectrometer Configuration window (numrfch).
- Gradients: Not Present, Present. Sets whether system has optional gradients for the X, Y, or Z axis. If present, the gradients are listed in the Configure panel in lower section of Spectrometer Configuration window (Gradients is not associated with any parameter).
- Configure: RF Channel 1 (Obs), RF Channel 2 (Dec), RF Channel 3 (Dec2), RF Channel 4 (Dec 3), RF Channel 5 (Dec4), Gradients. Sets which labels appear in the Configure panel in lower section of Spectrometer Configuration window (Configure is not associated with any parameter)
- Type of RF: U+ Direct Synthesis, U+ H1 Only, Direct Synthesis, Broadband, Fixed Frequency, Deuterium Decoupler, SIS Modulator. Sets type of frequency generation on the current rf channel (rftype and rfchtype).
- Synthesizer: Not Present, PTS 160, PTS 200, PTS 250, PTS, 320, PTS 500, PTS 620, PTS 1000. Sets type of PTS frequency synthesizer on the current rf channel (ptsval).
- Latching: Not Present, Present. On systems equipped with a special version of the PTS frequency synthesizer, sets how frequency values are sent on the current rf channel (latch).
- Frequency Overrange: Not Present, 10000 Hz, 100000 Hz. On systems equipped with a special version of the PTS frequency synthesizer, sets the presence of a signal phase stability option on the current rf channel (overrange).

C

- Step Size: 0.1 Hz, 0.2 Hz, 1 Hz, 100 Hz. Sets frequency step size on current rf channel. (parstep[7], parstep[8], parstep[16], parstep[20]).
- Coarse Attenuator: Not Present, 63 dB, 79 dB, 63.5 dB (SIS). Sets range of coarse attenuator if this attenuator is present on the current rf channel (cattn).
- Upper Limit: (number entered directly). Sets upper limit of the coarse attenuator if this attenuator is present on the current rf channel (parmax [17], parmax [9], parmax [18], parmax [21]).
- Fine Attenuator: Not Present, Present. Sets whether a fine attenuator is present or not on the current rf channel (fattn).
- Waveform Generator: Not Present, Present. Sets whether a waveform generator board is present or not on current rf channel (rfwg).
- Type of Amplifier: Class C, Linear Full Band, Linear Low Band, Shared, Linear Broadband. (Shared is fourth channel only.) Sets type of amplifier on the current rf channel (amptype).
- X Axis, Y Axis, Z Axis: None, WFG + GCU, Performa I, Performa II/III, Performa II/III+WFG, Performa XYZ, Performa XYZ+WFG, SIS (12 bit), Homospoil. On systems with gradients, sets type of gradient for each axis. The value is set separately for each axis (gradtype).
- Gradient Coil. Detects the gradient coil configuration file that defines the current installed gradient coil (sysqcoil).

Arguments: 'display' is a keyword that the system administrator can use to make

config run in the display mode rather than the interactive mode.

Examples: config

config('display')

See also: VnmrJ Installation and Administration

Related: amptype Amplifier type (P)

audiofilter Audio filter type (P)
cattn Coarse attenuator (P)
Console System console type (P)
fattn Fine attenuator (P)
fifolpsize FIFO loop size (P)

gradtype Gradients for X, Y, and Z axes (P)
hlfreq Proton frequency of spectrometer (P)
latch Frequency synthesizer latching (P)

lockfreq Lock frequency (P)
numrfch Number of rf channels (P)

overrange Frequency synthesizer overrange (P)
parmax Parameter maximum values (P)
parmin Parameter minimum values (P)
parstep Parameter step size values (P)

ptsval PTS frequency synthesizer value (P)
rfchtype Type of rf channel (P)

rftype Type of rf enamer (r)

rftype Type of rf generation (P)

rfwg RF waveform generator (P)

rotorsync Rotor synchronization (P)

sysgcoil System gradient coil (P)

system System type (P)

traymax Sample changer tray slots (P)

vttype Variable temperature controller present (P)

## confirm Confirm message using the mouse (C)

Syntax: confirm(message):response

Description: Displays a dialog box with the specified message and two buttons: Confirm and

Cancel. Clicking on the buttons with the mouse produces a return value.

Arguments: message is a single-line muticharacter string to be shown in the dialog box.

response is 1 if the user clicks the left button of the mouse on the Confirm button or presses the Return key; response is 0 if the user clicks the mouse

on the Cancel button.

Examples: confirm('Are you sure you want pw>100?'):\$response

See also: User Programming

# Console System console type (P)

Description: A global parameter that sets the type of system console The value is usually set

using the Console label in the Spectrometer Configuration window (opened

from config).

When go, au, or ga is entered, the value of the Console parameter is copied from the systemglobal parameter tree to the current experiment and named as the console parameter (lowercase c). If console does not exist in an old parameter set, rt via fixpar creates it and sets it to ''. Both console and Console are type acquisition. Macros can use Console and console to

take conditional action based on spectrometer type.

See also: VnmrJ Installation and Administration

Related: au Submit experiment to acquisition and process data (M)

config Display current configuration and possibly change it (M)

fixpar Correct parameter characteristics in experiment (M)

ga Submit experiment to acquisition and FT the results (M)

rt Retrieve FIDs (M)

go Submit experiment to acquisition (M)

system System type (P)

# contact time MAS cross-polarization spin-lock contact time (M)

Applicability: Systems with solids module.

Description: Processes data obtained using an array of values for a pulse-length parameter. It

runs the UNIX program expfit, which does an exponential curve fitting that determines the value of *Tch* and *T1rho*. The output is matched to the equation

 $I = [S0 - (S0 - S_{inf})*exp(-T/Tch))*exp(-T/T1rho)) + S_{inf}$ 

where *Tch* is the time constant of a spin-locked cross-polarization process, and *Tlybe in relevation time of <sup>13</sup>C polarization in the proton rotating field* 

T1rho is relaxation time of <sup>13</sup>C polarization in the proton rotating field.

The required input is file fp.out from the program fp and the values of the arrayed parameter. The output table is file analyze.list in the current experiment. The file analyze.out is used by the expl to display the results.

See also: User Guide: Solid-State NMR

 $Related: \quad \ \ \text{expfit} \qquad \quad Least-squares \ fit \ to \ polynomial \ or \ exponential \ curve \ (U)$ 

expl Display polynomial/exponential curves (C)

fp Find peak heights (C)

### continueMovie Continue movie in either forward or backward direction (C)

Syntax: continueMovie(rate)

Description: Like startMovie, but can continueMovie can play a movie forward or back

ward, and, instead of always starting from the beginning, it starts from the beginning if movie has not started yet, or continues from where it was stopped

(by stopMovie). Movie direction is controlled by parameter

aipMovieSetting[3] = 1 or -1.

Arguments: aipMovieRate, or a number for the rate

See also: startMovie, stopMovie, resetMovie.

convert Convert data set from a VXR-style system (M,U)

Syntax: convert(VXR file)

(From UNIX) cpos cvt VXR file

Description: Converts data stored on a VXR-style system (VXR, XL, or Gemini) to the

format used in software. The macro convert loads the data from VXR\_file into the current experiment and converts it to the new format. The UNIX command cpos\_cvt writes the converted data in a subdirectory of the current

working directory, using the original name of the data set.

Arguments: VXR file is the name of a VXR-style file to be converted to VnmrJ style

See also: NMR Spectroscopy User Guide

Related: cpos cvt Convert data set from a VXR-style system (C,U)

decomp Decompose a VXR-style directory (C)

convertbru Convert Bruker data (M,U)

Syntax: (From UNIX) convertbru file <options>

convertbru(file<,options>)

Description: A C-language program for converting 32-bit Bruker AMX data and 24- and 32-

bit Bruker AM data into a 32-bit format compatible with the Varian sread
program. After converting the Bruker data into the new format, the converted data can be read into VnmrJ using sread and can then be processed normally.
The parameters proc and proc1 are set appropriately by sread, so that wft

or wft2da correctly processes the data.

Bruker AM parameters are converted to Varian parameters as shown in the table "AM Parameter Conversion." Bruker parameter names that do not conflict with a Varian parameter name are converted under the original name: td, fw, ds, o1, o2, ns, te, id, sfo1, sfo2, and ro. Parameters proc and proc1 are set to 'rft' for all spectra (assuming TPPI data in both dimensions).

### **AM Parameter Conversion**

Bruker	Varian	Bruker	Varian
sweeps completed	ct	sp	satdly
td	np	dp	dpwr
dw	dw	te	temp=te-273
fw	fb=1.1*sw/2	id	sw1=1/id
ds	SS	sfo1	sfrq=sfo1+o1
SW	sw	sfo2	dfrq=sfo2+o2
experiments done	ni	p#	p#
01	tof	d#	d#
02	dof	s#	s#
rd (or d1 if rd=0)	rd	ro	spin
pw (or po if pw=0)	pw	rg	gain

Bruker	Varian	Bruker	Varian
p1	pw90	date	date
de	de	time	time
ns	nt		

Bruker AMX parameters are converted to Varian parameters as shown in the table "AMX Parameter Conversion." All Bruker parameters are converted under their original names if the name doesn't conflict with the name of a Varian parameter. Arrayed Bruker parameters like P and D are converted to the names P# and D#, where # is the index into the array.

Because **sread** is limited to 8-character parameter names, the parameters routwd1# and routwd2# are converted to rtwd1# and rtwd2#.

The parameter proc is set to 'ft' when the Bruker parameter ag mod is 1, and proc is set to 'rft' when aq mod is 2. proc1 is always set to rft, assuming TPPI in t1.

If there is a file named info in the directory with the Bruker data, it is read in and put into the text file for the converted data set.

### **AMX Parameter Conversion**

Bruker	Varian	Bruker	Varian
ns (from acqu)	nt	te	temp=te-273
ns (from acqus)	ct	sfo1	sfrq=sfo1
td (from acqus)	np	sfo2	dfrq=sfo2
td (from acqu2s)	ni	01	tof
sw_h	SW	02	dof
sw_h	dw=1.0e6/sw	ro	spin
sw_h (from acqu2s)	sw1	rg	gain
fw	fb=1.1*sw/2	date	date
ds	SS	date	time
rd (or d1 if rd=0)	rd	nucleus	tn
de	de	decnuc	dn
pw (or p0 if pw=0)	pw	pulprog	pslabel
p1	pw90	pulprog	seqfil

Arguments: file is the input file name. For AMX data, file should be the name of the directory that contains the acqus, acqu2s, and fid or ser files. For AM data, file should be the name of the file containing the AM data. The file argument is not required to have a .bru extension, but if it does, the .bru extension is removed before creating the output file. Unless the -cfile option is present, the output file will have the same name as the input file, but with a .cv extension, and will be written into the current working directory.

> options for AMX and AM data are the following, which can be entered in any order as long as file comes first (options are usually not necessary, but can be used to override the default actions of convertbru):

- -bam or -bamx specifies whether input is AM or AMX data. The default is determined from name of the input file given.
- -cfile specifies that the output file is given the name specified by file and is written with .cv appended to the name

C

- -dxxx, where xxx is the decoupler frequency (it must be a value between 10.0 and 640.0 MHz). The default is to read from data set.
- -f specifies that old output file is to be overwritten. The default is to not overwrite old files.
- -olsb or -omsb specifies whether the data has the least- or most-significant byte first. For AM data, the default is determined from data set. For AMX data, the default is -olsb.
- -pxxx, where xxx is the number of 24- or 32-bit words to skip before converting data. This option is for use with -t option to skip the header in AM data without converting it. Typical header sizes are 216 or 256 words. The default is 0.
- -s3 or -s4 specifies if AM data is 24-bit (3-byte) or 32-bit (4-byte). All AMX data is 32-bit. The default is determined from the data set.
- -tall, -thdr, or -tdata specifies whether convertbru should convert the header and the data, just the header, or just the data. The default is -tall.

Examples: Convert AM data from a UNIX shell (in all these examples, the file name is arbitrarily named br data):

- convertbru br\_data determines the file format and converts the header and data in the file br data.
- convertbru br\_data -d250.0 -cout determines the file format, converts the header and data in the br\_data, sets the decoupler frequency to 250.0 MHz, and writes to an output file named out.cv in the current working directory.
- convertbru br\_data -thdr determines file format and converts only the header in the file br\_data.
- convertbru br\_data -tdata -p256 -s3 -omsb converts only the data in br\_data after skipping the 256-word header. The data is converted assuming it is 24-bit AM data words with the most-significant byte first.

#### Convert AM data from VnmrJ:

• convertbru('br\_data','-tdata','-p256','-s3', '-omsb') converts only the data in br\_data after skipping the 256-word header. The data is converted assuming it is 24-bit AM data words with the most-significant byte first.

### Convert AMX data from a UNIX shell:

• convertbru br\_data -f converts acqus and acqu2s files to ASCII, if needed, and then converts data and overwrites the existing br\_data.cv file.

# Convert AMX data from VnmrJ:

- convertbru('br\_data','-f') converts acqus and acqu2s files to ASCII, if needed, and then converts data and overwrites the existing br data.cv file.
- convertbru('br\_data','-c/home/vnmr1/bdata/data1') converts acqus and acqu2s files to ASCII, if needed, and then converts the data and writes it to /home/vnmr1/bdata/data1.cv.

See also: NMR Spectroscopy User Guide

Related: readbrutape Read Bruker data files from 9-track tape (U)

sread Read converted data into VnmrJ (C)

wft2da Weight and Fourier transform phase-sensitive data (M)

copy Copy a file (C)

Syntax: copy(<'-r',>from file,to file)<:\$res>

Description: Makes a copy of a file and is identical to the cp command. All arguments are

passed. Command will abort with no return value if an illegal file name is used.

Arguments: '-r' — keyword requesting a recursive copy (i.e., copy a directory).

from\_file — name of the file (or directory if '-r' used) to be copied.

to\_file — name of the copy of the file (or directory). If the from\_file argument has an extension (e.g., .fid), be sure the to\_file argument has the

same extension.

: \$res — variable to hold the result of the copy process.

1 is returned if the copy is successful.

0 is returned if the copy failed.

Examples: copy('-r','/home/vnmr1/vnmrsys/seqlib','/vnmr/

seqlib')

copy('/home/vnmr1/vnmrsys/seqlib/d2pul', \

'/vnmr/seqlib/d2pul')

See also: NMR Spectroscopy User Guide

Related: cp Copy a file (C)

cos Find cosine value of an angle (C)

Syntax: cos(angle)<:n>

Description: Finds the cosine of an angle.

Arguments: angle is the angle, given in radians.

n is the return value with the cosine of angle. The default is to display the

cosine value in the status window.

Examples: cos(.5)

cos(val):cos\_val

See also: User Programming

Related: sin Find sine value of an angle (C)

Cosy Convert the parameter to a COSY experiment (M)

Description: Convert the parameter to a COSY experiment.

See also: NMR Spectroscopy User Guide

Related: cosyps Set up parameters for phase-sensitive COSY pulse sequence (M)

Dqcosy
Set up parameters for double-quantum filtered COSY (M)
relayh
Set up parameters for RELAYH pulse sequence (M)

cosyps Set up parameters for phase-sensitive COSY pulse sequence (M)

Description: Sets up a phase-sensitive COSY (homonuclear correlation) experiment.

See also: NMR Spectroscopy User Guide

Related: Cosy Set up parameters for COSY pulse sequence (M)

Dqcosy
Set up parameters for double-quantum filtered COSY (M)
relayh
Set up parameters for RELAYH pulse sequence (M)

cp Copy a file (C)

Syntax: cp(<'-r',>from file,to file)<:\$res>

Description: Makes a copy of a file and is identical to the copy command. All arguments are

passed. Command will abort with no return value if an illegal file name is used.

Arguments: '-r' is a keyword requesting a recursive copy (i.e., copy a directory).

from\_file is the name of the file (or directory if '-r' used) to be copied.
to\_file is the name of the copy of the file (or directory). If the from\_file
argument has an extension (e.g., .fid), be sure the to\_file argument has

the same extension.

:\$res variable to hold the result of the copy process.

1 is returned if the copy is successfully

0 is returned if the copy failed

Examples: cp('/home/vnmr1/vnmrsys/seqlib/d2pul', \

'/vnmr/seqlib/d2pul')

cp('-r','/home/vnmr1/vnmrsys/seqlib','/vnmr/seqlib')

See also: NMR Spectroscopy User Guide
Related: Copy Copy a file (C)

cp Cycle phase (P)

Description: Sets the values that real-time variable oph is calculated as, either 0,1,2,3

(cp='y') or 0 (cp='n'). The only circumstance where setting cp='n' may be useful is when displaying an FID with acqi. If there is an imbalance between the two receiver channels, the FID displayed for acqi may show alternating dc levels. The standard gf macro that prepares parameters for the

FID display in acqi automatically handles this issue.

Values: 'y' makes oph calculate as 0,1,2,3; this is the typical value.

'n' makes oph calculate as 0.

See also: User Programming

Related: acqi Interactive acquisition display process (C)

go Submit experiment to acquisition (C)

gf Prepare parameters for FID/spectrum display in acqi (M)

cpmgt2 Set up parameters for CPMGT2 pulse sequence (M)

Description: Macro to set up a CPMGT2 (Carr-Purcell Meiboom-Gill  $T_2$ ) experiment.

See also: NMR Spectroscopy User Guide

Related:  $t_2$   $T_2$  exponential analysis (M)

cpos\_cvt Convert data set from a VXR-style system (M,U)

Syntax: (From UNIX) cpos cvt VXR file

convert(VXR\_file)

Description: Converts data stored on a VXR-style system (Gemini, VXR, or XL) to the

format used in VnmrJ software. cpos\_cvt writes the converted data in a subdirectory of the current working directory, using the original name of the

data set. The command convert loads the data from VXR\_file into the

current experiment and converts it to the new format.

Arguments: VXR file is the file name in the VXR-style format to be converted to the

VnmrJ style.

Related: convert data set from a VXR-style system (C,U)

decomp Decompose a VXR-style directory (C)

Retrieve FIDs (C)

# cptmp Copy experiment data into experiment subfile (M)

Syntax: cptmp<(file)>

Description: Copies the data (parameters, FID, and transformed spectrum) from the current

experiment into a subdirectory inside curexp+'/subexp'.

Arguments: file is the name of the subfile to receive the data. The default is to take the

name from the transmitter nucleus (if seqfil='s2pul') or to use the pulse

sequence name.

Examples: cptmp

cptmp('cosy')

Related: curexp Current experiment directory (P)

rttmp Retrieve experiment data from experiment subfile (M)

seqfil Pulse sequence name (P)

svtmp Move experiment data into experiment subfile (M)

# cpx Create pbox shape file (M)

Syntax: cpx<(ref pw90,ref pwr) > or cpx<('g')>

Description: Calls UNIX command Pbox, which generates the specified pulse shape or

decoupling/spin locking pattern, as defined by the shapelib/Pbox.inp

file.

Arguments: ref pw90 is the reference 90° pulse width

ref\_pwr is the reference power level.

'g' is a keyword that is required only when generating gradient shapes and if

the file type is not specified otherwise.

Examples: cpx

cpx('g')

cpx(pw90\*compH,tpwr)

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

#### cgexp Load experiment from protocol (M)

Applicability: Liquids

Description: Macro to load an experiment from a protocol.

Syntax: cqexp(experiment <, apptype>)

The first argument is the experiment name, and the second argument is the

apptype. If the apptype is not specified, the previous apptype is used.

Examples: cqexp('Proton', 'std1d')

Related: apptype Application type (P)

execpars Set up the exec parameters (M)

cqfindz0 Run an experiment to find the value of z0 (M)

Applicability: Liquids

Description: A macro to run a deuterium experiment to find the correct value of z0 for a

given solvent. It requires an entry in the probe file for the number of deuterium Hz per DAC. Run the appropriate probe calibration for lk Hz per DAC to set the value in the probe file. The macro may be accessed through the Find z0 button

available on several panels.

Related: solvent Lock solvent (P)

Z0 field position (P)

cqgmap Perform gradient shimming utility functions (M)

Applicability: Liquids

Description: Macro runs gradient shimming utility functions.

Related: gmapshim Run gradient autoshimming, set parameters, map shims (M)

cqinit Initialize liquids study queue (M)

Applicability: Liquids

Description: Initializes the liquids study queue.

Related: cqreset Reset study queue parameters (M)

 ${\tt sqfilemenu} \qquad \qquad {\tt Study} \ queue \ file \ menu \ commands \ (M)$ 

cqpars Create study queue parameters for liquids (M)

Applicability: Liquids

Description: A macro to create study queue parameters for the Walkup interface.

See also: VnmrJ Walkup

Related: fixpar Correct parameter characteristics in experiment (M)

cgplot Macro to perform generic 2D plot (M)

Applicability: Liquids

Description: A macro to perform generic 2D plotting, including 1D experiment traces.

Usually called by other macros, and not used from the command line.

 $Related: \quad \text{plot} \qquad \qquad Automatically \ plot \ spectra \ (M)$ 

plot 2D Plot results of 2D peak picking (C)

plt2Darg Plot 2D arguments (P)

cgprotocol Macro to create protocols (M)

Applicability: Liquids

Description: A macro to create protocols for liquids applications. Called by the Make

protocols dialogs in the Utilities menu.

cqreset Reset study queue parameters (M)

Applicability: Liquids

Description: Reset liquids study queue parameters. Usually called by other macros when

starting a new study.

Related: cqinit Initialize liquids study queue (M)

sqfilemenu Study queue file menu commands (M)

cqsavestudy Macro to save study queue parameters (M)

Applicability: Liquids

Description: A macro to save study parameters in the liquids study queue. Usually called by

other macros when starting a new study.

Related: studypar Study parameters (P)

xmsubmit Submit sample(s) to the study queue (M)
xmendq End a chained study queue (M)

cqwtmenu Macro to set weighting functions from a panel (M)

Applicability: Liquids, Imaging

Description: A macro to set weighting functions from a panel. It is used for both 1D and 2D

weighting parameters. Called by processing parameter panels.

cr Cursor position in directly detected dimension (P)

Description: Contains the current cursor position. The rl macro uses cr to set the reference

line.

See also: NMR Spectroscopy User Guide

Related: centersw Move cursor to center of spectrum (M)

Current time-domain cursor position (P)

Clear ref. line in directly detected dimension (M)

delta Difference of two frequency cursors (P)

Set reference line in directly detected dimension (M)

cr1 Cursor position in 1st indirectly detected dimension (P)

Description: Contains the current cursor position along the first indirectly detected

dimension. Analogous to the <u>cr</u> parameter except that <u>crl</u> applies to the first indirectly detected dimension of a multidimensional data set. The <u>rll</u> macro

uses cr1 to set the reference line along this dimension.

See also: NMR Spectroscopy User Guide

Related: centerswl Move cursor to center of spectrum in 1st indirect dimension (M)

Cursor position in directly detected dimension (P)
Cursor position in 2nd indirectly detected dimension (P)
rl1 Set ref. line in 1st indirectly detected dimension (M)

#### cr2 Cursor position in 2nd indirectly detected dimension (P)

Description: Contains the current cursor position along the second indirectly detected

dimension. Analogous to the  $\tt cr$  parameter except that  $\tt cr2$  applies to the second indirectly detected dimension of a multidimensional data set. The  $\tt rl2$ 

macro uses cr2 to set the reference line along this dimension.

See also: NMR Spectroscopy User Guide

Related: centersw2 Move cursor to center of spectrum in 2nd indirect dimension (M)

Cr Cursor position in directly detected dimension (P)
Cr1 Cursor position in 1st indirectly detected dimension (P)
r12 Set ref. line in 2nd indirectly detected dimension (M)

# crcom Create user macro without using text editor (M)

Syntax: crcom(file,actions)

Description: Creates a macro file in the user's macro library (maclib) with the contents

given in the actions argument.

Arguments: file is the file name of the user macro to be created. If a macro of the same

name already exists, the user is asked whether or not to overwrite it.

actions is a string containing the actions making up the user macro. The string cannot include a carriage return. If a single quote is needed within the string, it must be preceded by a backslash (see second example below).

Examples: crcom('plot','pl pscale pap page')

crcom('lds','load=\'y\' su load=\'n\'')

See also: User Programming

#### create Create new parameter in a parameter tree (C)

Syntax: create(parameter<,type<,tree>>)

Description: Creates a parameter in one of the parameter trees. A parameter tree is a UNIX

file containing the attributes of parameters as formatted text. Refer to the

command paramvi for a description of the file contents.

Arguments: parameter is the name of the parameter to be created.

type is the type of values in the parameter to be created and can be one of the following values (default is 'real'):

• 'real' is a value with no limits on range and can be positive or negative.

- 'string' is a value composed of characters. Entry of strings can be limited to selected words by enumerating the possible values with the command setenumeral. For example, the enumerated values of intmod are 'off', 'partial', and 'full'. Therefore, intmod can be set only to one of these three string values, such as intmod= 'full'.
- 'delay' is a value from 0 to 8190, in unit of seconds.
- 'frequency' is a positive real number value.
- 'flag', like 'string', is a value composed of characters. Entry of flags can be limited to selected characters by enumerating the possible values with the command setenumeral. For example, the enumerated values of dmm are 'c', 'f', 'g', 'm', 'p', 'r', 'u', 'w', and 'x'. Therefore, dmm can only be set to a combinations of these nine characters, such as dmm='ccw'. If enumerated values are not set, the 'string' and 'flag' types are identical.
- 'pulse' is a value from 0 to 8190, in units of μs.

• 'integer' is a value composed of integers (0,1,2,3,...).

tree is one of the following types of parameter trees (default is 'current'):

- 'current' contains parameters that are adjusted to set up an experiment.

  The parameters are from the file curpar in the current experiment.
- 'global' contains user-specific parameters from the file global in the vnmrsys directory of the present UNIX user.
- 'processed' contains parameters with which the data was obtained.

  These parameters are from the file procpar in the current experiment.
- 'systemglobal' contains instrument-specific parameters from the text file /vnmr/conpar. Most of these parameters are defined using the config program. All users have the same systemglobal tree. Note that conpar is not written out when you exit; the only time conpar is ever modified is by the config program. Thus, any changes you make to conpar using create (or destroy, setvalue, etc.) are not permanent. To permanently create a parameter in conpar, you must use a text editor to change /vnmr/conpar.

Examples: create('a')

create('b','string')

create('c','real','global')

See also: User Programming

Related: destroy Destroy a parameter (C)

display Display parameters and their attributes (C)

fread Read parameters from file and load them into a tree (C)

**fsave** Save parameters from a tree to a file (C)

paramvi Edit a parameter and its attributes using vi text editor (M)

prune Prune extra parameters from current tree (C)
setenumeral Set values of a string variable in a tree (C)
setgroup Set group of a parameter in a tree (C)
setprotect Set protection mode of a parameter (C)

# creategcomp Create qcomp parameter (M)

Applicability: Systems with Varian, Inc. Cold Probes

Description: Macro to create the gcomp parameter with the appropriate attributes. gcomp is

created as a flag parameter in the global tree.

# crf Current time-domain cursor position (P)

Description: Contains current time-domain cursor position. To create crf and the other FID

display parameters axisf, dotflag, vpf, vpfi, and deltaf (if the parameter set is older and lacks these parameters), enter addpar ('fid').

Values: Number, in seconds.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

crl1 Clear ref. line in 1st indirectly detected dimension (C)

deltaf Difference of two time cursors (P)

fidpar Add parameters for FID display in current experiment (M)

crl Clear reference line in directly detected dimension (M)

Description: Clears frequency referencing along the directly detected dimension by setting

the reference parameters rfl and rfp to zero. crl also resets the referencing

parameters refpos and reffrq.

See also: NMR Spectroscopy User Guide

Related: crl1 Clear ref. line in 1st indirectly detected dimension (C)

crl2 Clear ref. line in 2nd indirectly detected dimension (C)
rl Set ref. line in directly detected dimension (M)
reffrq Reference frequency of reference line (P)
refpos Position of reference frequency (P)

rfl Ref. peak position in directly detected dime

rfl Ref. peak position in directly detected dimension (P)
rfp Ref. peak frequency in directly detected dimension (P)

crl1 Clear reference line in 1st indirectly detected dimension (M)

Description: Clears frequency referencing along the first indirectly detected dimension by

setting the reference parameters rfl1 and rfp1 to zero. crl1 also resets the

referencing parameters refpos1 and reffrq1.

See also: NMR Spectroscopy User Guide

Related: crl Clear ref. line in directly detected dimension (C)

rl1 Set ref. line in 1st indirectly detected dimension (M)

reffrq1Ref. frequency of reference line in 1st indirect dimension (P)refpos1Position of reference frequency in 1st indirect dimension (P)rfl1Ref. peak position in 1st indirectly detected dimension (P)rfp1Ref. peak frequency in 1st indirectly detected dimension (P)

cr12 Clear reference line in 2nd indirectly detected dimension (M)

Description: Clears frequency referencing along the second indirectly detected dimension by

setting the reference parameters rfl2 and rfp2 to zero. crl2 also resets the

referencing parameters refpos2 and reffrq2.

See also: NMR Spectroscopy User Guide

Related: crl Clear ref. line in directly detected dimension (C)

Set ref. line in 2nd indirectly detected dimension (M)

reffrq2Ref. frequency of reference line in 2nd indirect dimension (P)refpos2Position of reference frequency in 2nd indirect dimension (P)rf12Ref. peak position in 2nd indirectly detected dimension (P)rfp2Ref. peak frequency in 2nd indirectly detected dimension (P)

crmode Current state of the cursors in df, ds, or dconi programs (P)

Description: Stores the current state (box mode or cursor mode) of cursors in the df, ds, or

dconi interactive display programs. crmode is mostly used by programmable menus to determine the status of the cursors. It is stored in the file vnmrsys/

global.

Values: 'b' signifies the box mode, 'c' signifies the cursor mode.

See also: User Programming

Related: dconi Interactive 2D data display (C)

df Display a single FID (C)
ds Display a spectrum (C)

#### crof2 Recalculate rof2 so that lp = 0 (M)

Syntax: crof2<(alfa)>

Description: Recalculates a new value for rof2 (receiver gating time following a pulse)

based upon the current rof2 and lp (first-order phase) values, so that lp is rendered approximately 0. For crof2 to work properly, a trial spectrum must be obtained and phased to pure absorption. This spectrum provides the current rof2 and lp values for crof2. The value of the alfa delay is left constant,

provided rof2 does not become less than 1 µs.

crof2 pertains to processing 2D data. Unless lp is approximately 0, fpmult

affects both the dc offset and the curvature of the spectrum.

Arguments: alfa specifies a value for the alfa delay before acquisition.

Related: alfa Set alfa delay before acquisition (P)

cfpmult Calculate first point multiplier for 2D experiments (P)

fpmult First point multiplier for np FID data (P)

First-order phase along directly detected dimension (P)

rof2 Receiver gating time following a pulse (P)

# cryo noisetestRun Cold Probe conditioning experiments (M)

Applicability: Systems with Varian, Inc. Cold Probes

Description: Runs the probe conditioning experiments and analyzes the noise using the cnd

macro. Measures the hydrogen-induced noise and provides an efficient remedy.

Values: NOBURN – waits the operator input period of time between tests.

No arguments – macro will prompt for a time in minutes.

# cryoclient Start the CryoBay Monitor program (M, U)

Applicability: Systems with Cold Probes and CryoBay Monitor software.

Description: Starts the CryoBay Monitor software in a separate window. This program is a

CORBA client that requires an active CORBA server running on the CryoBay

PC.

See also: Cryogenic Systems Installation and Operation

#### ct Completed transients (P)

Description: Stores a nonuser-enterable informational parameter that changes during the

course of an experiment to reflect the number of completed transients. During most experiments, an accurate transient counter is displayed in the acquisition

status window, updated every five seconds.

The value of ct is displayed in the acquisition parameter group by the dg command and is only updated when data processing occurs on the FID. In an experiment that is accumulating and not processed until the acquisition is

complete, ct always indicates 0 until the end of the acquisition.

See also: NMR Spectroscopy User Guide

Related: dg Display parameters of acquisition/processing group (C)

#### ctext Clear the text of the current experiment (C)

Description: Clears the text from the current experiment text file (a block of text that may be

used to describe the sample and experiment).

See also: NMR Spectroscopy User Guide

Related: atext Append string to the current experiment text (M)

Display text or set new text for current experiment (C)

curexp Current experiment directory (P)

Description: Contains the full UNIX path to the currently active experiment. This parameter

is useful when accessing text files generated by various commands (e.g.,

cat (curexp+'/fp.out')).

See also: NMR Spectroscopy User Guide

Related: systemdir VnmrJ system directory (P)

userdir VnmrJ user directory (P)

curscan Scan currently in progress (P)

Applicability: Systems with LC-NMR accessory.

Description: Keeps track of which "scan" is currently in progress. If curscan does not

exist, the parlc macro can create it.

See also: NMR Spectroscopy User Guide

Related: parlc Create LC-NMR parameters (M)

curwin Current window (P)

Description: An arrayed global parameter. The first value is the index of the selected window

pane in the graphics window. The second value is the number of window pane

rows. The third value is the number of columns.

See also: NMR Spectroscopy User Guide

Related: fontselect Open FontSelect window (C)

jwin Activate current window (M)
mapwin List of experiment numbers (P)
setgrid Activate selected window (M)
setwin Activate selected window (C)

cutoff Data truncation limit (P)

Description: Defines the distance above and below the current vertical position vp at which

spectra and integrals are truncated. By arraying cutoff to have two different values, the truncation limits above and below the current vertical position can be controlled independently (e.g., cutoff=50 truncates data at vp+50 mm and vp-50 mm, and cutoff=50,10 truncates data at vp+50 mm and

vp-10 mm). cutoff='n' disables the action of cutoff.

cutoff is not active during interactive spectral displays (i.e., for the ds command), but is active during non-interactive spectral displays and plots (for

the dss and pl commands).

Values: 'n', number in mm.

See also: NMR Spectroscopy User Guide

Related: ds Display a spectrum (C)

dss Display stacked spectra (C)

pl Plot spectra (C)

vp Vertical position of spectrum (P)

cyclenoe Set up parameters for CYCLENOE pulse sequence (M)

Applicability: Systems in which the observe channel is equipped with direct synthesis rf and a

linear amplifier.

Description: Sets up a difference NOE experiment.

cylbr24 Set up parameters for cycled BR24 pulse sequence (M)

Applicability: Systems with solids module.

Description: Sets up a BR24 sequence with quadrature detection and prepulse for solids

multiple-pulse line narrowing.

See also: User Guide: Solid-State NMR

Related: br24 Set up parameters for BR24 pulse sequence (M)

cylmrev Set up parameters for cycled MREV8 pulse sequence (M)

Applicability: Systems with a solids module.

Description: Sets up a MREV8 sequence with quadrature detection and prepulse for solids

multiple-pulse line narrowing.

See also: User Guide: Solid-State NMR

Related: mrev8 Set up parameters for MREV8 pulse sequence (M)

cz Clear integral reset points (C)

Syntax: cz<(frequency1,frequency2,...)>

Description: Removes currently defined integral reset points.

Arguments: frequency1, frequency2, . . . are reset points corresponding to

specified frequencies to be removed. The default is remove all reset points.

Examples: cz

cz(800,600,250,60)

See also: NMR Spectroscopy User Guide

Related dli Display listed integral values (C)

dlni Display listed normalized integral values (C)

nli Find normalized integral values (C)

Z Add integral reset point at the cursor position (C)

d0 Overhead delay between FIDs (P) d1 First delay (P) d2 Incremented delay in 1st indirectly detected dimension (P) Set up parameters for D2PUL pulse sequence (M) d2pul d3 Incremented delay for 2nd indirectly detected dimension (P) d4 Incremented delay for 3rd indirectly detected dimension (P) Store gradient calibration value in DOSY sequences (P) DAC to G da Display acquisition parameter arrays (C) Increment for t1 dependent first-order phase correction (P) daslp date Date (P) Display horizontal LC axis (M) daxis Set up parameters for Dbppste pulse sequence (M) Dbppste Set up parameters for Dbppsteinept pulse sequence (M) Dbppsteinept dbsetup Set up VnmrJ database (U) dbupdate Update the VnmrJ database (U) dc Calculate spectral drift correction (C) dc2d Apply drift correction to 2D spectra (C) Drift correction group (P) dcg dcon Display non interactive color intensity map (C) dconi Interactive 2D data display (C) dconi Control display selection for the dconi program (P) Display color intensity map without screen erase (C) dconn Remove dc offsets from FIDs in special cases (P) dcrmv Display data file in current experiment (C) ddf ddff Display FID file in current experiment (C) Display phase file in current experiment (C) ddfp ddif Synthesize and show DOSY plot (C) ddrcr Direct digital receiver coefficient ratio (P) ddrpm Set ddr precession mode (P) Set ddr time constant (P) ddrtc dds Default display (M) dds seqfil Sequence-specific default display (M) debug Trace order of macro and command execution (C) Select the type of decoupler asynchronous mode (P) decasynctype deccwarnings Control reporting of DECC warnings from PSG (P) Decompose a VXR-style directory (M) decomp def osfilt Default value of osfilt parameter (P) defaultdir Default directory for Files menu system (P) delcom Delete a user macro (M) delete Delete a file, parameter directory, or FID directory (C) delexp Delete an experiment (M) Removes nucleus entry to probe file (M) deletenucleus dels Delete spectra from  $T_1$  or  $T_2$  analysis (C)

delta Cursor difference in directly detected dimension (P)
delta1 Cursor difference in 1st indirectly detected dimension (P)
delta2 Cursor difference in 2nd indirectly detected dimension (P)

deltafDifference of two time-domain cursors (P)DeptSet up parameters for DEPT experiment (M)deptglSet up parameters for DEPTGL pulse sequence (M)

deptproc Process array of DEPT spectra (M)

**destroy** Destroy a parameter (C)

**destroygroup** Destroy parameters of a group in a tree (C)

df Display a single FID (C)

df2d Display FIDs of 2D experiment (C)

dfid Display a single FID (C)

dfmode Current state of display of imaginary part of a FID (P)

dfrq Transmitter frequency of first decoupler (P)
dfrq2 Transmitter frequency of second decoupler (P)
dfrq3 Transmitter frequency of third decoupler (P)
dfrq4 Transmitter frequency of fourth decoupler (P)

dfs Display stacked FIDs (C)

dfsa Display stacked FIDs automatically (C)

dfsan Display stacked FIDs automatically without screen erase (C)

dfsh Display stacked FIDs horizontally (C)

dfshn Display stacked FIDs horizontally without screen erase (C)

dfsn Display stacked FIDs without screen erase (C)

dfww Display FIDs in whitewash mode (C)

dg Display group of acquisition/processing parameters (C)

dg Control dg parameter group display (P)
dg1 Display group of display parameters (M)
dg1 Control dg1 parameter group display (P)

dg2 Display group of 3rd and 4th rf channel/3D parameters (M)

dg2 Control dg2 parameter group display (P)

dgaDisplay group of spin simulation parameters (M)DgcsteSLSet up parameters for DgcsteSL pulse sequence (M)DgcstecosySet up parameters for Dgcstecosy pulse sequence (M)DgcstehmqcSet up parameters for Dgcstehmqc pulse sequence (M)

dglc Display group of LC-NMR parameters (M)
dglc Control dglc parameter group display (P)

dglp Control dglp parameter group of linear prediction parameters (P)

dgs Display group of shims and automation parameters (M)

dgs Control dgs parameter group display (P)

dhp Decoupler high-power control with class C amplifier (P)

dialog Display a dialog box from a macro (C)

diffparams Report differences between two parameter sets (U)

diffshims Compare two sets of shims (M,U)

digfilt Write digitally filtered FIDs to another experiment (M)

dir List files in directory (C)

display Display parameters and their attributes (C)

dla Display spin simulation parameter arrays (M)

dlalong Long display of spin simulation parameter arrays (C)

dli Display list of integrals (C)

dlivast Produce text file and process wells (M)

dll Display listed line frequencies and intensities (C)

dlni Display list of normalized integrals (M)

dlp Decoupler low-power control with class C amplifier (P)

dm Decoupler mode for first decoupler (P)
dm2 Decoupler mode for second decoupler (P)
dm3 Decoupler mode for third decoupler (P)
dm4 Decoupler mode for fourth decoupler (P)

dmf Decoupler modulation frequency for first decoupler (P) dmf2 Decoupler modulation frequency for second decoupler (P) dmf3 Decoupler modulation frequency for third decoupler (P) dmf4 Decoupler modulation frequency for fourth decoupler (P) dmfadj Adjust tip-angle resolution time for first decoupler (M) dmf2adj Adjust tip-angle resolution time for second decoupler (M) dmf3adj Adjust tip-angle resolution time for third decoupler (M) dmf4adj Adjust tip-angle resolution time for fourth decoupler (M) dmg Data display mode in directly detected dimension (P) dmg1 Data display mode in 1st indirectly detected dimension (P) dmg2 Data display mode in 2nd indirectly detected dimension (P) dmgf Absolute-value display of FID data or spectrum in acqi (P) dmm Decoupler modulation mode for first decoupler (P)

dmm Decoupler modulation mode for first decoupler (P)
dmm2 Decoupler modulation mode for second decoupler (P)
dmm3 Decoupler modulation mode for third decoupler (P)
dmm4 Decoupler modulation mode for fourth decoupler (P)

dn Nucleus for first decoupler (P)
dn2 Nucleus for second decoupler (P)
dn3 Nucleus for third decoupler (P)
dn4 Nucleus for fourth decoupler (P)

dndfid Retrieve and process fid data from the locator (M)

dndjoinJoin a work space from the locator (M)dndparRetrieve a parameter set from the locator (M)dndshimsRetrieve a shimset set from the locator (M)dnodeDisplay list of valid limNET nodes (M,U)doautodialogStart a dialog window using def file (M)

dodialog Start a dialog window with dialoglib file (M)

dof Frequency offset for first decoupler (P)
dof2 Frequency offset for second decoupler (P)
dof3 Frequency offset for third decoupler (P)
dof4 Frequency offset for fourth decoupler (P)

Doneshot Set up parameters for Doneshot pulse sequence (M)

dopardialog Start a dialog with dialoglib/experiment def file (M)

do\_pcss Calculate proton chemical shifts spectrum (C)

dosy Process DOSY experiments (M)

dosy2d Apptype macro for dosy 2D experiments (M)

dosyfrq Larmor frequency of phase encoded nucleus in DOSY (P)

dosygamma Gyromagnetic constant of phase encoded nucleus in DOSY (P)
dosytimecubed Gyromagnetic constant of phase encoded nucleus in DOSY (P)

dot1Set up a  $T_1$  experiment (M)dotflagDisplay FID as connected dots (P)

downsamp Downsampling factor applied after digital filtering (P)

dp Double precision (P)

dpcon Display plotted contours (C)

dpconnDisplay plotted contours without screen erase (C)dpfDisplay peak frequencies over spectrum (C)dpirDisplay integral amplitudes below spectrum (C)

dpirn

Display normalized integral amplitudes below spectrum (M)

Display integral amplitudes below spectrum (M)

dpirn Display normalized integral amplitudes below spectrum (C)

dpl Default plot (M)

**dpl** seqfil Sequence-specific default plot (M)

dplaneDisplay a 3D plane (M)dprDefault process (M)

dpr\_seqfi1Sequence-specific default process (M)dprofileDisplay pulse excitation profile (M)dprojDisplay a 3D plane projection (M)dpsDisplay pulse sequence (C)

dpwr Power level for first decoupler with linear amplifier (P)
dpwr2 Power level for second decoupler with linear amplifier (P)
dpwr3 Power level for third decoupler with linear amplifier (P)

dpwr4 Power level for fourth decoupler amplifier (P)

dpwrfFirst decoupler fine power (P)dpwrf2Second decoupler fine power (P)dpwrf3Third decoupler fine power (P)

dpwrmFirst decoupler linear modulator power (P)dpwrm2Second decoupler linear modulator power (P)dpwrm3Third decoupler linear modulator power (P)

Dqcosy Convert the parameter to a DQCOSY experiment (M)
draw Draw line from current location to another location (C)

dresMeasure linewidth and digital resolution (C)dresTip-angle resolution for first decoupler (P)dres2Tip-angle resolution for second decoupler (P)dres3Tip-angle resolution for third decoupler (P)dres4Tip-angle resolution for fourth decoupler (P)

ds Display a spectrum (C)

ds2d Display 2D spectra in whitewash mode (C)

ds2dn Display 2D spectra in whitewash mode without screen erase (C)

dsnarray Report statistical signal-to-noise for Cold Probes (M)

dscaleDisplay scale below spectrum or FID (C)dscoefDigital filter coefficients for downsampling (P)dseqDecoupler sequence for first decoupler (P)dseq2Decoupler sequence for second decoupler (P)dseq3Decoupler sequence for third decoupler (P)

dseq4Decoupler sequence for fourth decoupler (P)dsfbDigital filter bandwidth for downsampling (P)dshapeDisplay pulse shape or modulation pattern (M)

**dshapef** Display last generated pulse shape (M)

dshapei Display pulse shape or modulation pattern interactively (M)

dshim Display a shim "method" string (M)

dslsfrq Bandpass filter offset for downsampling (P)

dsn Measure signal-to-noise (C)

dsnmax Calculate maximum signal-to-noise (M)

dspDisplay calculated spectrum (C)dspType of DSP for data acquisition (P)dsplanesDisplay a series of 3D planes (M)

**dsptype** Type of DSP (P)

dss Display stacked spectra (C)

dssa Display stacked spectra automatically (C)

dssan Display stacked spectra automatically without erasing (C)

dssh Display stacked spectra horizontally (C)

dsshn Display stacked spectra horizontally without erasing (C)

dssl Label a display of stacked spectra (M)

dssn Display stacked spectra without screen erase (C)

dsvast Display VAST data in a stacked 1D-NMR matrix format (M)

dsvast2dDisplay VAST data in a pseudo-2D format (M)dswwDisplay spectra in whitewash mode (C)dtextDisplay a text file in graphics window (M)

dtrig Delay to wait for another trigger or acquire a spectrum (P)

dutyc Duty cycle for homodecoupling (optional) (P)

# d0 Overhead delay between FIDs (P)

Description: Defir

Defines the extra overhead delay at the start of each FID or array element. Overhead times between increments and transients are deterministic, i.e., both known and constant. However, the time between increments (typically x) is longer than the time between transients (y, not including times that are actually part of the pulse sequence, such as d1). Some experiments may benefit if it is ensured that these two times are not only constant but equal. To ensure that the times are constant and equal, insert the time d0 at the start of each transient (before the pulse sequence actually starts); the actual delay is then y+d0. However, the overhead time may differ with different system configurations. To keep the d0 delay consistent across systems, set d0 greater than the overhead delay. The inter-FID delay x is then padded so that y+d0=x+(d0-(x-y)).

Currently, d0 only takes into account the extra delay at the start of each array element. It does not take into account the overhead delays at the start and end of each scan. It also does not take into account delays when arraying status statements, shims, or spinner speeds.

The d0 parameter does not exist in any parameter set and must be created by the user. To create d0, enter create ('d0', 'delay'). If d0 is nonexistent, do not insert a delay between transients.

Values: 'n', 'y', or 0 to the maximum delay time (in seconds).

If d0 = 'n', the software calculates the overhead time for an array element and then delays that length of time at the beginning of subsequent transients for every array element. The calculated value of d0 can be viewed by entering d0 = 'y' in the input window.

If d0 is set to a value, that value is the length of delay time at the beginning of subsequent transients for every array element. If the value is greater than the array overhead time, the array overhead time is padded to d0.

See also: User Programming

Related: create Create new parameter in parameter tree (C)

# d1 First delay (P)

Description: Length of the first delay in the standard two-pulse sequence and most other

pulse sequences. This delay is used to allow recovery of magnetization back to

equilibrium, if such a delay is desired.

Values: 0.1 µs to 8190 sec, smallest value possible is 0.1 µs, finest increment possible

is 12.5 ns.

See also: NMR Spectroscopy User Guide

Related: alfa Set alfa delay before acquisition (P)

d2 Incremented delay in 1st indirectly detected dimension (P)
d3 Incremented delay in 2nd indirectly detected dimension (P)
d4 Incremented delay in 3rd indirectly detected dimension (P)

pad Preacquisition delay (P)

#### d2 Incremented delay in 1st indirectly detected dimension (P)

Description: Length of the second delay in the standard two-pulse sequence. The delay is

controlled by the parameters ni and sw1 in a 2D experiment.

Values: 0.1 µs to 8190 sec, smallest value possible is 0.1 µs, finest increment possible

is 12.5 ns.

See also: NMR Spectroscopy User Guide

Related: d1 First delay (P)

ni Number of increments in 1st indirectly detected dimension (P)
sw1 Spectral width in 1st indirectly detected dimension (P)

# d2pul Set up parameters for D2PUL pulse sequence (M)

Description: Sets up a standard two-pulse sequence using the decoupler as transmitter.

See also: NMR Spectroscopy User Guide

Related: dhp Decoupler high power with class C amplifier (P)

dn Nucleus for the first decoupler (P)
dof Frequency offset for first decoupler (P)

dpwr Power level for first decoupler with linear amplifiers (P) s2pul Set up parameters for standard two-pulse sequence (M)

tn Nucleus for the observe transmitter (P)
tof Frequency offset for observe transmitter (P)

Power level of observe transmitter with linear amplifiers (P)

# d3 Incremented delay for 2nd indirectly detected dimension (P)

Description: Length of a delay controlled by the parameters ni2 and sw2 in a 3D

experiment. The d2 delay, which is controlled by ni and sw1, is incremented

through its entire implicit array first before d3 is incremented. To create parameters d3, ni2, phase2, and sw2 to acquire a 3D data set in the current experiment, enter addpar('3d').

Values: 0.1 μs to 8190 sec, smallest value possible is 0.1 μs, finest increment possible

is 12.5 ns.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

d1 First delay (P)

Number of increments in 2nd indirectly detected dimension (P)

par3d Create 3D acquisition, processing, display parameters (C)

phase 2 Phase selection for 3D acquisition (P)

sw2 Spectral width in 2nd indirectly detected dimension (P)

# d4 Incremented delay for 3rd indirectly detected dimension (P)

Description: Length of a delay controlled by the parameters ni3 and sw3 in a 4D

experiment. The d3 delay, which is controlled by ni2 and sw2, is incremented through its entire implicit array first before d4 is incremented. To create parameters d4, ni3, phase3, and sw3 to acquire a 4D data set in the current

experiment, enter addpar ('4d').

Values: 0.1 µs to 8190 sec, smallest value possible is 0.1 µs, finest increment possible

is 12.5 ns.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

d1 First delay (P)

Number of increments in 3rd indirectly detected dimension (P)

par4d Create 4D acquisition parameters (C)
phase3 Phase selection for 4D acquisition (P)

Spectral width in 3rd indirectly detected dimension (P)

#### DAC to G Store gradient calibration value in DOSY sequences (P)

Description: DAG\_to\_G is automatically set by the setup\_dosy macro by retrieving the

gradient strength from the probe calibration file if probe<>'' and storing it in DAC\_to\_G. If probe=''(i.e., the probe is not defined), then DAC\_to\_G

is set to the current value of the global parameter gcal

See also: NMR Spectroscopy User Guide.

Related: dosy Process DOSY experiments (M)

setup\_dosy Set up gradient levels for DOSY experiments (M)

setgcal Set the gradient calibration constant (M)

# da Display acquisition parameter arrays (C)

Syntax: da<(par1<,par2><,par3...>)>

Description: Displays arrayed acquisition parameters.

Arguments: par1, par2, par3, . . . are names of parameters to be displayed. The default

is to display all such parameters.

Examples: da

da('d2')

See also: NMR Spectroscopy User Guide

Related: dg Display parameters of acquisition/processing group (C)

daslp Increment for t1 dependent first-order phase correction (P)

Description: Causes "shearing" of f<sub>1</sub> traces of a 2D dataset and is used to rotate the narrow

projection of some solids correlations into the  $f_1$  dimension. Several solids experiments for Dynamic Angle Spinning (DAS) and a triple-quantum filtered 2D MAS experiment require the use of daslp. (Note that the command rotate shears two traces and is inapplicable for these experiments.)

When created, the value of lp for each increment of a 2D experiment is incremented by the value of daslp after the first Fourier transformation. The incremented phase correction is applied to the interferogram created from the coefficient table by ftld, ftld, wftld and wftld, when coefficients are present. daslp is also used with ftlda, ftlda, wftlda and wftlda.

Values: Real values, typically similar in size to the value of parameter lp.

See also: NMR Spectroscopy User Guide

Related: ftld Fourier transform along f2 dimension (C)

ftlda Fourier transform phase-sensitive data (M)

ft2d Fourier transform 2D data (C)

Fourier transform phase-sensitive data (M)

1p First-order phase in directly detected dimension (P)

rotate Rotate 2D data (C)

wftld Weight and Fourier transform f2 for 2D data (C)
wftlda Weight and Fourier transform phase-sensitive data (M)

wft2d Weight and Fourier transform 2D data (C)

wft2da Weight and Fourier transform phase-sensitive data (M)

date Date (P)

Description: An informational parameter taken from the UNIX-level calendar (which is set

by the UNIX system operator only and cannot be entered by the user).

Whenever data are acquired, the date is copied from UNIX and written into the acquisition parameters, thus maintaining a record of the date of acquisition.

See also: NMR Spectroscopy User Guide

daxis Display horizontal LC axis (M)

Applicability: Systems with LC-NMR accessory.

Syntax: daxis(time, major tic, minor tic)

Description: Displays a horizontal LC axis. Horizontal axes are assumed to be used with "LC

plots" of an entire LC run and are labeled accordingly.

Arguments: time is the time scale, in minutes (decimal values are fine), of the axis.

major\_tic is spacing, in minutes (decimal values are fine), of major tics.
minor\_tic is spacing, in minutes (decimal values are fine), of minor tics.

See also: NMR Spectroscopy User Guide

Related: paxis Display horizontal LC axis (M)

Dbppste Set up parameters for Dbppste pulse sequence (M)

Description: Converts a parameter set to Dbppste experiment; replaces the macro bppste.

See also: NMR Spectroscopy User Guide

Related: dosy Process DOSY experiments (M)

fiddle Perform reference deconvolution (M)

setup dosy Set up gradient levels for DOSY experiments (M)

#### Dbppsteinept Set up parameters for Dbppsteinept pulse sequence (M)

Description: Converts a parameter set to Dbppsteinept experiment.

See also: NMR Spectroscopy User Guide

Related: dosy Process DOSY experiments (M)

fiddle Perform reference deconvolution (M)

setup dosy Set up gradient levels for DOSY experiments (M)

#### dbsetup Set up VnmrJ database (U)

Syntax: dbsetup <vnmr adm|remove|standard|imaging>

dbsetup vnmr adm <remove|standard|imaging>

As Root:

dbsetup vnmr\_adm VnmrJ\_Home\_dir <standard|imaging>

Arguments: vnmr adm is the login ID of the VnmrJ system administrator.

remove only removes the data-database; does not recreate a database.

standard creates the database for standard use.

imaging creates the database for imaging spectroscopy.

Description: The UNIX script dbsetup is used during the installation of VnmrJ software

and can only be run by the VnmrJ administrator (vnmr\_adm) or the UNIX administrator (root). Normally it is never used again. dbsetup creates and deletes the data-database in /vnmr/pgsql/data and the user information in

/vnmr/adm/users.

When run as root at least two arguments must be supplied, the login ID of the  $\,$ 

VnmrJ administrator and the VnmrJ home directory. When run as root

dbsetup will delete and recreate the data-database in /vnmr/pgsql/data for all users in /vnmr/adm/users. If no user list exists yet, the list is created with the VnmrJ administrator as the only user. The mode can be specified with the third argument as 'standard' or 'imaging'; if neither is specified the mode is taken from the global file of the VnmrJ administrator. It defaults to standard. The VnmrJ administrator does not need to supply any of the

arguments.

Note that additional users are created using vnmrj adm.

Examples: dbsetup

dbsetup vnmr1

See also: NMR Spectroscopy User Guide

VnmrJ Imaging NMR

VnmrJ Installation and Administration

#### dbupdate Update the VnmrJ database (U)

Applicability: Systems with the VnmrJ software.

Syntax: dbsupdate stop|once [slow ms]|forever [slow ms]

Arguments: slow ms is an optional argument used to slow down the database update so as

not to use all of the available CPU time. slow ms=0 is full speed.

slow ms=1000 uses about 2-5% of the CPU.

The dbupdate command is runs under nice so that any other process will be able to take the CPU away from this update anyway. The default slow ms for

forever is 1000. The default slow ms for once is 0.

Description: A UNIX command to start and stop a program to update the VnmrJ database

used by the Locator. This command might be needed at a data station to view newly acquired data. The database at the spectrometer will automatically be

updated.

#### dc Calculate spectral drift correction (C)

Description: Turns on a linear baseline correction. The beginning and end of the straight line

to be used for baseline correction are determined from the display parameters sp and wp. dc applies this correction to the spectrum and stores the definition of the straight line in the parameters lvl (level) and tlt (tilt). The correction

is turned off by the command cdc.

Care must be taken to ensure that a resonance does not appear too close to either end of the spectrum, or dc can produce the opposite effect from that intended;

namely, it induces a sloping baseline where none was present!

See also: NMR Spectroscopy User Guide

Related: bc 1D and 2D baseline correction (C)

cdc Cancel drift correction (C)
dc Drift correction group (P)

lvl Zero-order baseline correction (P)

sp Start of plot (P)

tlt First-order baseline correction (P)

wp Width of plot (P)

# dc2d Apply drift correction to 2D spectra (C)

Syntax: dc2d('f1'|'f2')

Description: Computes a drift correction and applies it to each individual trace.

Arguments: 'f1' is a keyword to apply drift correction in the  $f_1$  axis direction.

'f2' is a keyword to apply drift correction in the f2 axis direction.

Examples: dc2d('f1')

dc2d('f2')

See also: NMR Spectroscopy User Guide

Related: axis Axis label for displays and plots (P)

bc 1D and 2D baseline correction (C)

# dcg Drift correction group (P)

Description: Contains the results of the dc or cdc command. This parameter cannot be set

in the usual way but it can be queried by entering dcg? to determine whether

drift correction is active.

Values: 'dc' indicates drift correction is active.

'cdc' indicates drift correction is inactive.

See also: NMR Spectroscopy User Guide

Related: cdc Cancel drift correction (C)

dc Calculate spectral drift correction (C)

#### dcon Display noninteractive color intensity map (C)

Syntax: dcon<(options)>

Description: Produces a "contour plot," actually a color intensity map, in the graphics

window. The parameters sp and wp, sp1 and wp1, and sp2 and wp2 control which portion of the spectrum is displayed. The parameters sf and wf, sf1 and wf1, and sf2 and wf2 control which portion of time-domain data (FIDs and interferograms) is displayed. The parameter transposed solvets which

and interferograms) is displayed. The parameter trace selects which dimension is displayed along the horizontal axis. The parameters sc, wc, sc2,

and wc2 control where on the screen the display occurs. The parameter th is

active as a threshold to black out all contours whose intensity is below th. That is, if th=7, the colors 1 to 6 are not used for the display. The parameter vs controls the vertical scale of the spectrum.

dcon displays either absolute-value mode or phase-sensitive 2D data. In av mode, data are shown in 15 different colors (starting with black), with each color representing a factor of two in intensity (a single color is used on monochrome screens). In the ph mode, the normal display of colors ranges from –6 to +6, each representing a factor of two in intensity, with the color black representing intensity 0 in the center.

Arguments: options can be any of the following:

- 'linear' is a keyword to use linear instead of logarithmic increments.
- 'phcolor' is a keyword to use a phased color set with positive and negative peaks.
- 'avcolor' is a keyword to use an absolute-value color set with positive peaks. Negative contours only *cannot* be displayed, but if the data can be rephased, 180° added to rpl, and dcon('avcolor') entered again, the same thing is accomplished by inverting the phase of all peaks. Alternatively, dpcon can display negative peaks only.
- 'gray' is a keyword to use a gray scale color set.
- 'noaxis' is a keyword to omit the display outline and any horizontal or vertical axis.
- 'plot' causes the dcon display to be sent to the plotter instead of being drawn on the graphics window.

#### Examples: dcon

dcon('gray')

dcon('linear','phcolor','plot')

See also: NMR Spectroscopy User Guide

Related: dconi Interactive 2D data display (C)

dconi Control display selection for the dconi program (P)
dconn Display color intensity map without screen erase (C)

dpcon Display plotted contours (C)

imageprint Plot noninteractive gray scale image (M)

Start of chart (P)

Start of chart in second direction (P)

sf Start of FID (P) sp Start of plot (P)

Start of plot in 1st indirectly detected dimension (P)

Start of plot in 2nd indirectly detected dimension (P)

th Threshold (P)

trace Mode for *n*-dimensional data display (P)

wc Width of chart (P)

wc2 Width of chart in second direction (P)

wf Width of FID (P)
wp Width of plot (P)

wp1 Width of plot in 1st indirectly detected dimension (P)
wp2 Width of plot in 2nd indirectly detected dimension (P)

# dconi Interactive 2D data display (C)

Syntax: dconi<(options)>

Description: Opens a 2D data display that can be interactively adjusted. The dconi program

can accommodate any data set that can be displayed by dcon, dpcon, and

ds2d, including 2D FIDs, interferograms, 2D spectra, planes from 3D data sets, and images. These data sets are generated by the commands df2d, ft1d, ft2d, and ft3d.

Arguments: options can be any of the following (note that the dconi parameter is also available to control the dconi program display):

- 'dcon' is a keyword to display a color intensity map; this is the default mode, but 'dcon' is provided for compatibility with certain macros. If 'dcon' is the first argument, it can be followed by any of the keywords 'linear', 'phcolor', 'avcolor', 'gray', and 'noaxis'; all of these keywords have the same meaning as when used with dcon.
- 'dpcon' is a keyword to display a true contour plot. If 'dpcon' is the first argument, it can be followed by any of the keywords 'pos', 'neg', and 'noaxis', and then followed by values for levels and spacing. All of these options have the same meaning as when used with dpcon.
- 'ds2d' is a keyword to display a stacked plot in whitewash mode (after the first spectra, each spectra is blanked out in regions in which it is behind an earlier spectra). If 'ds2d' is the first argument, it can be followed by any of the keywords 'nobase', 'fill', 'fillnb', and 'noaxis'. All of these keywords have the same meaning as used with ds2d.
- 'again' is a keyword to make dconi identify which display mode is currently being used and redraw the screen in that mode.
- 'restart' is a keyword to activate dconi without redrawing the 2D data set. This action causes dconi to make sure that 2D data is already displayed.
- 'toggle' is a keyword to toggle between the cursor and box modes.
- 'trace' is a keyword to draw a trace above the spectrum.
- 'expand' is a keyword to toggle between the expand and full views of the spectrum.
- 'plot' is a keyword to plot a projection or a trace.
- 'hproj max' is a keyword to do a horizontal projection of the maximum
- 'hproj sum' is a keyword to do a horizontal projection of the sum of all traces.
- 'vproj max' is a keyword to do a vertical projection of the maximum
- 'vproj sum' is a keyword to do a vertical projection of the sum of all traces.

Cursor difference in 1st indirectly detected dimension (P)

Examples: dconi

```
dconi('dcon','gray','linear')
dconi('dpcon')
```

See also: NMR Spectroscopy User Guide

Related: boxes Draw boxes selected by the mark command (C) Current state of cursors in dfid, ds, or dconi (P) crmode dcon Display noninteractive color intensity map (C) dconi Control display selection for the dconi program (P) dconn Display color intensity map without screen erase (C)

> Display FIDs of 2D experiment (C) df2d dpcon Display plotted contours (C)

Display 2D spectra in whitewash mode (C) ds2d

130

delta1

ft1d	Fourier transform along f <sub>2</sub> dimension (C)
ft2d	Fourier transform 2D data (C)
ft3d	Perform a 3D Fourier transform on a 3D FID data set (M,U)
imconi	Display 2D data in interactive gray-scale mode (M)
is	Integral scale (P)
112d	Automatic and interactive 2D peak picking (C)
proj	Project 2D data (C)
sf	Start of FID (P)
sp	Start of plot (P)
sp1	Start of plot in 1st indirectly detected dimension (P)
th	Threshold (P)
vs2d	Vertical scale for 2D displays (P)
vsadj	Automatic vertical scale adjustment (M)
wf	Width of FID (P)
wp	Width of plot (P)
wp1	Width of plot in 1st indirectly detected dimension (P)

# dconi Control display selection for the dconi program (P)

# Description:

Controls the selection of the 2D display that follows entering the dconi command. Because dconi is implicitly executed by ft2d, the dconi parameter also controls the display that follows the ft2d or wft2d command.

dconi can be a string parameter in the "current" parameter set. Its syntax is similar to an argument string passed to the dconi program. For example, if dconi = 'dpcon, pos, 12, 1.2', the dconi command displays twelve positive contours with dpcon, using a spacing of 1.2. The first component of the dconi string must be the name of the display program, such as dcon, dconn, dpcon, dpconn, ds2d, or ds2dn. Subsequent components of the string are arguments appropriate for that display program. Because the entire dconi parameter is a string, single quotes around words are not necessary and mixing words and numbers is not a problem, as the example above shows.

If the dconi parameter does not exist or is set to the null string (''), the dconi program uses its normal default. If the dconi parameter is set to a string (e.g., dconi='dcon, gray, linear' for image display), and arguments are supplied to the dconi program, (e.g., dconi('dpcon')), the supplied arguments to the command take precedence. In the case of the examples above, a contour map, not an image, is displayed.

If the dconi parameter does not exist in the current experiment, it can be created by the commands create ('dconi', 'string')
setgroup ('dconi', 'display')

Values: ' ' (two single quotes) indicates that this parameter is ignored.

String 'display\_program' selects the named program for 2D displays. String 'display\_program, option1, option2' selects the named program for 2D displays with options appropriate to the program.

Examples: dconi='dpcon' selects contour drawing rather than default color map

dconi='dcon, gray, linear' selects image display mode.

See also: NMR Spectroscopy User Guide; VnmrJ Imaging NMR

Related: dcon Display noninteractive color intensity map (C)

dconi Interactive 2D data display (C)

dconn Display color intensity map without screen erase (C)

dpcon Display plotted contours (C)

dpconn
Display plotted contours without screen erase (C)
Display 2D spectra in whitewash mode (C)

ds2dn Display 2D spectra in whitewash mode without screen erase (C)

ft2d Fourier transform 2D data (C)

imconi Display 2D data in interactive gray-scale mode (M) wft2d Weight and Fourier transform 2D data (C)

dconn Display color intensity map without screen erase (C)

Syntax: dconn<(options)>

Description: Produces a "contour plot," actually a color intensity map, on the screen the same

as the dcon command, but without erasing the screen before starting the plot.

The options available are the same as the dcon command.

See also: NMR Spectroscopy User Guide

Related: dcon Display noninteractive color intensity map (C)

dconi Control display selection for the dconi program (P)

dcrmv Remove dc offsets from FIDs in special cases (P)

Description: If dcrmv exists and is set to 'y', hardware information is used to remove the

dc offset from the FID providing ct=1. This only works on systems with sw less than 100 kHz. If this feature is desired for a particular experiment, create dcrmv in that experiment by entering create ('dcrmv', 'string')

setgroup('dcrmv','processing') dcrmv='y'

To create image parameters dcrmv, grayctr and graysl in the current

experiment, enter addpar ('image').

See also: NMR Spectroscopy User Guide; VnmrJ Imaging NMR

Related: addpar Add selected parameters to the current experiment (M)

**create** Create new parameter in a parameter tree (C)

Ct Completed transients (P)

dc Calculate spectral drift correction (C)
setgroup Set group of a variable in a tree (C)

ddf Display data file in current experiment (C)

Syntax: ddf<(block number,trace number,first number)>

Description: Displays the file header of the data file in the current experiment. If entered with

arguments, it also displays a block header and part of the data file of that block.

Arguments: block\_number is the block number. Default is 1.

trace number is the trace number within the block. Default is 1.

first number is the first data element number within the trace. Default is 1.

See also: User Programming

Related: ddff Display FID file in current experiment (C)

ddfp Display phase file in current experiment (C)

ddff Display FID file in current experiment (C)

Syntax: ddff<(block number,trace number,first number)>

Description: Displays the file header of the FID file in the current experiment. If entered with

arguments, it also displays a block header and part of the FID data of the block.

Arguments: block number is the block number. Default is 1.

trace number is the trace number within the block. Default is 1.

first number is the first data element number within the trace. Default is 1.

See also: *User Programming* 

Related: Display data file in current experiment (C)

> ddfp Display phase file in current experiment (C)

Display phase file in current experiment (C) ddfp

Syntax: ddfp<(block number, trace number, first number) >

Description: Displays the file header of the phase file in the current experiment. With

arguments, it also display a block header and part of the phase file data of that

block.

Arguments: block number is the block number. Default is 1.

trace number is the trace number within the block. Default is 1.

first number is the first data element number within the trace. Default is 1.

See also: User Programming

Related: ddf Display data file in current experiment (C)

> ddff Display FID file in current experiment (C)

#### ddif Synthesize and show DOSY plot (C)

Syntax: ddif(<option>,lowerlimit,upperlimit)

Description: Synthesizes a 2D spectrum from 1D spectra using the information produced by

> the dosy macro. ddif takes the 1D spectrum and a table of diffusion data stored in the file diffusion\_display.inp in the current experiment and synthesizes a 2D DOSY spectrum. It is normally run by dosy, but can be directly run, for example, to recalculate a 2D DOSY spectrum with different digitization.

Arguments: option is either 'i' or 'c'.

'i' is for a display in which the 2D peak volume is proportional to 1D peak

height.

'c' is for a display in which the 2D peak height equals the 1D.

lowerlimit is the lower diffusion limit (in units of  $10^{-10}$  m<sup>2</sup>/s).

upperlimit is the upper diffusion limit (in units of  $10^{-10}$  m<sup>2</sup>/s).

If arguments are not supplied, ddif defaults to showing the full range of diffusion coefficients in the file diffusion\_display.inp in the current experiment. Make sure that the first increment of the DOSY data set has been transformed with the desired fn2D before using ddif. Digitization of the resultant spectrum is determined by fn2D in the spectral (F2) domain and fn1 in the diffusion (F1) domain. Make sure that the product fn2D\*fn1 is not too large, or memory and processing time problems might result. Typical values are fn2D=16384 (max: 64k) and fn1=512. After dosy or ddif, 1D data is overwritten by the 2D (the dosy macro keeps a copy of the 1D data, which can be retrieved with the command undosy). Similarly, after a DOSY spectrum has been calculated, it can be retrieved with the command redosy.

See also: NMR Spectroscopy User Guide

Related: dosy Process DOSY experiments (M)

> fn2D Fourier number to build up 2D DOSY display in frequency domain (P) Restore the previous 2D DOSY display from the subexperiment (M) redosy

undosy Restore original 1D NMR data from the subexperiment (M)

#### ddrcr Direct digital receiver coefficient ratio (P)

Applicability: VNMRS systems and 400 - MR systems

Syntax: ddrcr=<value>

Description: Sets the filter sharpness or filter coefficient ratio. The default value of 75 is used

if the parameter does not exist.

Examples: create('ddrce','integer')

setlimit('ddrcr',1000,2,1)

ddrcr=300

Values: Integer values between 2 and 1000

See also: NMR Spectroscopy User Guide and VnmrJ User Programming.

Related: sw Spectral width in directly detected dimension (P)

# ddrpm Set ddr precession mode (P)

Applicability: VNMRS systems

Syntax: ddrpm=<'mode'>

Values: mode can be either of following:

Mode Description

p Pulse — default if no argument is supplied.

The value is calculated as follows if ddrpm does not exist or ddrpm='p':

 $ddrtc = alfa + rof2 + 2 * pw[1] / \pi$ 

e Echo — The value is calculated as follows: ddrtc = alfa.

See also: VnmrJ User Programming

Related: setrc Set frequency referencing based upon lock signal shift (M)

ddrtc Set ddr precession mode (P)

# ddrtc Set ddr time constant (P)

Applicability: VNMRS systems

Syntax: ddrtc=<'value'>

Description: The value of ddrtc is set in the setrc macro and is determined by the ddrpm

parameter.

A value of ddrtc = alfa is used by psg if the ddrtc parameter does not

exist.

Values: value 0 to 1000 µsec.

See also: VnmrJ User Programming

Related: setrc Set frequency referencing based upon lock signal shift (M)

setlp0 Set parameters for zero linear phase (M)

ddrpm Set ddr precession mode (P)

#### dds Default display (M)

 $Description: \ Looks \ for \ sequence-specific \ default \ display \ macro \ (\verb"dds_seqfil") \ and$ 

executes if one is found. If not, the dds macro displays 1D, 2D, or array

spectrum as the case may be.

Related: dds seqfil Sequence-specific default display (M)

dpl Default plot (M)
dpr Default process (M)

# dds segfil Sequence-specific default display (M)

Description: Sequence-specific default display. These macros are called by the dds macro.

Examples: dds\_NOESY1D

dds\_TOCSY1D

Related: dds Default display (M)

dpl Default plot (M)
dpr Default process (M)

#### debug Trace order of macro and command execution (C)

Syntax: debug('c'|'C')

Description: Controls VnmrJ command and macro tracing. When turned on, debug displays

a list of each command and macro in the shell tool from which VnmrJ was started. If VnmrJ is started when the user logs in, or if it was started from a drop-down menu or the CDE tool, the output goes to a Console window. If no Console window is present, the output goes into a file in the /var/tmp directory. This last option is not recommended. Nesting of the calls is indicated by indentation of the output. This feature is primarily a debugging tool for

MAGICAL programming.

To associate the debut('c') output with a particular terminal, enter tty. The system responds with /dev/pts/yyy, where yyy is a numerical value. On

the VnmrJ command line, enter jFunc(55, '/dev/pts/yyy'),

substituting the numerical value for the yyy.

Arguments: 'c' is a keyword to turn on command and macro tracing.

'C' is a keyword to turn off command and macro tracing.

Examples: debug('c')

debug('C')

See also: User Programming

### decasynctype Select the type of decoupler asynchronous mode (P)

Applicability: VNMR systems, 400 MR

Syntax: decasynctype=<value>

Description: Optional parameter to select the type of decoupler asynchronous mode. The

default type is p (progressive mode), which simulates a free running decoupler with respect to the pulse sequence timing. The other available options are b (bit reversed mode) and r (random mode). The b mode (bit-reversed mode) may be especially appropriate for reducing unwanted decoupling side-band intensity, when the number of transients is small. In the r mode (random mode) the starting decoupling stage is randomized. The decasynctype setting affects the decoupling on all the rf channels (dm, dm2, dm3 etc.). This parameter is optional and if it does not exist the decoupler asynchronous mode is set top

(progressive mode).

Values: 'p'— progressive mode, mode used by Inova systems

'b' — bit-reversed mode

'r' — random mode

Examples: decasynctype='p' for progressive mode.

Related: dm Decoupler mode for first decoupler (P)

dm2 Decoupler mode for second decoupler (P)
dm3 Decoupler mode for third decoupler (P)

# deccwarnings Control reporting of DECC warnings from PSG (P)

Applicability: Systems with DECC (Digital Eddy Current Compensation) boards for gradient

compensation.

Description: A global parameter that controls whether PSG will warn the user when the ECC

corrections are large enough that they could exceed the capabilities of the DECC board. By default, this parameter does not exist, and a warning is printed whenever an experiment is started if the ECC amplitudes are possibly too large. The warning does indicate a definite be a problem, only that not enough ECC drive capability is available to compensate for an instantaneous gradient swing from minus the maximum gradient strength to the maximum positive gradient.

To disable the warnings, create this global string parameter and set it to 'n'.

Values: 'n' or 'N' to suppress warnings. If the value starts with any other character, the

normal warnings are printed.

# decomp Decompose a VXR-style directory (M)

Syntax: decomp<(VXR\_file)>

Description: Takes a library, as loaded from a VXR-style system (VXR, XL, or Gemini), and

extracts each entry into a separate UNIX file. The file can be obtained from a magnetic tape or over limNET. decomp creates a UNIX subdirectory in the current working directory and uses that to write each entry as a UNIX file. The

name of the UNIX subdirectory is derived from the library name.

Arguments: VXR file is the name of the original file. It must have an extension in the

form .NNN, where NNN is the number of entries in the original library. A limit

of 432 entries is imposed.

See also: NMR Spectroscopy User Guide

Related: convert data set from a VXR-style system (C,U)

# def osfilt Default value of osfilt parameter (P)

Description: A global parameter that establishes the default type of digital filter,

Analog*Plus*<sup>TM</sup> or brickwall, when DSP is configured. The *actual* filter used in any experiment is set by the local parameter <code>osfilt</code>. Usually, <code>def\_osfilt</code> is set to the value for normal use, and then <code>osfilt</code> is changed within a given

experiment if different filter characteristics are desired.

Values: 'a' or 'A' for the Analog Plus digital filter. This filter is flatter in the passband

and drops off somewhat more sharply than analog filters.

'b' or 'B' for the brickwall digital filter. This filter is extremely flat across the passband and drops off sharply on the edge; however, the enhanced filtering

comes at the expense of somewhat reduced baseline performance.

See also: NMR Spectroscopy User Guide

Related: dsp Type of DSP for data acquisition (P)

osfilt Oversampling filter for real-time DSP (P)

# defaultdir Default directory for Files menu system (P)

Description: Stores the name to the default directory for use with the Directory Menu in the

Files menu system. Initial value for defaultdir is the home or login

directory of the user. Selecting the Default button in the Directory Menu sets the current directory to the value of defaultdir. The opposite action, setting the value of defaultdir to the current directory, occurs when the Set Default button in the Directory Menu is selected. If the entry for a directory is marked and the Set Default button is selected, the directory marked becomes the new value of defaultdir.

See also: NMR Spectroscopy User Guide

delcom Delete a user macro (M)

Syntax: delcom(file)

Description: Deletes a macro file in a user's macro library (maclib). Note that delcom

will not delete a macro in the VnmrJ system macro library.

Arguments: file is the file name of the user's macro to be deleted.

Examples: delcom('lds')
See also: User Programming

Related: crcom Create user macro without using a text editor (C)

macrorm Remove a user macro (C)

delete Delete a file, parameter directory, or FID directory (C)

Syntax: delete(file1<,file2,...>)

Description: Delete files and directories in a somewhat safer manner than the rm command.

Using rm is not recommended in VnmrJ because rm allows wildcard characters (\* and ?) in the file description and recursive file deletion with the -r option. The delete command does not allow wildcard characters or the -r option,

but you can still use the delete command to delete a file as well as

remove .fid and .par directories, normally the only directories that need to be removed (experiment directories are deleted with the <a href="mailto:delexp">delexp</a> macro).

Arguments: file1, file2, ... are the names of one or more files or directories to be

deleted. When the delete command is entered, it first searches for file1. If it finds that file and it is not a directory, file1 is deleted. If file1 is not found, .fid is appended to the file name and delete searches for the file in that .fid directory. If the file is found, it is removed; otherwise, .par is appended to the file name and delete searches for the file in that .par directory. If the file is found, it is removed; otherwise, the command takes no action and continues to the next file name. The process is repeated for each file

name given as an argument.

Examples: delete('/home/vnmr1/memo')

delete('/vnmr/fidlib/fid1d')

See also: NMR Spectroscopy User Guide

Related: delexp Delete an experiment (M)

rm Delete file (C)
rmdir Remove directory (C)

delexp Delete an experiment (M)

Syntax: delexp(experiment\_number)

Description: Deletes an experiment.

Arguments: experiment number is the number (from 2 through 9999) of the

experiment to be deleted (experiment 1 cannot be deleted). delexp also

deletes the corresponding jexpXXX macro if necessary.

Examples: delexp(321)

See also: NMR Spectroscopy User Guide

Related: cexp Create an experiment (M)

jexp Join existing experiment (C)

# deletenucleus Removes nucleus entry from current probe file (M)

Applicability: ALL

Description: All lines for the specified nucleus are removed from the current probe file. The

argument should correspond to an entry in the probe file.

Syntax: deletenucleus('nucleus')

Arguments: nucleus — name followed by atomic number, e.g. C13 not 13C.

Examples: deletenucleus('Si29')

Related: addnucleus Adds nucleus entry to probe file (M)

addprobe Create new probe directory and probe file (M)

# dels Delete spectra from $T_1$ or $T_2$ analysis (C)

Syntax: dels(index1<,index2,...>)

Description: Deletes the spectra selected from the file fp.out (the output file of fp) used

by the t1 or t2 analysis. Spectra may be restored by rerunning fp.

Arguments: index1, index2, . . . are the indexes of the spectra to be deleted.

Examples: dels(7)

dels(2,5)

See also: NMR Spectroscopy User Guide

Related: dll Display listed line frequencies and intensities (C)

fp Find peak heights or phases (C)

get11 Get frequency and intensity of a line (C)

t1  $T_1$  exponential analysis (M) t2  $T_2$  exponential analysis (M)

#### delta Cursor difference in directly detected dimension (P)

Description: Difference between two frequency cursors along the directly detected

dimension. The value is changed by moving the right cursor, relative to the left,

in the ds or dconi display.

Values: Positive number, in Hz.

See also: NMR Spectroscopy User Guide

Related: dconi Interactive 2D data display (C)

delta1 Cursor difference in 1st indirectly detected dimension (P)
delta2 Cursor difference in 2nd indirectly detected dimension (P)

ds Display a spectrum (C)

split Split difference between two cursors (M)

delta1 Cursor difference in 1st indirectly detected dimension (P)

Description: Difference of two frequency cursors along the first indirectly detected

dimension. Analogous to the delta parameter except that delta1 applies to

the first indirectly detected dimension of a multidimensional data set.

Values: Positive number, in Hz.

See also: NMR Spectroscopy User Guide

Related: delta Cursor difference in directly detected dimension (P)

delta2 Cursor difference in 2nd indirectly detected dimension (P)

Description: Difference of two frequency cursors along the second indirectly detected

dimension. Analogous to the delta parameter except that delta2 applies to the second indirectly detected dimension of a multidimensional data set.

Values: Positive number, in Hz.

See also: NMR Spectroscopy User Guide

Related: delta Cursor difference in directly detected dimension (P)

deltaf Difference of two time-domain cursors (P)

Description: Difference between the two time-domain cursors of the df (or dfid) display.

To create this parameter and the other FID display parameters axisf,

dotflag, vpf, vpfi, and crf (if the parameter set is older and lacks these

parameters), enter addpar('fid').

Values: Number, in seconds.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

Current time-domain cursor position (P)

df Display a single FID (C)
dfid Display a single FID (C)

Dept Set up parameters for DEPT experiment (M)

Description: Set up parameters for DEPT experiment

See also: NMR Spectroscopy User Guide

Related: adept Automatic DEPT analysis and spectrum editing (C)

autodept Automated complete analysis of DEPT data (M)
deptq1 Set up parameters for DEPTGL pulse sequence (M)

deptproc Process array of DEPT spectra (M)
padept Plot automatic DEPT analysis (C)
ppcal Proton decoupler pulse calibration (M)

deptgl Set up parameters for DEPTGL pulse sequence (M)

Description: Macro for the DEPTGL pulse sequence for spectral editing and polarization

transfer experiments.

See also: NMR Spectroscopy User Guide

Related: Dept Set up parameters for DEPT pulse sequence (M)

deptproc Process array of DEPT spectra (M)

Description: Automatically processes arrays of DEPT-type spectra. The FIDs are

transformed (using lb=2.5), phased, and scaled. In foreground operation, a stacked display is produced. By default, an automatic DEPT analysis (adept)

is performed.

See also: NMR Spectroscopy User Guide

Related: adept Automatically edit DEPT spectra (C)

Dept Set up parameters for DEPT experiment

Line broadening along the directly detected dimension (P)

pldept Plot DEPT type spectra (M)
procplot Automatically process FIDs (M)

destroy Destroy a parameter (C)

Syntax: destroy(parameter<,tree>)

Description: Removes a parameter from one of the parameter trees. If the destroyed

parameter was an array, the array parameter is automatically updated.

If destroy is called for a non-existent parameter, the command will abort with a message. If an optional return value is given, it will indicate success (1) or

failure (0) and the command will not abort.

Arguments: parameter is the name of the parameter to be destroyed.

tree is a keyword for the type of parameter tree: 'global', 'current', 'processed', or 'systemglobal'. The default is 'current'. Refer to

the create command for more information on types of trees.

Examples: destroy('a')

destroy('c','global')

See also: User Programming

Related: array Parameter order and precedence (P)

create Create new parameter in a parameter tree (C)
display Display parameters and their attributes (C)

Edit a variable and its attributes using vi text editor (C)

prune Prune extra parameters from current tree (C)

destroygroup Destroy parameters of a group in a tree (C)

Syntax: destroygroup(group<,tree>)

Description: Removes parameters of a group from one of the parameters trees.

Arguments: group is a keyword for the type of parameter group: 'all', 'sample',

'acquisition', 'processing', 'display', or 'spin'.

tree is a keyword for the type of parameter tree: 'global', 'current',

or 'processed'. The default is 'current'. Refer to the create

command for more information on trees.

Examples: destroygroup('sample')

destroygroup('all','global')

See also: User Programming

Related: create Create new parameter in a parameter tree (C)

destroy Destroy a parameter (C)

display Display parameters and their attributes (C)

groupcopy Copy parameters of group from one tree to another (C)

setgroup Set group of a variable in a tree (C)

# df Display a single FID (C)

Description: Displays a single FID. Parameter entry after an FID has been displayed causes

the display to be updated. The FID is left-shifted by the number of complex data points specified by the parameter <code>lsfid</code>. The FID is also phase-rotated (zero-order only) by the number of degrees specified by the parameter <code>phfid</code>. Left shifting and phasing can be avoided by setting <code>lsfid</code> and <code>phfid</code> to <code>'n'</code>. df

is identical in function to the dfid command.

Arguments: index (used with syntax 1) is the number of a particular FID for arrayed 1D

experiments or for 2D experiments. Default is 1.

options (used with syntax 2) is any of the following:

- 'toggle' is a keyword to switch between box and cursor modes.
- 'restart' is a keyword to redraw the cursor if it has been turned off.
- 'expand' is a keyword to switch between expanded and full views of the FID.
- 'imaginary' is a keyword to switch on and off the display of the imaginary FID.
- 'sfwf' is a keyword to interactively adjust the start and width of the FID display.
- 'phase' is a keyword to enter an interactive phasing mode.
- 'dscale' is a keyword to toggle the scale below the FID on and off.

Examples: df

df(4)

df('restart')

See also: NMR Spectroscopy User Guide

Related: crmode Current state of cursors in dfid, ds, or dconi (P)

dfid Display a single FID (C)

df2d Display FIDs of 2D experiment (C)

dfmode Current state of display of imaginary part of a FID (P)

lsfid Number of complex points to left-shift the np FID (P)

phfid Zero-order phasing constant for the np FID (P)

# df2d Display FIDs of 2D experiment (C)

Syntax: df2d<(<'nf',><array index>)>

Description: Produces a color intensity map of the raw 2D FIDs as a function of t<sub>1</sub> and t<sub>2</sub>.

The display can be modified by subsequent display commands, for example, df2d dconn will display the 2D FIDs without clearing the graphics screen.

Arguments: 'nf' is a keyword specifying that the data has been collected in the

compressed form using nf. In other words, each array element is collected as

one 2D FID or image comprised of nf FIDs or traces.

array index is the index of the array to be displayed.

Examples: df2d

df2d(1)

See also: NMR Spectroscopy User Guide

Related: dconi Interactive 2D data display (C)

df Display a single FID (C)

D

dfid Display a single FID (C)

Syntax: (1) dfid<(index) >
 (2) dfid<(options) >

Description: Functions the same as the df command. See df for information.

See also: NMR Spectroscopy User Guide

Related: df Display a single FID (C)

dfmode Current state of display of imaginary part of a FID (P)

Description: Holds a string variable that reflects the state of display of the imaginary part of

a FID. dfmode is primarily used by the programmable menu dfid to determine the status of the display of the imaginary part of a FID.

Values: 'r' indicates the current display is real only.

'i' indicates the current display is imaginary.

'z' indicates the display is zero imaginary.

See also: User Programming

dfrq Transmitter frequency of first decoupler (P)

Description: Contains the transmitter frequency for the first decoupler. dfrq is

automatically set when the parameter dn is changed and should not be

necessary for the user to manually set.

Values: Frequency, in MHz. The value is limited by synthesizer used with the channel.

See also: NMR Spectroscopy User Guide

Related: dfrq2 Transmitter frequency of second decoupler (P)

dfrq3 Transmitter frequency of third decoupler (P)

dfrq4 Transmitter frequency of fourth decoupler (P)

dn Nucleus for first decoupler (P)

dof Frequency offset for first decoupler (P)

sfrq Transmitter frequency of observe nucleus (P)

spcfrq Display frequencies of rf channels (M)

dfrq2 Transmitter frequency of second decoupler (P)

Applicability: Systems with a second decoupler.

Description: Contains the transmitter frequency for the second decoupler. dfrq2 is

automatically set when parameter dn2 is changed and should not be necessary

for the user to manually set.

Values: Frequency, in MHz. Value is limited by synthesizer used with the channel. If

dn2=' ' (two single quotes with no space in between) and a second decoupler

is present in the console, dfrq2 is internally set to 1 MHz.

See also: NMR Spectroscopy User Guide

Related: dn2 Nucleus for second decoupler (P)

dof2 Frequency offset for second decoupler (P)

dfrq3 Transmitter frequency of third decoupler (P)

Applicability: Systems with a third decoupler.

Description: Contains the transmitter frequency for the third decoupler. dfrq3 is

automatically set when the parameter dn3 is changed and should not be

necessary for the user to manually set.

Values: Frequency, in MHz. Value is limited by synthesizer used with the channel. If

dn3=' ' (two single quotes with no space in between) and a third decoupler is

present in the console, dfrq3 is internally set to 1 MHz.

See also: NMR Spectroscopy User Guide

Related: dn3 Nucleus for third decoupler (P)

dof3 Frequency offset for third decoupler (P)

# dfrq4 Transmitter frequency of fourth decoupler (P)

Applicability: Systems with a deuterium decoupler channel as the fourth decoupler.

Description: Contains the transmitter frequency for the fourth decoupler. dfrq4 is

automatically set when the parameter dn4 is changed and should not be

necessary for the user to manually set.

Values: Frequency, in MHz. Value is limited by a synthesizer used with the channel. If

dn4='' (two single quotes with no space in between) and a fourth decoupler

is present in the console, dfrq4 is internally set to 1 MHz.

See also: NMR Spectroscopy User Guide

Related: dn4 Nucleus for fourth decoupler (P)

dof4 Frequency offset for fourth decoupler (P)
spcfrq Display frequencies of rf channels (M)

rftype type of rf generation

# dfs Display stacked FIDs (C)

Syntax: dfs<(<start><,finish><,step><,'all'|'imag'><,color>)>

Description: Displays one or more FIDs. The position of the first FIDs is governed by the

parameters wc, sc, and vpf. A subsequent FID is positioned relative to the

preceding FID by the parameters vo and ho.

Arguments: start is the index number of the first FID for multiple FIDs. It can also be the

index number of a particular FID for arrayed 1D or 2D data sets.

finish is the index number of the last FID for multiple FIDs. To include all FIDs, set start to 1 and finish to arraydim (see example below).

step is the increment for the FID index. The default is 1.

'all' is a keyword to display all of the FIDs. This is the default.

'imag' is a keyword to display only the imaginary FID channel.

color is the color of the display: 'red', 'green', 'blue', 'cyan',

'magenta', 'yellow', 'black', or 'white'.

Examples: dfs(1,arraydim,3)

dfs('imag')

See also: NMR Spectroscopy User Guide

Related: arraydim Dimension of experiment (P)

dfsa Display stacked FIDs automatically (C)

dfsan Display stacked FIDs automatically without screen erase (C)

dfsh Display stacked FIDs horizontally (C)

dfshn Display stacked FIDs horizontally without screen erase (C)

dfsn Display stacked FIDs without screen erase (C)

dfww Display FIDs in whitewash mode (C)

ho Horizontal offset (P)

plfid Plot FID (C)

Plot FIDs in whitewash mode (C)

Start of chart (P)
vo Vertical offset (P)

vpf Current vertical position of FID (P)

WC Width of chart (P)

## dfsa Display stacked FIDs automatically (C)

Syntax: dfsa<(<start><,finish><,step><,'all'|'imag'><,color>)>

Description: Displays one or more FIDs automatically by adjusting the parameters vo and

ho to fill the screen in a lower left to upper right presentation (wc must be set to less than full screen width for this to work). The position of the first FID is

governed by parameters wc, sc, and vpf.

Arguments: start is the index number of the first FID for multiple FIDs. It can also be the

index number of a particular FID for arrayed 1D or 2D data sets.

finish is the index number of the last FID for multiple FIDs. step is the increment for the FID index. The default is 1.

'all' is a keyword to display all of the FIDs. This is the default.
'imag' is a keyword to display only the imaginary FID channel.

color is the color of the display: 'red', 'green', 'blue', 'cyan',

'magenta', 'yellow', 'black', or 'white'.

See also: NMR Spectroscopy User Guide

Related: dfs Display stacked FIDs (C)

dfsan Display stacked FIDs automatically without screen erase (C)

## dfsan Display stacked FIDs automatically without screen erase (C)

Syntax: dfsan<(<start><,finish><,step><,'all'|'imag'><,color>)>

Description: Functions the same as the command dfsa except the graphics window is not

erased before starting the display. This allows composite displays of many FIDs

to be created. The arguments are the same as dfsa.

See also: NMR Spectroscopy User Guide

Related: dfsa Display stacked FIDs automatically (C)

### dfsh Display stacked FIDs horizontally (C)

Syntax: dfsh<(<start><,finish><,step><,'all'|'imag'><,color>)>

Description: Displays one or more FIDs horizontally by setting vo to zero and adjusting ho,

sc, and wc to fill the screen from left to right with the entire array. The position

of the first FID is governed by parameters wc, sc, and vpf.

Arguments: start is the index number of the first FID for multiple FIDs. It can also be the

index number of a particular FID for arrayed 1D or 2D data sets.

finish is the index number of the last FID for multiple FIDs. To display all

FIDs, set finish to the parameter arraydim.

step is the increment for the FID index. The default is 1.

'all' is a keyword to display all of the FIDs. This is the default.

'imag' is a keyword to display only the imaginary FID channel.

color is the color of the display: 'red', 'green', 'blue', 'cyan',

'magenta', 'yellow', 'black', or 'white'.

See also: NMR Spectroscopy User Guide

Related: dfs Display stacked FIDs (C)

dfshn Display stacked FIDs horizontally without screen erase (C)

# dfshn Display stacked FIDs horizontally without screen erase (C)

Syntax: dfshn<(<start><,finish><,step><,'all'|'imag'><,color>)>

Description: Functions the same as the command dfsh except the graphics window is not

erased before starting the display. This allows composite displays of many FIDs

to be created. The arguments are the same as dfsh.

See also: NMR Spectroscopy User Guide

Related: dfsh Display stacked FIDs horizontally (C)

# dfsn Display stacked FIDs without screen erase (C)

Syntax: dfsn<(<start><,finish><,step><,'all'|'imag'><,color>)>

Description: Functions the same as the command dfs except the graphics window is not

erased before starting the display. This allows composite displays of many FIDs

to be created. The arguments are the same as dfs.

See also: NMR Spectroscopy User Guide

Related: dfs Display stacked FIDs (C)

### dfww Display FIDs in whitewash mode (C)

Syntax: dfww<(<start><,finish><,step><,'all'|'imag'><,color>)>

Description: Displays FIDs in whitewash mode (after the first FID, each FID is blanked out

in regions in which it is behind an earlier FID). The position of the first FIDs is

governed by parameters wc, sc, and vpf.

Arguments: start is the index number of the first FID for multiple FIDs. It can also be the

index number of a particular FID for arrayed 1D or 2D data sets.

finish is the index number of the last FID for multiple FIDs.

step is the increment for the FID index. The default is 1.

'all' is a keyword to display all of the FIDs. This is the default.

'imag' is a keyword to display only the imaginary FID channel.

color is the color of the display: 'red', 'green', 'blue', 'cyan',

'magenta', 'yellow', 'black', or 'white'.

See also: NMR Spectroscopy User Guide

Related: dfs Display stacked FIDs (C)

pfww Plot FIDs in whitewash mode (C)

# dg Display group of acquisition/processing parameters (C)

Syntax: dg('template',<'file name'>)

Description: Displays the group of acquisition and 1D/2D processing parameters. To display

an individual parameter, enter the name of the parameter followed by a question mark (e.g., sw?). Parameters do not have to be displayed in order to be entered

or changed. The dq display is controlled by the string parameter dq.

Arguments: template is the name of the template parameter. The default is 'dg'. See the

manual *User Programming* for rules on constructing a template. The macros dg dg1, dg2, dglp, and dgs activate dg with a template argument such as 'dg','dg1', 'dg2', 'dglp', 'dgs', etc. or a user defined template.

 $\verb|file_name| is the name of the file to which the \verb|dg| command will write the$ 

parameters specified by template.

Examples: dg

dg('dgexp')
dg('dg','dgout')

See also: NMR Spectroscopy User Guide; User Programming

Related: ? Display the value of an individual parameter (C)

da Display acquisition parameter arrays (C)

dglp Display group of linear prediction parameters (C)
da Display acquisition parameter arrays (P)

da Display acquisition parameter arrays (P)
dg Control dg parameter group display (P)

dglp Control dglp parameter group of linear prediction parameters (P)

dg1 Display group of display parameters (M)

dg2 Display group of 3rd and 4th rf channel/3D parameters (M)

dgs Display group of special/automation parameters (M)

# dg Control dg parameter group display (P)

Description: Controls the display of the dg command for the group of acquisition and 1D/2D

processing parameters. dg, a string parameter, can be modified with the

command paramvi('dg').

See also: NMR Spectroscopy User Guide

Related: dg Display group of acquisition/processing parameters (C)

paramvi Edit a parameter and its attributes with vi text editor (C)

# dg1 Display group of display parameters (M)

Description: Displays the group of display parameters. To display an individual parameter,

enter the name of the parameter followed by a question mark (e.g., sp?). Parameters do not have to be displayed in order to be entered or changed. The

dg1 display is controlled by the string parameter dg1.

See also: NMR Spectroscopy User Guide

Related: ? Display individual parameter value (C)

dg1 Control dg1 parameter group display (P)

dg Display group of acquisition/processing parameters (C)

# dg1 Control dg1 parameter group display (P)

Description: Controls the display of the dg1 command for the group of display parameters.

dg1, a string parameter, can be modified with paramvi('dg1').

See also: NMR Spectroscopy User Guide

Related: dg1 Display group of display parameters (M)

Edit a parameter and its attributes with *vi* text editor (C)

### dg2 Display group of 3rd and 4th rf channel/3D parameters (M)

Description: Displays the group of acquisition parameters associated with a second

decoupler channel on a system with a third rf channel. It also displays the group

of parameters associated with selective 2D processing of 3D data sets. To display an individual parameter, enter the name of the parameter followed by a question mark (e.g., sw?). Parameters do not have to be displayed in order to be entered or changed. The dg2 display is controlled by the string parameter

dq2.

See also: NMR Spectroscopy User Guide

Related: dq Display group of acquisition/processing parameters (C)

dg2 Control dg2 parameter group display (P)

dg2 Control dg2 parameter group display (P)

Description: Controls the display of the dq2 command for the group of 3rd and 4th rf

channel/3D parameters. dg2, a string parameter, can be modified with the command paramvi ('dg2'). To retrieve the dg2 and ap display templates

for the current experiment, enter addpar('3rf').

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

dg2 Display group of 3rd and 4th rf channel/3D parameters (M)

paramvi Edit a parameter and its attributes with vi text editor (M)

dga Display group of spin simulation parameters (M)

Description: Displays the file of spin simulation parameters (Group A). There is one such

group of parameters in the data system, not one per experiment as with normal

NMR parameters.

See also: NMR Spectroscopy User Guide

Related: dg Display group of acquisition/processing parameters (C)

dla Display spin simulation parameter arrays (C)

DgcsteSL Set up parameters for DgcsteSL pulse sequence (M)

Description: Converts a parameter set to DgcsteSL experiment.

See also: NMR Spectroscopy User Guide

Related: dosy Process DOSY experiments (M)

fiddle Perform reference deconvolution (M)

setup\_dosy Set up gradient levels for DOSY experiments (M)

Dgcstecosy Set up parameters for Dgcstecosy pulse sequence (M)

Description: Converts a parameter set to Dgcstecosy experiment

See also: NMR Spectroscopy User Guide

Related: dosy Process DOSY experiments (M)

makeslice Synthesize 2D projection of a 3D DOSY spectrum (C)
setup\_dosy Set up gradient levels for DOSY experiments (M)
showoriginal Restore first 2D spectrum in 3D DOSY spectrum (M)

Dgcstehmqc Set up parameters for Dgcstehmqc pulse sequence (M)

Description: Converts a parameter set to Dgcstehmqc experiment

D

See also: NMR Spectroscopy User Guide

Related: dosy Process DOSY experiments (M)

makesliceSynthesize 2D projection of 3D DOSY spectrum (C)setup\_dosySet up gradient levels for DOSY experiments (M)showoriginalRestore first 2D spectrum in 3D DOSY spectrum (M)

dglc Display group of LC-NMR parameters (M)

Applicability: Systems with LC-NMR accessory.

Description: Displays parameters related to LC-NMR on a separate screen. This macro is

equivalent to the command dg('dglc').

See also: NMR Spectroscopy User Guide

Related: dglc Control LC-NMR parameter display (P)

dglc Control dglc parameter group display (P)

Applicability: Systems with LC-NMR accessory.

Description: Controls the display of the LC-NMR parameters by the macro dglc and the

equivalent command dg('dglc'). If this parameter does not exist, the

parlc macro can create it.

See also: NMR Spectroscopy User Guide

Related: dglc Display LC-NMR parameters (M)

parlc Create LC-NMR parameters (M)

dglp Display group of linear prediction parameters (C)

Syntax: dqlp

Description: Displays the linear prediction parameters group. Parameters do not have to be

displayed in order to be entered or changed. The dqlp display is controlled by

the string parameter dqlp.

Examples: dqlp

See also: NMR Spectroscopy User Guide; User Programming
Related: dg Control dg parameter group display (P)

dgs Display group of shims and automation parameters (M)

Description: Displays the group of shims and automation parameters. To display an

individual parameter, enter name of the parameter followed by a question mark (e.g., sw?). Parameters do not have to be displayed in order to be entered or

changed. The dgs display is controlled by the parameter dgs.

See also: NMR Spectroscopy User Guide

Related: dg Display group of acquisition/processing parameters (C)

dgs Control dgs parameter group display (P)

dgs Control dgs parameter group display (P)

Description: Controls display of the dgs command for the group of shims and automation

parameters. dgs, a string parameter, can be modified by paramvi ('dgs').

Related: dgs Display group of special/automation parameters (M)

paramvi Edit a parameter and its attributes with vi text editor (C)

dhp Decoupler high-power control with class C amplifier (P)

Applicability: System with a class C amplifier.

Description: dhp selects a decoupler high-power level for systems with class C amplifiers

on the decoupler channel. Specific values of dhp should be calibrated periodically for any particular instrument and probe combination. As a rough

guide, dhp=75 corresponds to approximately 2 watts at 200 MHz.

CAUTION: Decoupler power greater than 2 watts in a switchable probe will

damage the probe. Always carefully calibrate high-power decoupling

to avoid exceeding 2 watts of power.

For systems equipped with a linear amplifier on the decoupler channel, dhp is

nonfunctional and is replaced by the parameter dpwr.

Note that dhp runs in the opposite direction from dlp (i.e., for dhp a higher number means more power, for dlp a higher number means less power).

Values: 0 to 255 (where 255 is maximum power) in uncalibrated, non-linear units.

'n' selects low-power decoupling under the control of the parameter dlp.

See also: NMR Spectroscopy User Guide

Related: dlp Decoupler low power with class C amplifier (P)

dpwr Power level for first decoupler with linear amplifier (P)

Nucleus for observe transmitter (P)

dialog Display a dialog box from a macro (C)

Syntax: dialog(definition\_file,output\_file<,'nowait'>)

Description: Opens a dialog box from a macro. The output is written to a file that can be read

by the macro using the lookup command.

Arguments: definition file is the name of the file (specified by an absolute path)

that defines the layout of the dialog box.

output file is the name of the file (specified by an absolute path) where the

results of the dialog box are written.

'nowait' is a keyword to return immediately, without waiting for input into

the dialog box.

Examples: dialog(userdir+'/dialoglib/array,'/tmp/array')

See also: User Programming

Related: lookup Look up words and lines from a text file (C)

diffparams Report differences between two parameter sets (U)

Syntax: diffparams <-list> file1 file2 <macroname>

Description: Reports differences between parameter sets. A macro can optionally be created

that will convert file1 into file2.

Arguments: file1 and file2 are parameter files, like \$HOME/vnmrsys/exp1/

procpar \$HOME/vnmrsys/exp1/curpar \$HOME/vnmrsys/global /vnmr/conpar xyz.fid/procpar file1 and file2 can also be directories (xyz.fid or xyz.par, or a local experiment like ~/vnmrsys/exp1); in this case diffparams will look for a subfile procpar

in these directories. The optional <code>-list</code> argument will cause a list of the parameters which are different to be printed. If the <code>-list</code> option is used, the macro feature is turned off. If a parameter exists in <code>file1</code> but not <code>file2</code>, it is not listed. If a parameter exists in <code>file2</code> but not <code>file1</code>, it is listed. If the parameter exists in both files, it is listed if the values are different. It is not listed if other information associated with the parameter is different. This other information is things like protection bits, maximum values, group, type, etc.

An optional third argument specifies the pathname of a macro to output. This macro will contain the MAGICAL commands necessary to convert file1 into

file2.

Examples: diffparams abc.fid xyz.fid

diffparams -list abc.fid xyz.fid

diffparams ~/vnmrsys/exp1 ~/vnmrsys/exp3

diffparams ~/vnmrsys/exp1 ~/vnmrsys/exp3 ~/vnmrsys/

maclib/change1to3

# diffshims Compare two sets of shims (M,U)

Syntax: diffshims(shimfile1,shimfile2)

(From UNIX) diffshims shimfile1 shimfile2

Description: Compares values for room-temperature shims stored in two separate files.

Arguments: shimfile1 and shimfile2 are names of separate files containing shim

values. Both files must have been written using the svs command.

See also: NMR Spectroscopy User Guide

Related: svs Save shim coil settings (C)

# digfilt Write digitally filtered FIDs to another experiment (M)

Syntax: digfilt(exp number<,option>)

Description: Saves digitally filtered FIDs to another experiment.

Arguments: exp number specifies the number of the experiment, from 1 to 9, for saving

the FIDs.

option is one of the keywords 'nodc', 'zero', 'lfs', 'zfs', or 't2dc'. Use a keyword for an option if the same option was used when

processing the data with ft, wft, ft2d, or wft2d.

See also: NMR Spectroscopy User Guide

Related: downsamp Sampling factor applied after digital filtering (P)

ft Fourier transform 1D data (C) ft2d Fourier transform 2D data (C)

wft Weight and Fourier transform 1D data (C)
wft2d Weight and Fourier transform 2D data (C)

# dir List files in directory (C)

Syntax: dir<(string)>

Description: Displays files in a directory on the text window. The dir command is identical

to the ls and lf commands.

Arguments: string is a string argument containing the options and/or directory names

used if this were the UNIX 1s command (e.g., dir('-1 \*.fid') requests a long listing (-1) of all files ending with .fid (\*.fid)). If no argument is

entered, dir lists all files in the current working directory.

Examples: dir

dir('data')
dir('-l \*.fid')

See also: NMR Spectroscopy User Guide

Related: 1f List files in directory (C)

1s List files in directory (C)

# display Display parameters and their attributes (C)

Syntax: display(parameter|'\*'|'\*\*'<,tree>)

Description: Displays one or more parameters and their attributes from a parameter tree.

Arguments: Three levels of display are available: parameter, '\*', and '\*\*'.

- parameter is the name of a single parameter and the display is of its attributes (e.g., display('a') displays the attributes of parameter a in the (default) current tree).
- '\*' is a keyword to display the name and values of all parameters in a tree (e.g., display('\*', 'global') displays all parameter names and values in the global tree).
- '\*\*' is a keyword to display the attributes of all parameters in a tree (e.g., display('\*\*', 'processed') displays the attributes of all parameters in the processed tree).

tree is the type of parameter tree and can be 'global', 'current', 'processed', or 'systemglobal'. The default is 'current'. Refer to the create command for more information on types of trees.

Examples: display('a')

display('\*','global')
display('\*\*','processed')

See also: User Programming

Related: create Create new parameter in a parameter tree (C)

destroy Destroy a parameter (C)

paramvi Edit a parameter and its attributes with the vi text editor (C)

prune Prune extra parameters from current tree (C)

# dla Display spin simulation parameter arrays (M)

Syntax: dla<('long')>

Description: Displays the parameters containing the line assignments for spin simulation

iteration (matching simulated spectra to actual data). A <code>clindex</code> value of a calculated transition gives the index of the assigned measured line. The value is

zero for unassigned transitions.

Arguments: 'long' is a keyword to display the parameters containing the line assignments

for spin simulation iteration (matching simulated spectra to actual data) and put the line assignments into the file spini.la. This option is most useful when the dla display is too large to display all the calculated transitions in the text window. The dlalong command operates the same as the dla('long')

command.

Examples: dla

dla('long')

D

See also: NMR Spectroscopy User Guide

Related: assign Assign transitions to experimental lines (M)

clindexIndex of experimental frequency of a transition (P)dgaDisplay parameters of spin simulation group (C)dlalongLong display of spin simulation parameter arrays (C)

dlalong Long display of spin simulation parameter arrays (C)

Syntax: dlalong

Description: Puts line assignments into the file spini.la in a more complete form, then

displays this file in the text window. It is most useful when the dla display is too large to display all the calculated transitions in the text window. The

dla('long') command operates the same as dlalong.

See also: NMR Spectroscopy User Guide

Related: dla Display spin simulation parameter arrays (M)

# dli Display list of integrals (C)

Description: Displays a list of integrals at the integral reset points. The frequency units of the

displayed list of integrals is controlled by the parameter axis. The reset points may be defined with the z command and these frequencies are stored in lifrq. The calculated amplitudes of the integral region are stored in liamp. The reset points are stored as hertz and are not referenced to rfl and rfp. The amplitudes are stored as the actual value; they are not scaled by ins or by

insref. When the integral blanking mode is used (i.e.,

intmod='partial'), only the integrals corresponding to the displayed

integral regions are listed.

The displayed integral value can be scaled with the setint macro. The

integral is scaled by the parameters ins and insref.

See also: NMR Spectroscopy User Guide

Related: axis Axis label for displays and plots (P)

Cz Clear integral reset points (C)

dlni Display list of normalized integrals (M)

ins Integral normalization scale (P)

insref Fourier number scaled value of an integral (P)

liamp Amplitudes of integral reset points (P)lifrq Frequencies of integral reset points (P)

nli Find integral values (C)

Reference peak position in directly detected dimension (P)

Reference peak frequency in directly detected dimension (P)

set int Set value of an integral (M)

z Add integral reset point at cursor position (C)

# dlivast Produce text file and process wells (M)

Applicability: VAST accessory.

Syntax: dlivast<(last)>

Description: Produces a text file containing the integral of the partial regions and processes

the wells.

Arguments: last is the number of the last well. The default is 96.

Related: combiplate View a color map for visual analysis of VAST microtiter plate (U)

combishow Display regions as red, green, and blue in CombiPlate window (M)

# dll Display listed line frequencies and intensities (C)

Syntax: dll<('pos'<,noise\_mult>)><:number\_lines,scale>

Description: Displays a list of line frequencies and amplitudes that are above a threshold

defined by th. Frequency units are defined by the parameter axis. The results of this calculation are stored in llfrq and llamp. The frequencies are stored as Hz and are not referenced to rfl and rfp. Amplitudes are stored as the

actual data point value; they are not scaled by vs.

Arguments: 'pos' is a keyword to list only positive lines.

noise\_mult is a numerical value that determines the number of noise peaks listed for broad, noisy peaks. The default value is 3. A smaller value results in more peaks, a larger value results in fewer peaks, and a value of 0.0 results in a line listing containing all peaks above the threshold th. Negative values of

noise mult are changed to 3.

number lines is a return argument with the number of lines above the

threshold.

scale is a return argument with a scaling factor for line amplitudes. This scaling factor accounts for vs and whether the lines are listed in absolute

intensity mode or normalized mode.

Examples: dll

dll('pos')
dll(2.5)
dll:r1,sc

See also: NMR Spectroscopy User Guide

Related: axis Axis label for displays and plots (P)

dels Delete spectra from  $T_1$  or  $T_2$  analysis (C)

Find peak heights (C)

getll Get frequency and intensity of a line (C)

llampList of line amplitudes (P)llfrqList of line frequencies (P)

nl Position the cursor at the nearest line (C)
nll Find line frequencies and intensities (C)

rfl Reference peak position in directly detected dimension (P)
rfp Reference peak frequency in directly detected dimension (P)

th Threshold (P)
vs Vertical scale (P)

# dlni Display list of normalized integrals (M)

Description: Displays integrals in a normalized format. The parameter ins represents the

value of the sum of all the integrals. When the integral blanking mode is used (i.e., intmod='partial'), only the integrals corresponding to the displayed

integral regions are listed and are used in the summation.

See also: NMR Spectroscopy User Guide

cz Clear integral reset points (C)
dli Display list of integrals (C)
ins Integral normalization scale (P)

nli Find integral values (C)

Z Add integral reset point at cursor position (C)

# dlp Decoupler low-power control with class C amplifier (P)

Applicability: Systems with a class C amplifier.

Description: dlp controls the decoupler power level for systems with a class C decoupler

amplifier in the low-power mode, generally used for homonuclear decoupling. dlp specifies dB of attenuation of the decoupler, below a nominal 1 watt value.

dlp is active only if dhp='n'.

On systems with a decoupler linear amplifier, dlp is nonfunctional and dpwr

controls decoupler power.

Values: 0 to 39 (in dB of attenuation, 0 is maximum power).

See also: NMR Spectroscopy User Guide

Related: dhp Decoupler high-power control with class C amplifier (P)

dm Decoupler mode for first decoupler (P)

dmf Decoupler modulation frequency for first decoupler (P)
dpwr Power level for first decoupler with linear amplifier (P)

# dm Decoupler mode for first decoupler (P)

Description: Determines the state of first decoupler during different status periods within a

pulse sequence (refer to the manual *User Programming* for a discussion of status periods). Pulse sequences may require one, two, three, or more different decoupler states. The number of letters that make up the dm parameter vary appropriately, with each letter representing a status period (e.g., dm='yny' or dm='ns'). If the decoupler status is constant for the entire pulse sequence, it

can be entered as a single letter (e.g., dm='n').

Values: 'n', 'y', 'a', or 's' (or a combination of these values), where:

'n' specifies no decoupler rf.

'y' specifies the asynchronous mode. In this mode, the decoupler rf is gated on and modulation is started at a random places in the modulation sequence.

'a' specifies the asynchronous mode, the same as 'y'.

's' specifies the synchronous mode in which the decoupler rf is gated on and modulation is started at the beginning of the modulation sequence.

See also: NMR Spectroscopy User Guide

Related: dm2 Decoupler mode for second decoupler (P)

dm3 Decoupler mode for third decoupler (P)
dm4 Decoupler mode for fourth decoupler (P)

dmf Decoupler modulation frequency for first decoupler (P)
dmm Decoupler modulation mode for first decoupler (P)

dn Nucleus for first decoupler (P)

decasynctype Select the type of decoupler asynchronous mode (P)

# dm2 Decoupler mode for second decoupler (P)

Applicability: Systems with a second decoupler.

Description: Determines the state of second decoupler during different status periods within

a pulse sequence. It functions analogously to dm.

Values: Same as dm, except that if dn2='' (two single quotes with no space in

between) and a second decoupler is present in the console, dm2 assumes a

default value of 'n' when go is executed.

See also: NMR Spectroscopy User Guide

Related: Decoupler mode of first decoupler (P)

> dmf2 Decoupler modulation frequency for second decoupler (P) dmm2 Decoupler modulation mode for second decoupler (P)

dn2 Nucleus for second decoupler (P)

#### Decoupler mode for third decoupler (P) dm3

Applicability: Systems with a third decoupler.

Description: Determines the state of third decoupler during different status periods within a

pulse sequence. It functions analogously to dm.

Values: Same as dm, except that if dn3=' ' (two single quotes with no space in

between) and a third decoupler is present in the console, dm3 assumes a default

value of 'n' when go is executed.

See also: NMR Spectroscopy User Guide

Related: Decoupler mode of first decoupler (P) dm

> dmf3 Decoupler modulation frequency for third decoupler (P) dmm3 Decoupler modulation mode for third decoupler (P)

Nucleus for third decoupler (P) dn3

Select the type of decoupler asynchronous mode (P) decasynctype

#### Decoupler mode for fourth decoupler (P) dm4

Applicability: Systems with a deuterium decoupler channel as the fourth decoupler.

Determines the state of fourth decoupler during different status periods within Description:

a pulse sequence. It functions analogously to dm.

Values: Same as dm, except that if dn4=' ' (two single quotes with no space in

between) and a fourth decoupler is present in the console, dm4 assumes a

default value of 'n' when go is executed.

See also: NMR Spectroscopy User Guide

Related: Decoupler mode of first decoupler (P)

> dmf4 Decoupler modulation frequency for fourth decoupler (P) dmm4 Decoupler modulation mode for fourth decoupler (P)

Nucleus for fourth decoupler (P) dn4

Select the type of decoupler asynchronous mode (P) decasynctype

#### Decoupler modulation frequency for first decoupler (P) dmf

Description: Controls modulation frequency of the first decoupler. It specifies 1/pw90 at

the particular power level used. After calibrating the decoupler field strength  $\gamma H_2$  (expressed in units of Hz), dmf should be set equal to  $4*\gamma H_2$  for WALTZ,

MLEV16, GARP, and XY32 (when available).

dmf is inactive for CW mode decoupling (dmm='c').

dmf is also active for square wave mode decoupling (dmm='r') and fm-fm mode (dmm='f') decoupling. For dmm='f', the modulation frequency is swept back and forth between about 0.5% and 5% of the dmf frequency (e.g., if dmf is 100 kHz, the modulation is swept between approximately 500 Hz and 5 kHz). A reasonable optimum value for dmf when dmm= 'f' is the decoupler

frequency divided by 4000.

Values: 5 Hz to 2 MHz in steps of 5 Hz (steps are actually approximately 4.768 Hz).

For GARP modulation, the dmf value is internally multiplied by 45, making the

limit of possible dmf values to 5 Hz to 44.4 kHz when dmm='q'.

See also: NMR Spectroscopy User Guide

Related: dmf2 Decoupler modulation frequency for second decoupler (P)

> dmf3 Decoupler modulation frequency for third decoupler (P) dmf4 Decoupler modulation frequency for fourth decoupler (P) Decoupler modulation mode for first decoupler (P) dmm

0 ewg 90° pulse width (P)

#### dmf2 Decoupler modulation frequency for second decoupler (P)

Applicability: Systems with a second decoupler.

Description: Controls the modulation frequency of the second decoupler. It functions

analogously to the parameter dmf.

Values: Same as dmf except that if dn2=' ' (two single quotes with no space in

> between) and a second decoupler is present in the console (numrfch greater than 2), dmf2 assumes a default value of 1000 Hz when go is executed.

See also: NMR Spectroscopy User Guide

Related: dm2 Decoupler mode for second channel (P)

> dmf Decoupler modulation frequency for first decoupler (P) Decoupler modulation mode for second decoupler (P) dmm2

dn2 Nucleus for second decoupler (P) numrfch Number of rf channels (P)

#### dmf3 Decoupler modulation frequency for third decoupler (P)

Applicability: Systems with a third decoupler.

Description: Controls the modulation frequency of the third decoupler. It functions

analogously to the parameter dmf.

Values: Same as dmf except that if dn3=' ' (two single quotes with no space in

between) and a third decoupler is present in the console (numrfch equals 4),

dmf3 assumes a default value of 1000 Hz when go is executed.

See also: NMR Spectroscopy User Guide

Related: Decoupler mode for third channel (P)

> dmf Decoupler modulation frequency for first decoupler (P) dmm3 Decoupler modulation mode for third decoupler (P)

dn3 Nucleus for third decoupler (P) Number of rf channels (P) numrfch

#### Decoupler modulation frequency for fourth decoupler (P) dmf4

Systems with a deuterium decoupler channel as the fourth decoupler. Applicability: Description:

Controls the modulation frequency of the fourth decoupler. It functions

analogously to the parameter dmf.

Values: Same as dmf except that if dn4=' ' (two single quotes with no space in

between) and a fourth decoupler is present in the console (numrfch equals 5),

dmf4 assumes a default value of 1000 Hz when go is executed.

Related: dm4 Decoupler mode for fourth channel (P)

dmf Decoupler modulation frequency for first decoupler (P)
dmm4 Decoupler modulation mode for fourth decoupler (P)

dn4 Nucleus for fourth decoupler (P)
numrfch Number of rf channels (P)

# dmfadj Adjust tip-angle resolution time for first decoupler (M)

Syntax: dmfadj<(tipangle resolution)>

Description: Adjusts the parameter dmf so that time associated with the first decoupler tip-

angle resolution is an integral multiple of 50 ns. This eliminates time truncation error in execution of programmable decoupling or spin-locking sequence by the waveform generator. For example, the tip-angle resolution for an MLEV-16 decoupling sequence should be  $90.0^\circ$  since every pulse in that sequence can be represented as an integral multiple of  $90.0^\circ$ ; however, the tip-angle resolution

for a GARP decoupling sequence should be 1.0°.

Arguments: tipangle resolution specifies the necessary tip-angle resolution for the

programmable decoupling or spin-locking sequence to be executed. The default

value is the current value of the parameter dres.

Examples: dmfadj

dmfadj (90.0)

See also: NMR Spectroscopy User Guide

Related: dmf Decoupler modulation frequency for first decoupler (P)

dmf2adjAdjust tip-angle resolution time for second decoupler (M)dmf3adjAdjust tip-angle resolution time third decoupler (M)dmf4adjAdjust tip-angle resolution time fourth decoupler (M)dresTip angle resolution for programmable decoupling (P)

## dmf2adj Adjust tip-angle resolution time for second decoupler (M)

Applicability: Systems with a second decoupler.

Syntax: dmf2adj<(tipangle\_resolution)>

Description: Adjusts the parameter dmf2 to make time associated with the second decoupler

tip-angle resolution an integral multiple of 50 ns. dmf2adj functions

analogously to the macro dmfadj.

Arguments: tipangle\_resolution specifies the necessary tip-angle resolution for the

programmable decoupling or spin-locking sequence to be executed. The default

value is the current value of the parameter dres2.

Examples: dmf2adj

dmf2adj(90.0)

See also: NMR Spectroscopy User Guide

Related: dmf2 Decoupler modulation frequency for second decoupler (P)

dmfadj Adjust decoupler tip-angle resolution time (M)
dres2 Tip angle resolution for second decoupler (P)

# dmf3adj Adjust tip-angle resolution time for third decoupler (M)

Applicability: Systems with a third decoupler.

Syntax: dmf3adj<(tipangle resolution)>

D

Description: Adjusts the parameter dmf3 to make time associated with the third decoupler

tip-angle resolution an integral multiple of 50 ns. dmf3adj functions

analogously to the macro dmfadj.

Arguments: tipangle resolution specifies the necessary tip-angle resolution for the

programmable decoupling or spin-locking sequence to be executed. The default

value is the current value of the parameter dres3.

Examples: dmf3adj

dmf3adj(90.0)

See also: NMR Spectroscopy User Guide

Related: dmf3 Decoupler modulation frequency for third decoupler (P)

dres3 Tip-angle resolution for third decoupler (P)

# dmf4adj Adjust tip-angle resolution time for fourth decoupler (M)

Applicability: Systems with a deuterium decoupler as the fourth decoupler.

Syntax: dmf4adj<(tipangle resolution)>

Description: Adjusts the parameter dmf4 to make time associated with the fourth decoupler

tip-angle resolution an integral multiple of 50 ns. dmf4adj functions

analogously to the macro dmfadj.

Arguments: tipangle resolution specifies the necessary tip-angle resolution for the

programmable decoupling or spin-locking sequence to be executed. The default

value is the current value of the parameter dres4.

Examples: dmf4adj

See also: NMR Spectroscopy User Guide

Related: dmf4 Decoupler modulation frequency for fourth decoupler (P)

dres4 Tip-angle resolution for fourth decoupler (P)

# dmg Data display mode in directly detected dimension (P)

Description: Controls the mode of data display along the directly detected dimension. dmg

is in the display group and can be set manually or by executing the commands ph, av, pwr, or pa for the values 'ph', 'av', 'pwr', or 'pa', respectively.

Values: 'ph' sets the *phased mode* in which each real point in the displayed spectrum

is calculated from a linear combination of real and imaginary points comprising

each respective complex data point.

'av' sets the *absolute-value mode* in which each real point in the displayed spectrum is calculated as the square root of the sum of squares of the real and imaginary points comprising each respective complex data point.

'pwr' sets the *power mode* in which each real point in the displayed spectrum is calculated as the sum of squares of the real and imaginary points comprising each respective complex data point.

'pa' sets the *phase angle* mode in which each real point in the displayed spectrum is calculated as the phase angle from the arc tangent of the real and imaginary points comprising each respective complex data point.

See also: NMR Spectroscopy User Guide

Related: aig Absolute intensity group (P)

Set absolute-value mode in directly detected dimension (C)

dcg Drift correction group (P)

dmg1 Data display mode in 1st indirectly detected dimension (P)

dmg2 Data display mode in 2nd indirectly detected dimension (P)

Fourier transform 1D data (C)

Fourier transform along f<sub>2</sub> dimension (C)

ft2d Fourier transform 2D data (C)

pa Set phase angle mode in directly detected dimension (C)
ph Set phased mode in directly detected dimension (C)

pmode Processing mode for 2D data (P)

pwr Set power mode in directly detected dimension (C)
wft Weigh and Fourier transform 1D data (C)
wft1d Weigh and Fourier transform of 2D data (C)
wft2d Weigh and Fourier transform 2D data (C)

# dmg1 Data display mode in 1st indirectly detected dimension (P)

Description: Controls the mode of data display along the first indirectly detected dimension

of a multidimensional data set. dmg1 is in the display group and can be set manually or by executing the commands ph1, av1, pwr1, or pa1 for the values 'ph1', 'av1', 'pwr1', or 'pa1', respectively. If dmg1 does not exist or if it is set to the empty string (dmg1=''), VnmrJ uses the value of dmg to decide the display mode along the first indirectly detected dimension.

Values: 'ph1' sets phased mode.

'av1' sets absolute-value mode.

'pwr1' sets power mode.

'pa1' sets phase angle mode.

See also: NMR Spectroscopy User Guide

Related: av1 Set absolute-value mode in 1st indirectly det. dim. (C)

dmg Data display mode in directly detected dimension (P)
pal Set phase angle mode in 1st indirectly detected dimension (C)
phl Set phased mode in 1st indirectly detected dimension (C)
pwrl Set power mode in 1st indirectly detected dimension (C)

## dmg2 Data display mode in 2nd indirectly detected dimension (P)

Description: Controls the mode of data display along the second indirectly detected

dimension of a multidimensional data set. dmg2 is in the display group and can be set manually or by executing the commands ph2, av2, or pwr2 for the values 'ph2', 'av2', or 'pwr2', respectively. If dmg2 does not exist or if it is set to the empty string (dmg2=''), VnmrJ uses the value of the parameter dmg instead of dmg2 to decide the display mode along the second indirectly

detected dimension.

Values: 'ph2' sets phased mode.

'av2' sets absolute-value mode.

'pwr2' sets power mode.

See also: NMR Spectroscopy User Guide

Related: av2 Set absolute-value mode in 2nd indirectly det. dim. (C)

dmg Data display mode in directly detected dimension (P)
ph2 Set phased mode in 2nd indirectly det. dim. (C)
pwr2 Set power mode in 2nd indirectly det. dim. (C)

# dmgf Absolute-value display of FID data or spectrum in acqi (P)

Description: If the parameter dmgf exists and is set to 'av', the FID display in the acqi

program is set to the absolute-value mode, which displays the square root of the sum of the squares of the real and imaginary channels. dmgf has no function

outside of the acqi program. This display mode may cause the displayed FID to exceed the displayed ADC limits in acqi by as much as a factor of the square root of 2.

See also: NMR Spectroscopy User Guide

Related; acqi Interactive acquisition display process (C)

av Set absolute-value mode in directly detected dimension (C)
gf Prepare parameters for FID/spectrum display in acqi (M)

### dmm Decoupler modulation mode for first decoupler (P)

Description: Sets the modulation modes for the first decoupler. In the standard two-pulse

sequence, dmm typically has a single state because the decoupler modulation is normally not changed during the pulse sequence, but this is not fixed. For example, dmm='ccw' gives single-frequency CW decoupling during the first part of the sequence and WALTZ-16 decoupling during acquisition.

In pulse sequences using the decoupler for pulsing (INEPT, DEPT, HETCOR, etc.), decoupler modulation must be set to 'c' during periods of the pulse sequence when the decoupler is to be pulsed.

Values: 'c', 'f', 'g', 'm', 'p', 'r', 'u', 'w', and 'x' are available;:

• 'c' sets continuous wave (CW) modulation.

• 'f' sets fm-fm modulation (swept-square wave).

• 'g' sets GARP modulation.

• 'm' sets MLEV-16 modulation.

• 'n' sets noise modulation.

• 'p' sets programmable pulse modulation using the dseq parameter to specify the decoupling sequence.

• 'r' sets square-wave modulation.

• 'u' sets user-supplied modulation using external hardware.

• 'w' sets WALTZ-16 modulation.

• 'x' sets XY32 modulation.

See also: NMR Spectroscopy User Guide

Related: dm Decoupler mode for first decoupler (P)

dmfDecoupler modulation frequency for first decoupler (P)dmm2Decoupler modulation mode for second decoupler (P)dmm3Decoupler modulation mode for third decoupler (P)dmm4Decoupler modulation mode for fourth decoupler (P)dseqDecoupler sequence for the first decoupler (P)

## dmm2 Decoupler modulation mode for second decoupler (P)

Applicability: Systems with a second decoupler.

Description: Sets the type of decoupler modulation for the second decoupler during different

status periods within a pulse sequence. It functions analogously to dmm.

Values: 'c', 'f', 'g', 'm', 'p', 'r', 'u', 'w', and 'x' are available. Refer to

dmm for the definition of these values (note that if the mode 'p' is selected, dseq2 specifies the decoupling sequence). If dn2='' (two single quotes) and a second decoupler is present in the console (numrfch greater than 2),

dmm2 is internally set to 'c' when go is executed.

Related: dm2 Decoupler modulation for the second decoupler (P)

dmf2 Decoupler modulation frequency for the second decoupler (P)

dmm Decoupler modulation mode for first decoupler (P)

dn2 Nucleus for the second decoupler (P)

dseq2 Decoupler sequence for the second decoupler (P)

numrfch Number of rf channels (P)

# dmm3 Decoupler modulation mode for third decoupler (P)

Applicability: Systems with a third decoupler.

Description: Sets type of decoupler modulation for the third decoupler during different status

periods within a pulse sequence. It functions analogously to dmm.

Values: 'c', 'f', 'g', 'm', 'p', 'r', 'u', 'w', and 'x' are available. Refer to

dmm for the definition of these values (note that if the mode 'p' is selected, dseq3 specifies the decoupling sequence). If dn3='' (two single quotes) and a third decoupler is present in the console (numrfch equal to 4), dmm3 is

internally set to 'c' when go is executed.

See also: NMR Spectroscopy User Guide

Related: dm3 Decoupler modulation for third decoupler (P)

dmf 3 Decoupler modulation frequency for third decoupler (P)
dmm Decoupler modulation mode for first decoupler (P)

dn3 Nucleus for the third decoupler (P)

dseq3 Decoupler sequence for the third decoupler (P)

numrfch Number of rf channels (P)

## dmm4 Decoupler modulation mode for fourth decoupler (P)

Applicability: Systems with a deuterium decoupler channel as the fourth decoupler.

Description: Sets type of decoupler modulation for the fourth decoupler during different

status periods within a pulse sequence. It functions analogously to dmm.

Values: 'c', 'f', 'g', 'm', 'r', 'u', 'w', and 'x' are available. Refer to dmm for

the definition of these values. If dn4='' (two single quotes) and a fourth decoupler is present in the console (numrfch greater than 4), dmm4 is

internally set to 'c' when go is executed.

See also: NMR Spectroscopy User Guide

Related: dm4 Decoupler modulation for the fourth decoupler (P)

dmf4 Decoupler modulation frequency for the fourth decoupler (P)

dmm Decoupler modulation mode for first decoupler (P)

dn4 Nucleus for the fourth decoupler (P)

dseq4 Decoupler sequence for the fourth decoupler (P)

numrfch Number of rf channels (P)

## dn Nucleus for first decoupler (P)

Description: Changing the value of dn causes a macro (named \_dn) to be executed that

extracts values for dfrq and dof from lookup tables. The tables, stored in the

directory /vnmr/nuctables, are coded by atomic weights.

Values: In the lookup tables, typically 'H1', 'C13', 'P31', etc.

Related: dfrq Transmitter frequency of first decoupler (P)

dn2
 dn3
 Nucleus for second decoupler (P)
 dn4
 Nucleus for third decoupler (P)
 dof
 Frequency offset for first decoupler (C)
 tn
 Nucleus for observe transmitter (P)

## dn2 Nucleus for second decoupler (P)

Applicability: Systems with a second decoupler.

Description: Changing the value of dn2 causes a macro (named \_dn2) to be executed that

extracts values for dfrq2 and dof2 from lookup tables. Otherwise, dn2 functions analogously to the parameters tn and dn. If an experiment does not use the second decoupler channel, the channel can be disabled by setting dn2='' (two single quotes with no space in between). This sets dm2='n', dmm2='c', dmf2=1000 (in Hz), dfrq2=1 (in MHz), dof2=0, dpwr2=0,

dseq2='', and dres2=1.

See also: NMR Spectroscopy User Guide

Related: dfrq2 Transmitter frequency of second decoupler (P)

dn Nucleus for first decoupler (P)

dof2 Frequency offset for second decoupler (C)

numrfch Number of rf channels (P)

Nucleus for observe transmitter (P)

# dn3 Nucleus for third decoupler (P)

Applicability: Systems with a third decoupler.

Description: Changing the value of dn3 causes a macro (named \_dn3) to be executed that

extracts values for dfrq3 and dof3 from lookup tables. Otherwise, dn3 functions analogously to the parameters tn and dn. If an experiment does not use the third decoupler channel, the channel can be disabled by setting dn3='' (two single quotes with no space in between). This sets dm3='n', dmm3='c', dmf3=1000 (in Hz), dfrq3=1 (in MHz), dof3=0, dpwr3=0, dseq3='',

and dres3=1.

See also: NMR Spectroscopy User Guide

Related: dn Nucleus for first decoupler (P)

dfrq3 Transmitter frequency of third decoupler (P)
dof3 Frequency offset for third decoupler (C)

numrfch Number of rf channels (P)

Nucleus for observe transmitter (P)

# dn4 Nucleus for fourth decoupler (P)

Applicability: Systems with a deuterium decoupler channel as the fourth decoupler.

Description: Changing the value of dn4 causes a macro (named dn4) to be executed that

extracts values for dfrq4 and dof4 from lookup tables. Otherwise, dn4 functions analogously to the parameters tn and dn except that the only valid value for dn4 is 'H2'. If an experiment does not use the fourth decoupler channel, the channel can be disabled by setting dn4='' (two single quotes with no space in between). This sets dm4='n', dmm4='c', dmf4=1000 (in Hz),

dfrq4=1 (in MHz), dof4=0, dpwr4=0, dseq4='', and dres4=1.

Related: dfrq4 Transmitter frequency of fourth decoupler (P)

dn Nucleus for first decoupler (P)

dof4 Frequency offset for fourth decoupler (C)

numrfch Number of rf channels (P)

Nucleus for observe transmitter (P)

## dndfid Retrieve and process fid data from the locator (M)

Applicability: Liquids, Imaging, Solids

Description: Retrieve fid data from an item selected in the locator. Data is also processed if

Process data on drag-and-drop from locator is selected in the System settings

dialog in the Utilities menu.

Related: dndjoin Join a work space from the locator (M)

dndparRetrieve a parameter set from the locator (M)dndshimsRetrieve a shimset set from the locator (M)

locaction Locator action (M)

locprotoexecExecute a protocol from the locator (M)xmmakenodeMake a new study queue node (M)

## dndjoin Join a work space from the locator (M)

Description: Join the work space selected by the locator.

Related: dndfid Retrieve and process fid data from the locator (M)

dndparRetrieve a parameter set from the locator (M)dndshimsRetrieve a shimset set from the locator (M)

locaction Locator action (M)

locprotoexec Execute a protocol from the locator (M)

xmmakenode Make a new study queue node (M)

# dndpar Retrieve a parameter set from the locator (M)

Description: Retrieve a parameter set selected by the locator.

Related: dndfid Retrieve and process fid data from the locator (M)

dndjoin

Join a work space from the locator (M)

dndshims

Retrieve a shimset set from the locator (M)

locaction Locator action (M)

locprotoexecExecute a protocol from the locator (M)xmmakenodeMake a new study queue node (M)

## dndshims Retrieve a shimset set from the locator (M)

Description: Retrieve a shimset set selected by the locator.

Related: dndfid Retrieve and process fid data from the locator (M)

dndjoinJoin a work space from the locator (M)dndparRetrieve a parameter set from the locator (M)

locaction Locator action (M)

locprotoexec Execute a protocol from the locator (M) xmmakenode Make a new study queue node (M) D

dnode Display list of valid limNET nodes (M,U)

Applicability: Systems with limNET.

Description: Displays the contents of the user's limNET node database (i.e., all remote nodes

available to limNET). Each node is listed by name, Ethernet address (6

hexadecimal bytes), and burst size

See also: NMR Spectroscopy User Guide

Related: eaddr Display Ethernet address (M,U)

doautodialog Start a dialog window using def file (M)

Applicability: Systems with automation.

Syntax: doautodialog

Description: Internal macro used by enter to start a dialog window using the def file for

an experiment in the dialoglib directory.

Related: enter Enter sample information for automation run (M,U)

dodialog Start a dialog window with dialoglib file (M)

Syntax: dodialog

Description: Internal macro that starts a dialog window using a dialog file in the

dialoglib directory.

dof Frequency offset for first decoupler (P)

Description: Controls the frequency offset of the first decoupler. Higher numbers move the

decoupler to higher frequency (toward the left side of the spectrum). The frequency accuracy of the decoupler offset is generally 0.1 Hz. The value is

specified in the config program.

Values: -100000 to 100000 Hz (approximate, depends on frequency), in steps of 0.1 Hz.

See also: NMR Spectroscopy User Guide

Related: config Display current configuration and possible change it (M)

dof2 Frequency offset for second decoupler (P)
dof3 Frequency offset for third decoupler (P)
dof4 Frequency offset for fourth decoupler (P)
tof Frequency offset for observe transmitter (P)

dof2 Frequency offset for second decoupler (P)

Applicability: Systems with a second decoupler.

Description: Controls the frequency offset for the second decoupler. dof2 functions

analogously to the parameters tof and dof.

Values: −100000 to 100000 Hz (approximate, depends on frequency), in steps of 0.1 Hz.

If dn2=' ' (two single quotes with no space in between) and a second decoupler channel is present in the console, dof2 assumes a default value of 0

when go is executed.

See also: NMR Spectroscopy User Guide

Related: dn2 Nucleus for second decoupler (P)

dof Frequency offset for first decoupler (P)
tof Frequency offset for observe transmitter (P)

dof3 Frequency offset for third decoupler (P)

Applicability: Systems with a third decoupler.

Description: Controls the frequency offset for the third decoupler. dof3 functions

analogously to the parameters tof and dof.

Values: -100000 to 100000 Hz (approximate, depends on frequency), in steps of 0.1 Hz.

If dn3='' (two single quotes with no space in between) and a third decoupler channel is present in the console, dof3 assumes a default value of 0 when go

is executed.

See also: NMR Spectroscopy User Guide

Related: dn3 Nucleus for third decoupler (P)

dof Frequency offset for first decoupler (P)
tof Frequency offset for observe transmitter (P)

dof4 Frequency offset for fourth decoupler (P)

Applicability: Systems with a deuterium decoupler channel as the fourth decoupler.

Description: Controls the frequency offset for the fourth decoupler. dof4 functions

analogously to the parameters tof and dof.

Values: -100000 to 100000 Hz (approximate, depends on frequency), in steps of 2.384

Hz. If dn4='' (two single quotes with no space in between) and a fourth decoupler channel is present in the console, dof4 assumes a default value of 0

when go is executed.

See also: NMR Spectroscopy User Guide

Related: dn4 Nucleus for fourth decoupler (P)

dof Frequency offset for first decoupler (P)
tof Frequency offset for observe transmitter (P)

Doneshot Set up parameters for Doneshot pulse sequence (M)

Description: Converts a parameter set to Doneshot experiment.

See also: NMR Spectroscopy User Guide

Related: dosy Process DOSY experiments (M)

fiddle Perform reference deconvolution (M)

setup\_dosy Set up gradient levels for DOSY experiments (M)

dopardialog Start a dialog with dialoglib/experiment def file (M)

Description: Internal macro that starts a dialog window using a def file in the directory

dialoglib/experiment.

do pcss Calculate proton chemical shifts spectrum (C)

Syntax: do pcss<(<threshold><, max cc><, max width)>

Description: Strips a high-resolution proton spectrum down to a list of chemical shifts. The

list is saved in the file pcss.outpar. If no argument is given, do\_pcss automatically calculates the threshold and uses default values for the maximum allowable coupling constant and the maximum width of a spin multiplet.

Arguments: threshold sets the level whether a point belongs to a peak or is noise.

max cc is the maximum allowable coupling constant in the spectrum. Default

is 20 Hz.

max\_width is the maximum width of a spin multiplet in the spectrum.

Default is 60 Hz.

Examples: do pcss

do\_pcss(10)
do pcss(9,20,80)

See also: NMR Spectroscopy User Guide

Related: pcss Calculate and show proton chemical shifts spectrum (M)

# dosy Process DOSY experiments (M)

Syntax: dosy(<'prune'>,<lowerlimit,upperlimit>)

Description: Performs a DOSY (diffusion ordered spectroscopy) analysis of the data in an

array of spectra.

dosy uses the commands dll and fp to determine the heights of all signals above the threshold defined by the parameter th and then fits the decay curve for each signal to a Gaussian using the program dosyfit. It stores a summary of all diffusion coefficients and their estimated standard errors and various other results as follows:

• In the directory \$HOME/vnmrsys/Dosy: diffusion\_display.inp, general\_dosy\_stats, calibrated\_gradients, fit\_errors, and diffusion\_spectrum

• In the current experiment: a second copy of diffusion\_display.inp.

The command showdosy has been incorporated into dosy.

Arguments: prune starts a dialog to allow one or more spectra to be omitted from the

analysis.

lowerlimit is the lower diffusion limit (in units of  $10^{-10}$  m<sup>2</sup>/s) to be

displayed.

upperlimit is the upper diffusion limit (in units of  $10^{-10}$  m<sup>2</sup>/s) to be

displayed.

Without arguments, dosy uses all the experimental spectra and covers the

whole diffusion range seen in the experimental peaks.

See also: NMR Spectroscopy User Guide

Related: ddif Synthesize and display DOSY plot (C)

fiddle Perform reference deconvolution (M)

setup\_dosy Set up gradient levels for DOSY experiments (M)

# dosy2d Apptype macro for dosy 2D experiments (M)

Applicability: Liquids

Description: Performs the actions for 2D dosy protocols to set up, process, and plot

experiments. It is only available if the Dosy software is installed.

Related: apptype Application type (PM)

execpars Set up the exec parameters (M)

# dosyfrq Larmor frequency of phase encoded nucleus in DOSY (P)

Description: Stores the NMR frequency of the phase encoded nucleus in DOSY experiments.

It is directly set by the DOSY sequences.

See also: NMR Spectroscopy User Guide

Related: dosy Process DOSY experiments (M)

dosygamma Gyromagnetic constant of phase encoded nucleus in DOSY (P)

Description: Stores the gyromagnetic constant of the phase encoded nucleus in DOSY

experiments. It is automatically set by the DOSY sequences and used by the

dosy macro.

See also: NMR Spectroscopy User Guide

Related: dosy Process DOSY experiments (M)

# dosytimecubed Gyromagnetic constant of phase encoded nucleus in DOSY (P)

Description: Time cubed factor in the expression for diffusional attenuation. It is

automatically set by the DOSY sequences and used by the dosy macro.

See also: NMR Spectroscopy User Guide

Related: dosy Process DOSY experiments (M)

### dot1 Set up a $T_1$ experiment (M)

Syntax: dot1<(min T1 estimate, max T1 estimate, time) >

Description: Sets up all parameters to perform a  $T_1$  experiment, including d1, pw, p1, nt,

and an array of d2 values, based on information entered you enter. Make sure that the parameter pw90 is set properly and contains the correctly calibrated 90° pulse width because dot1 uses this information. If you have not done a pulse

width calibration recently, you may wish to do so now.

Minimum and maximum  $T_1$  for the peaks of interest are estimates. Do the best you can. Your estimates are used to select optimum values of d2. If the  $T_1$  does not fall between your two guesses, your experiment may not be optimum, but it should still be usable unless your estimates are extremely far off. When you are

satisfied with the parameters, enter ga or au to acquire the data.

Arguments:  $min_T1_estimate$  is the estimated minimum expected  $T_1$ . The default is

the system prompts the user for the value.

max T1 estimate is the estimated maximum expected  $T_1$ . The default is

the system prompts the user for the value.

time is the total time in hours that the experiment should take. The default is

the system prompts the user for the value.

Examples: dot1

dot1(1,2,.5)

See also: NMR Spectroscopy User Guide

Related: d1 First delay (P)

d2 Incremented delay in 1st indirectly detected dimension (P)
ga Submit experiment to acquisition and FT the result (C)

go Submit experiment to acquisition (C)

nt Number of transients (P)
p1 First pulse width (P)
pw Pulse width (P)
pw90 90° pulse width (P)

#### dotflag Display FID as connected dots (P)

Description: When sparse FID data points are displayed, they are displayed as unconnected

dots. If dotflag exists and is set to 'n', the FID dots will be connected. To create dotflag, enter create ('dotflag', 'flag'). To create dotflag and the FID display parameters axisf, vpf, vpfi, crf, and

deltaf (if the parameter set is older and lacks these parameters), enter

addpar('fid').

Values: 'n' sets connecting the dots. 'y' sets not connecting the dots.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

**create** Create new parameter in a parameter tree (C)

df Display a single FID (C)

# downsamp Downsampling factor applied after digital filtering (P)

Description: Specifies the downsampling factor applied after digital filtering. The spectral

width of the data set after digital filtering and downsampling is sw divided by downsamp, where sw is the acquired spectral width. If downsamp does not exist in the current experiment, enter addpar ('downsamp') to add it. addpar ('downsamp') creates the digital filtering and downsampling parameters downsamp, dscoef, dsfb, dslsfrq, and filtfile.

Values: Number for the downsampling factor. 1 sets digital filtering with a filter

bandwidth specified by dsfb without downsampling.

'n' sets normal data processing without digital filtering.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to current experiment (M)

digfiltWrite digitally filtered FID to another experiment (M)dscoefDigital filter coefficients for downsampling (P)dsfbDigital filter bandwidth for downsampling (P)dslsfrqBandpass filter offset for downsampling (P)filtfileFile of FIR digital filter coefficients (P)

pards Create additional parameters used by downsampling (M)

Spectral width in directly detected dimension (P)

## dp Double precision (P)

Description: Sets whether data are acquired in a 16-bit or 32-bit integer format.

Values: 'n' sets 16-bit format, 'y' sets 32-bit format. If the 200-kHz receiver option is

installed (Max. Narrowband Width set to 200 kHz in the Spectrometer Configuration window), dp is forced to 'n' if 120000<sw<=200000. If sw>200000, dp is forced to 'y'. On wideline systems, dp='y' is required

when sw>100000.

See also: NMR Spectroscopy User Guide

Related: sw Spectral width in directly detected dimension (P)

# dpcon Display plotted contours (C)

Syntax: dpcon(<options, ><levels, spacing>)

Description: Produces a true contour plot display.

Arguments: options must precede levels and spacing in the argument list and can

be one or more of the following:

• 'pos' is a keyword to limit the display to positive peaks only in phased spectra. The default is both positive and negative peaks.

• 'neg' is a keyword to limit the display to negative peaks only in phased spectra.

 'noaxis' is a keyword to omit outlining the display and drawing the horizontal or vertical axis.

levels is the maximum number of contours to be shown. The default is 4. spacing is the spacing by relative intensity of successive contour levels. The default is 2.

Examples: dpcon

dpcon('pos',6)
dpcon(15,1.4)

See also: NMR Spectroscopy User Guide

Related: dcon Display noninteractive color intensity map (C)

dconi Control display selection for the dconi program (P)
dpconn Display plotted contours without screen erase (C)

pcon Plot contours on plotter (C)

# dpconn Display plotted contours without screen erase (C)

Syntax: dpconn(<options,><levels,spacing>)

Description: Produces a true contour plot display exactly the same as the dpcon command,

but without erasing the screen before drawing. The arguments are entered the

same as dpcon.

See also: NMR Spectroscopy User Guide

Related: dpcon Display plotted contours (C)

# dpf Display peak frequencies over spectrum (C)

Description:

Displays peak frequencies in the graphics window, with units specified by the axis parameter. Only those peaks greater than th high are selected. If the interactive command ds is active, dpf deactivates it.

Two basic modes of label positioning are available: labels placed at the top, with *long leaders* extending down to the tops of the lines (syntax 1 using 'top' keyword) or labels positioned just above each peak, with *short leaders* (syntax 2 using 'leader' keyword). The default is short leaders.

Arguments: 'noll' is a keyword to display frequencies using last previous line listing.

'pos' (or 'noneq') is a keyword to display positive peaks only.

noise\_mult is a numerical value that determines the number of noise peaks displayed for broad, noisy peaks. The default is 3. A smaller value results in more peaks, a larger value results in fewer peaks, and a value of 0.0 results in a line listing containing all peaks above the threshold th. Negative values of noise\_mult are changed to a value of 3. The noise\_mult argument is inactive when the 'noll' keyword is specified.

'top' is a keyword to display peak labels at the top with long leaders. In this mode, the height of labels is varied by changing the parameter wc2.

'leader' is a keyword to display labels positioned just above each peak.

length specifies the leader length, in mm, if labels are positioned just above each peak. The default is 20.

Examples: dpf('pos')

dpf('leader',30)

dpf('top','noll')

dpf('pos',0.0,'leader',30)

See also: NMR Spectroscopy User Guide

Related: axis Axis label for displays and plots (P)

dpir Display integral amplitudes below spectrum (C)

dpirn Display normalized integral amplitudes below spectrum (M)

pir Plot integral amplitudes below spectrum (C)

Plot normalized integral amplitudes below spectrum (M)

ppf Plot peak frequencies over spectrum (M)

th Threshold (P)

vp Vertical position of spectrum (P)
wc2 Width of chart in second direction (P)

# dpir Display integral amplitudes below spectrum (C)

Description: Displays integral amplitudes below the appropriate spectral regions.

See also: NMR Spectroscopy User Guide

Related: dpf Display peak frequencies over spectrum (C)

dpirn Display normalized integral amplitudes below spectrum (M)

pir Plot integral amplitudes below spectrum (C)

pirn Plot normalized integral amplitudes below spectrum (M)

ppf Plot peak frequencies over spectrum (M)

## dpirn Display normalized integral amplitudes below spectrum (M)

Description: Equivalent to the command dpir except that the sum of the integrals is

normalized to the value of the parameter ins.

See also: NMR Spectroscopy User Guide

Related: dpir Display integral amplitudes below spectrum (C)

ins Integral normalization scale (P)

pirn Plot normalized integral amplitudes below spectrum (M)

# dpiv Display integral values below spectrum (M)

Syntax: dpiv<(vertical position)>

Description: Labels integrals with a bracket below the spectrum and a vertical number

indicating the integral value.

vertical labels for narrower regions

• avoids label overlap by label shifting

• more flexible vertical positioning

The vertical position defaults to a location just underneath the scale labels, assuming there is enough room below the scale. If the vertical position is too low, the vertical position is allowed to approach the position of the spectrum up to 1 mm. If the spectral position is so low that the integral labels would overlap with the spectrum, an error message is produced (indicating the minimum vp), and the command aborts. No error message is produced in case of overlap with the scale. The minimum for vp depends on the plotter and the character size, and in the case of dpiv also on the size of the graphics window.

Use an optional argument to force the vertical position to any value; no checking is done, and no error message is produced in case of overlap. piv(vp-2) produces integral labels with the brackets ending 2 mm below the position of the spectrum.

dpiv follows this convention: the output is controlled by ins and insref and not by is. Restore the is integration mode by creating a (local or global)

parameter oldint and set oldint= 'y':

create('oldint','flag','global')

oldint='y'

oldint='n' (or destroy the parameter) switches back to the default

integration mode.

Examples: vp=25 dpiv

vp=50 pl pscale piv(0)

Related: dpir Display integral amplitudes below spectrum (C)

dpirn Display normalized integral amplitudes below spectrum (C)
dpivn Display normalized integral amplitudes below spectrum (M)
pirn Plot normalized integral amplitudes below spectrum (C)

Plot integral amplitudes below spectrum (C)

Plot integral amplitudes below spectrum (M)

Plot normalized integral amplitudes below spectrum (M)

# dpivn Display normalized integral values below spectrum (M)

Syntax: dpivn<(vertical position)>

Description: Labels integrals with a bracket below the spectrum and a vertical number

indicating the integral value.

See dpiv for description and use.

Related: dpir Display integral amplitudes below spectrum (C)

dpirn Display normalized integral amplitudes below spectrum (C)

dpiv Display integral amplitudes below spectrum (M)

Plot normalized integral amplitudes below spectrum (C)

Plot integral amplitudes below spectrum (C)

Plot integral amplitudes below spectrum (M)

Plot normalized integral amplitudes below spectrum (M)

## dpl Default plot (M)

Description: Looks for sequence-specific default plot macro (dpl seqfil) and executes

if one is found.

Related:  $dpl_seqfil$  Sequence-specific default plot (M)

dpr Default process (M)
dds Default display (M)

# dpl seqfil Sequence-specific default plot (M)

Description: Sequence-specific default plot. These macros are called by the dpl macro.

Examples: dpl\_NOESY1D

dpl TOCSY1D

Related: dp1 Default plot (M)

dpr Default process (M)
dds Default display (M)

dplane Display a 3D plane (M)

Syntax: dplane(<plane type, >plane number)

Description: Displays the 2D color map of a particular data plane from a 3D spectral data set.

The 3D parameters are loaded into VnmrJ each time dplane is executed. The parameter path3d specifies the absolute path to the directory (without the .extr file extension) where the 2D planes extracted from the 3D spectral

data set reside.

Arguments: plane type is one of the keywords 'f1f3', 'f2f3', and 'f1f2' for the

 $f_1f_3$ ,  $f_2$ ,  $f_3$ , and  $f_1f_2$  planes, respectively. If plane\_type is specified, the parameter plane is updated with that new value. plane is then used to

determine the type of 3D plane to be displayed.

plane number specifies which plane of a particular type is to be displayed:

• For plane f<sub>1</sub>f<sub>3</sub>, the range of plane\_number is 1 to fn2/2

• For plane  $f_2f_3$ , the range of plane\_number is 1 to fn1/2

• For plane  $f_1f_2$ , the range of plane\_number is 1 to fn/2

Examples: dplane(3)

dplane('f1f2',2)

See also: NMR Spectroscopy User Guide

Related: dsplanes Display a series of 3D planes (M)

dproj Display a 3D plane projection (M)

getplane Extract planes from a 3D spectral data set (M)

nextpl Display the next 3D plane (M)

path3d Path to currently displayed 2D planes from a 3D data set (P)

plane Currently displayed 3D plane type (P)
prevpl Display the previous 3D plane (M)
plplanes Plot a series of 3D planes (M)

dpr Default process (M)

Description: Looks for sequence-specific default plot macro (dpr\_seqfil) and executes

if one is found.

Related:  $dpr_seqfil$  Sequence-specific default process (M)

dpl Default plot (M)
dds Default display (M)

dpr seqfil Sequence-specific default process (M)

Description: Sequence-specific default plot. These macros are called by the dpr macro.

Examples: dpr\_NOESY1D

dpr TOCSY1D

Related: dpr Default process (M)

dpl Default plot (M)
dds Default display (M)

dprofile Display pulse excitation profile (M)

Syntax: dprofile<(axisflag<,profile<,shapefile>>)>

Description: Displays the X, Y and Z excitation (inversion) profile for a pulse shape

generated by the Pbox software. If shapefile is not provided, the last simulation data stored in the shapelib/pbox.sim file are displayed.

Arguments: The axisflaq and profile arguments can be given in any order.

axisflag is 'y' to display the full spectrum and a frequency scale, or 'n'

to suppress the scale and spectrum. The default is 'n'.

profile is a character string identifying the desired profile. 'xyz' selects X, Y, and Z (inversion) profiles; 'xy' selects only the excitation (transverse) profiles; 'x' selects only the X transverse excitation profile; and 'z' selects only the inversion profile. The default is transverse.

only the inversion profile. The default is 'xyz'.

shapefile is the name of a \*.RF or \*.DEC file, including the extension.

Examples: dprofile

dprofile('y','xy')

dprofile('xy','n','softpls.RF')

See also: NMR Spectroscopy User Guide

Related: pprofile Plot pulse excitation profile (M)

Pbox Pulse shaping software (U)

# dproj Display a 3D plane projection (M)

Syntax: dproj < (plane\_type) >

Description: Displays 2D color map of the 2D projection plane from a 3D spectral data set.

The projection is a skyline projection. The 3D parameters are loaded into VnmrJ each time dproj is executed. For this macro, the parameter path3d specifies the directory (without the .extr extension) where the 2D projection resides

that has been created from the 3D spectral data set.

Arguments: plane type is one of the keywords 'f1f3', 'f2f3', and 'f1f2' for the

 $f_1f_3$ ,  $f_2$ ,  $f_3$ , and  $f_1f_2$  planes, respectively. If plane\_type is specified, the parameter plane is updated with that value. plane is then used to determine

the type of 2D projection to be displayed.

Examples: dproj

dproj('f1f2')

See also: NMR Spectroscopy User Guide

Related: dplane Display a 3D plane (M)

dsplanes Display a series of 3D planes (M)

getplane Extract planes from a 3D spectral data set (M)

nextpl Display the next 3D plane (M)

path3d Path to currently displayed 2D planes from a 3D data set (P)

plane Currently displayed 3D plane type (P)
plplanes Plot a series of 3D planes (M)
prevpl Display the previous 3D plane (M)

# dps Display pulse sequence (C)

Syntax: dps<(file),x,y,width,height>

Description: Displays a picture of pulse sequences consisting of three to five parts. The top

part is the transmitter pulse sequence (Tx). The second part is the decoupler pulse sequence (Dec). The third part might be the second or third decoupler (Dec2 or Dec3) pulse sequence or gradients (X, Y, or Z), depending on the

program. The lowest part is the status.

The pulse parameters are displayed if there is enough space an if the length of the parameter name is less than thirty letters. The value of each pulse is also displayed. If the value delay or width is less than zero, a question mark (?) is displayed. The time units are displayed in color (on a color monitor). The height of pulses is scaled according to their power level.

dps also displays spin lock, transmitter gating, observe transmitter power, and

other information.

Arguments: file specifies the name of the file containing the pulse sequences. The default

is the file seqfil.

x, y specifies the start of the position with respect to the lower-left corner of the

window.

width, height are in proportion to wcmax and wc2max.

See also: NMR Spectroscopy User Guide

Related: pps Plot pulse sequence (C)

seqfil Pulse sequence name (P)
wc Width of chart (P)

wcmax Maximum width of chart (P)

wc2max Maximum width of chart in second direction (P)

# dpwr Power level for first decoupler with linear amplifier (P)

Applicability: Systems with a linear amplifier.

Description: On systems equipped with a linear amplifier, a 63-dB or 79-dB attenuator

between the decoupler transmitter and the amplifier controls the power level.

The system value for the attenuator upper safety limit is set fin the Spectrometer Configuration window (opened by config). The Upper Limit entry sets this value. For broadband decoupling of <sup>1</sup>H nuclei, typical values range from 36 to 49 dB. For homonuclear decoupling, typical values range from 5 to 15 dB.

Values: 79 dB, -16 to +63, in steps of 1 dB.

Decoupler power greater than 2 watts in a switchable probe will damage the probe. Always carefully calibrate decoupling to avoid exceeding 2 watts. The maximum value for dpwr on a 200-, 300-, or 400-MHz system with a linear amplifier on the decoupler channel has been set to 49, corresponding to about 2 watts of power. Before using dpwr=49 for continuous decoupling, ensure safe operation by measuring the output power. This should be done during system

installation and checked periodically by the user.

See also: VnmrJ Installation and Administration

Related: cattn Coarse attenuator (P)

config Display current configuration and possible change it (M)

dpwrfFirst decoupler fine power (P)dpwr2Power level for second decoupler (P)dpwr3Power level for third decoupler (P)dpwr4Power level for fourth decoupler (P)

fattn Fine attenuator (P)

Power level of observe transmitter with linear amplifiers (P)

**tpwrf** Observe transmitter fine power (P)

# dpwr2 Power level for second decoupler with linear amplifier (P)

Applicability: Systems with a linear amplifier as the second decoupler.

Description: Controls the coarse attenuator (63 dB or 79 dB) that resides between the

transmitter board and the linear amplifier associated with the second decoupler.

The system value for the attenuator upper safety limit is set in the Spectrometer Configuration window (opened by config).

Values: 79 dB, -16 to +63, in steps of 1 dB.

If dn2=' ' (two single quotes) and a second decoupler channel is present in the

console, dpwr2 assumes a default value of 0 when go is executed.

CAUTION: Decoupler power greater than 2 watts in a switchable probe will

damage the probe. Always carefully calibrate decoupling to avoid exceeding 2 watts. The maximum value for dpwr2 on a 200-, 300-, or 400-MHz system with a linear amplifier on the decoupler channel has been set to 49, corresponding to about 2 watts of power. Before using dpwr2=49 for continuous decoupling, ensure safe operation by measuring the output power. This should be done during system

installation and checked periodically by the user.

NMR Spectroscopy User Guide See also:

Related: cattn Coarse attenuator type (P)

> Display current configuration and possible change it (M) config

Nucleus for second decoupler (P)

dpwr3 Power level for third decoupler with linear amplifier (P)

Applicability: Systems with a linear amplifier as the third decoupler.

Description: Controls the coarse attenuator (63 dB or 79 dB) that resides between the

> transmitter board and the linear amplifier associated with the third decoupler. The system value for the attenuator upper safety limit is set in the Spectrometer

Configuration window (opened by config).

If 63-dB attenuator installed: 0 to 63 (63 is max. power), in units of dB. Values:

If 79-dB attenuator installed: -16 to 63 (63 is max. power), in units of dB. If dn3=' ' (two single quotes) and a third decoupler channel is present in the

console, dpwr3 assumes a default value of 0 when go is executed.

CAUTION: Decoupler power greater than 2 watts in a switchable probe will

damage the probe. Always carefully calibrate decoupling to avoid exceeding 2 watts. The maximum value for dpwr3 on a 200-, 300-, or 400-MHz system with a linear amplifier on the decoupler channel has been set to 49, corresponding to about 2 watts of power. Before using dpwr3=49 for continuous decoupling, ensure safe operation by measuring the output power. This should be done during system

installation and checked periodically by the user.

NMR Spectroscopy User Guide Related: cattn Coarse attenuator type (P)

> Display current configuration and possible change it (M) config

dn3 Nucleus for third decoupler (P)

Power level for fourth decoupler amplifier (P) dpwr4

Applicability: Systems with deuterium decoupler channel as the fourth decoupler.

Description: Controls the coarse attenuator (45 dB range) that resides on the Lock

> Transceiver board and the amplifier associated with the fourth decoupler. The system value for the attenuator upper safety limit is set in the Spectrometer

Configuration window (opened by config).

Values: 48-dB attenuator: 15 to 63 (63 is max. power), in units of dB.

If dn4=' ' (two single quotes) and a third decoupler channel is present in the

console, dpwr4 assumes a default value of 0 when go is executed.

See also:

**CAUTION:** Decoupling power greater than 5 watts applied to a triple-resonance

probe will damage the probe. The maximum value for dpwr4 is 63, corresponding to about 35 watts to the probe. A value of dpwr4 equal to 52 corresponds to about 5 watts and will produce approximately a 1 kHz decoupling field. Always carefully calibrate decoupling power to avoid exceeding 5 watts. Before using dpwr4=52 continuous decoupling, ensure safe operation by measuring the output power. Measurement should be taken during system installation and checked

periodically by the user.

See also: NMR Spectroscopy User Guide

Related: cattn Coarse attenuator type (P)

config Display current configuration and possible change it (M)

dn3 Nucleus for third decoupler (P)

dpwrf First decoupler fine power (P)

Applicability: Systems with an optional fine attenuator on the decoupler channel.

Description: Controls the first decouple fine attenuator. Systems with this attenuator are

designated within the Spectrometer Configuration window (opened by

config) by the status of the Fine Attenuator entry. The fine attenuator is linear

and spans 6 dB.

Values: 0 to 4095 (where 4095 is maximum power). If dpwrf does not exist in the

parameter table, a value of 4095 is assumed.

See also: User Programming, User Guide: Solids; CP/MAS Installation,

Related: config Display current configuration and possibly change it (M)

dpwr Power level for first decoupler with linear amplifiers (P)

dpwrf2 Second decoupler fine power (P)
dpwrf3 Third decoupler fine power (P)

dpwrm First decoupler linear modulator power (P)

fattn Fine attenuator (P)

Power level of observe transmitter with linear amplifiers (P)

tpwrf Transmitter fine power (P)

dpwrf2 Second decoupler fine power (P)

Applicability: Systems with an optional fine attenuator on the second decoupler channel.

Description: Controls the second decoupler fine attenuator, functioning analogously to

dpwrf.

Values: 0 to 4095 (where 4095 is maximum power). If dpwrf2 does not exist in the

parameter table, a value of 4095 is assumed.

See also: User Programming

Related: dpwrf First decoupler fine power (P)

dpwrf3 Third decoupler fine power (P)

Applicability: Systems with an optional fine attenuator on the third decoupler channel.

Description: Controls the third decoupler fine attenuator, functioning analogously to dpwrf.

Values: 0 to 4095 (where 4095 is maximum power). If dpwrf3 does not exist in the

parameter table, a value of 4095 is assumed.

See also: User Programming

Related: dpwrf First decoupler fine power (P)

dpwrm First decoupler linear modulator power (P)

Applicability: Systems with a first decoupler linear modulator.

The fine power control is linear and spans 0 to dpwr.

Values: 0 to 4095 (where 4095 is maximum power). If dpwrm does not exist in the

parameter table, a value of 4095 is assumed.

See also: User Programming; User Guide: Solids; CP/MAS Installation

Related: dpwrm2 Second decoupler linear modulator power (P)

dpwrm3 Third decoupler linear modulator power (P)
tpwrm Observe transmitter linear modulator power (P)

dpwrm2 Second decoupler linear modulator power (P)

Applicability: Systems with a second decoupler linear modulator.

Description: Controls the second decoupler linear modulator systems.

Values: 0 to 4095 (where 4095 is maximum power). If dpwrm2 does not exist in the

parameter table, a value of 4095 is assumed.

See also: User Programming

Related: dpwrm First decoupler linear modulator power (P)

dpwrm3 Third decoupler linear modulator power (P)

Applicability: Systems with a third decoupler linear modulator.

Description: Controls the third decoupler linear modulator systems.

Values: 0 to 4095 (where 4095 is maximum power). If dpwrm3 does not exist in the

parameter table, a value of 4095 is assumed.

See also: User Programming

Related: dpwrm First decoupler linear modulator power (P)

Dgcosy Convert the parameter to a DQCOSY experiment (M)

Description: Convert the parameter to a double-quantum filtered (DQCOSY) experiment

See also: NMR Spectroscopy User Guide

Related: cosyps Set up parameters for phase-sensitive COSY (M)

Cosy Set up parameters for COSY pulse sequence (M) relayh Set up parameters for COSY pulse sequence (M)

draw Draw line from current location to another location (C)

Syntax: draw(<'keywords'>x,y)

Description: Draws a line from the current location to the absolute location with coordinates

given by the arguments.

Arguments: 'keywords' identifies the output device ('qraphics' | 'plotter'),

drawing mode ('xor' | 'normal'), and drawing capability

('newovly'|'ovly'|'ovlyC').

• 'graphics'|'plotter' is a keyword for the output device. The default is 'plotter'. The output selected is passed to subsequent pen, move, or draw commands and remains active until a different output is

specified.

- 'xor', 'normal' is a keyword for the drawing mode when using the 'graphics' output device. The default is 'normal'. In the 'xor' mode, if a line is drawn such that one or more points of the line are in common with a previous 'xor' line, the common points are erased. In the normal mode, the common points remain. The mode selected is passed to subsequent draw, pen, and move commands and remains active until a different mode is specified.
- 'newovly', 'ovly', and 'ovlyC' are keywords that specify an interactive drawing capability that is slightly slower than the 'xor' mode but more consistent in color. 'newovly' clears any previous draws, boxes, and writes made with the 'ovly' modes and draws the figure. 'ovly' draws without clearing so that multisegment figures can be created. 'ovlyC' clears without drawing.

x, y are the absolute coordinates, in mm, of the endpoint of the line to be drawn. The range of x is 0 at the left edge of the chart and wcmax at the right edge. The range of y is -20 at the bottom of the chart and wc2max at the top.

Examples: draw('graphics','xor'.wcmax-sc,vp+th)

draw(wcmax-sc-wc\*(cr-delta-sp)/wp,wc2max)

See also: NMR Spectroscopy User Guide

Related: gin Return current mouse position and button values (C)

> Move to an absolute location (C) move Select a pen or color for drawing (C) pen wcmax

Maximum width of chart (P)

Maximum width of chart in second direction (P)) wc2max

#### Measure linewidth and digital resolution (C) dres

Syntax: dres<(<freq<,fractional height>>)> :linewidth,digital resolution

Description: Analyzes the line defined by the current cursor position for its linewidth (width

at half-height) and digital resolution.

Arguments: freq is the frequency of the line. The default is the parameter cr. This

overrides using the current cursor position as the frequency.

fractional height is the linewidth is measured at this height. linewidth is the value returned for the linewidth of the line.

digital resolution is the value returned for the digital resolution of the

line.

dres:\$width,\$res Examples:

dres(cr, 0.55)

See also: NMR Spectroscopy User Guide; User Programming

Related: Current cursor position (P)

Measure signal-to-noise (C) dsn

#### dres Tip-angle resolution for first decoupler (P)

Applicability: Systems with waveform generators.

Controls the tip-angle resolution to be used within a waveform generator Description:

> decoupling sequence on the first decoupler. The optimum value is a function of the decoupling sequence to be used: for WALTZ-16, dres=90.0; for

MLEV16-240, dres=30.0; and for GARP1, dres=1.0.

Values: 1.0 to 90.0, in units of degrees. In reality, dres can assume values as small of

0.7 (but no smaller) and can be specified in units of 0.1°. To use this capability,

change the limits of dres by using destroy ('dres')

create('dres','real') setlimit('dres',360,0.7,0.1).

Making corresponding changes within the fixpar macro ensures that dres

is created in the desired way with each new parameter set.

See also: NMR Spectroscopy User Guide

Related: dmfadj Adjust decoupler tip-angle resolution time (M)

dres2 Tip angle resolution for second decoupler (P)
dres3 Tip angle resolution for third decoupler (P)
fixpar Correct parameter characteristics in experiment (M)

#### dres2 Tip-angle resolution for second decoupler (P)

Applicability: Systems with waveform generators.

Description: Controls the tip-angle resolution to be used within a waveform generator

decoupling sequence on the second decoupler. The optimum value is a function of the decoupling sequence to be used: for WALTZ-16, dres2=90.0; for

MLEV16-240, dres2=30.0; and for GARP1, dres2=1.0.

Values: 1.0 to 90.0, in units of degrees. See also: *NMR Spectroscopy User Guide* 

Related: dmf2adj Adjust second decoupler tip-angle resolution time (M)

dres Tip-angle resolution for first decoupler (P)

# dres3 Tip-angle resolution for third decoupler (P)

Applicability: Systems with waveform generators.

Description: Controls the tip-angle resolution to be used within a waveform generator

decoupling sequence on the third decoupler. The optimum value is a function of the decoupling sequence to be used: for WALTZ-16, dres3=90.0; for

MLEV16-240, dres3=30.0; and for GARP1, dres3=1.0.

Values: 1.0 to 90.0, in units of degrees. See also: *NMR Spectroscopy User Guide* 

Related: dmf3adj Adjust third decoupler tip-angle resolution time (M)

dres Tip-angle resolution for first decoupler (P)

#### dres4 Tip-angle resolution for fourth decoupler (P)

Applicability: Systems with deuterium decoupler channel as the fourth decoupler.

Description: Controls the tip-angle resolution to be used for the decoupling sequence on the

fourth decoupler. The optimum value is a function of the decoupling sequence to be used: for WALTZ-16, dres4=90.0; for MLEV16-240, dres4=30.0;

and for GARP1, dres4=1.0.

Values: 1.0 to 90.0, in units of degrees. See also: *NMR Spectroscopy User Guide* 

Related: dmf4adj Adjust fourth decoupler tip-angle resolution time (M)

dres Tip-angle resolution for first decoupler (P)

D

# ds Display a spectrum (C)

```
Syntax: (1) ds<(index) >
     (2) ds<(options) >
```

Description: Displays a single spectrum. Parameter intmod controls integral display:

- intmod='off' turns off the integral display
- intmod='full' displays the entire integral
- intmod='partial' displays every other integral region

Parameter entry after a spectrum has been displayed with the ds command causes the spectrum to be updated.

Two additional parameters control the behavior of the ds command:

- The parameter <a href="phasing">phasing</a> (in the "global" parameter set) controls the percentage of the spectrum updated during interactive phasing. This parameter can be set in the range of 10 to 100. A value of 100 causes the entire spectrum to be updated. A value of 20 causes the area between the two horizontal cursors to be updated.
- The parameter lvltlt (in the "current" parameter set) controls the
  sensitivity of the interactive lvl and tlt adjustments. lvltlt can be set
  to any positive real number. It is basically a multiplier for the sensitivity.
  The default value is 1.0. Larger values make the adjustments larger. Smaller
  values make the adjustments smaller.

For arrayed 1D spectra or for 2D spectra, a particular trace can be viewed by supplying the index number as an argument. For 2D data sets, spectra can be displayed from either the  $f_1$  or  $f_2$  domain by setting the parameter trace equal to 'f1' or 'f2', respectively. After entering ft1d, interferograms can be viewed by setting trace='f1' and then typing ds.

Spectra are scaled according to the number of completed transients ct. If nt is arrayed (nt=1, 2, 4, 8), each spectrum is scaled by its own ct.

Arguments:

index (used with syntax 1) is the index number of a particular trace to be displayed in arrayed 1D spectra or in 2D spectra (syntax 1).

options (used with syntax 2) is any of the following keywords:

- 'toggle' switches between the box and the cursor modes.
- 'restart' redraws the cursor if it has been turned off.
- 'expand' toggles between expanded and full view of the spectrum.
- 'spwp' interactively adjusts start and width of the spectrum display.
- 'phase' enters an interactive phasing mode.
- 'thresh' interactively adjusts the threshold.
- 'z' interactively sets integral resets.
- 'dscale' toggles the scale below the spectrum on and off.
- 'lvltlt' interactively adjusts the lvl and tlt parameters.
- 'scwc' interactively adjusts the start and width of chart.
- 'noclear' start or restart the ds display without clearing the graphics screen
- 'exists' exit the ds display, leaving a non-interactive dss display.

Examples: ds

```
ds(7)
ds('restart')
```

See also: NMR Spectroscopy User Guide

Related: crmode Current state of cursors in dfid, ds, or dconi (P)

Ct Completed transients (P)

exists

ftld Fourier transform along f<sub>2</sub> dimension (C)

intmod Integral display mode (P)

First-order phase in directly detected dimension (P)

lvl Zero-order baseline correction (P)

lvltlt Control sensitivity of lvl and tlt adjustments (P)

nt Number of transients (P)

phasing Control update region during ds phasing (P)

rp Zero-order phase in directly detected dimension (P)

 ${\tt select} \qquad \qquad {\tt Select \ a \ spectrum \ without \ displaying \ It \ (C)}$ 

tlt First-order baseline correction (P)
trace Mode for n-dimensional data display (P)
wftld Weight and Fourier transform f<sub>2</sub> for 2D data (C)

# ds2d Display 2D spectra in whitewash mode (C)

Syntax: ds2d<(options)>

Description: Displays a stacked plot of 2D spectra in whitewash mode (after the first spectra,

each spectra is blanked out in regions in which it is behind an earlier spectra). Color does not represent intensity (unlike dcon), because intensity can be seen visually, but instead successive traces are displayed in different colors so that

color represents frequency.

Arguments: options can be any of the following keywords:

• 'nobase' is a keyword to activate the th parameter to suppress all intensity below the th level.

• 'fill' is a keyword to fill in the peaks. When using 'fill', th operates linearly and not logarithmically (factors of 2) as it does in the contour or color intensity displays.

• 'fillnb' is a keyword to combine base suppression and peak filling. When using 'fillnb', th operates linearly and not logarithmically (factors of 2) as it does in the contour or color intensity displays.

 'noaxis' is a keyword to omit outlining the display and drawing the horizontal and vertical axis.

Examples: ds2d

ds2d('fillnb')

See also: NMR Spectroscopy User Guide

Related: dcon Display noninteractive color intensity map (C)

dconi Control display selection for the dconi program (P)

ds2dn Display 2D spectra in whitewash mode without screen erase (C)

Plot 2D spectra in whitewash mode (C)

th Threshold (P)

## ds2dn Display 2D spectra in whitewash mode without screen erase (C)

Syntax: ds2dn<(options)>

Description: Displays a stacked plot of 2D spectra in whitewash mode (after the first spectra,

each spectra is blanked out in regions in which it is behind an earlier spectra) the same as ds2d but without erasing the screen before drawing. The

arguments are the same as ds2d.

Examples: ds2dn

ds2dn('fillnb')

NMR Spectroscopy User Guide See also:

Related: ds2d Display 2D spectra in whitewash mode (C)

#### Report statistical signal-to-noise for Cold Probes (M) dsnarray

Applicability: Systems with Varian, Inc. Cold Probes

Description: Report the statistical S/N of a series of repeated gNhsqc data sets acquired with

a labeled protein sample.

#### dscale Display scale below spectrum or FID (C)

dscale<(<rev><,axis><,label><,vp0><,sp0><,color><,pen>)>

Description: Displays a scale under a spectrum or FID.

rev – reverses the direction of the scale. That is, the smaller numbers will be at Arguments:

the left side of the scale. If used, 'rev' must be the first argument.

axis – If the letter p, h, k, etc. is supplied, it will be used instead of the current value of the parameter axis. For an FID scale, if the letter s, m, or u is supplied, it will be used instead of the current value of the parameter axisf.

label – If a string of 2 or more characters is supplied, it will be used as the

axis label.

vp0 – This is supplied as the first real number. It defines the vertical position where the scale is drawn. The default is 5 mm below the current value of the parameter vp.

sp0 – This is supplied as the second real number. It is a modified start of plot. If, for example, the display is from 347 to 447 hz, but the scale is desired to read 0 to 100 hz., sp0 would be input as 0.

wp0 – This is supplied as the third real number. It is a modified width of plot. If, for example, the display is from 347 to 447 hz, but the scale is desired to read 0 to 550 Units. sp0 would be input as 0, wp0 would be 550, and the label would be 'Units'.

An optional color or pen number can be supplied to dscale or pscale. The available colors and pens are: 'red', 'green', 'blue', 'cyan',

'magenta', 'yellow', 'black', 'white' 'pen1', 'pen2', 'pen3',..., 'pen8'

Examples: dscale

dscale('rev')

dscale('h',0,'green') dscale('h', vp-10,0)

See also: NMR Spectroscopy User Guide

Related: axis Axis label for displays and plots (P)

> axisf Axis label for FID displays and plots (P) pscale Plot scale below spectrum or FID (C) Vertical position of spectrum (P)

#### dscoef Digital filter coefficients for downsampling (P)

Description: Specifies the number of coefficients used in the digital filter. This parameter

does not need to be changed as the parameter downsamp is changed, because dscoef is automatically adjusted by VnmrJ to give filter cutoffs that are the

same, regardless of the value of downsamp. This is done by using

dscoef\*downsamp/2 coefficients in the digital filter. VnmrJ always rounds dscoef\*downsamp/2 to an odd number. If dscoef does not exist in the current experiment, enter addpar ('downsamp') to add it. Entering addpar ('downsamp') creates the digital filtering and downsampling parameters downsamp, dscoef, dsfb, dslsfrq, and filtfile.

Values: Number of digital filter coefficients. The default is 61. A larger number of

coefficients gives a filter with sharper cutoffs; a smaller number gives a filter

with more gradual cutoffs.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to current experiment (M)

downsampDownsampling factor applied after digital filtering (P)dsfbDigital filter bandwidth for downsampling (P)dslsfrqBandpass filter offset for downsampling (P)filtfileFile of FIR digital filter coefficients (P)

pards Create additional parameters used for downsampling (M)

## dseq Decoupler sequence for first decoupler (P)

Applicability: Systems with waveform generators.

Description: Specifies the decoupling sequence (without the .DEC file extension) to be used

during any period of programmable decoupling on the first decoupler under status control (i.e., dmm='p'). The decoupling sequence must be located in the user's shapelib directory or in the VnmrJ system's shapelib directory.

See also: NMR Spectroscopy User Guide

Related: dmm Decoupler modulation mode for first decoupler (P)

dseq2 Decoupler sequence for second decoupler (P)
dseq3 Decoupler sequence for third decoupler (P)

## dseq2 Decoupler sequence for second decoupler (P)

Applicability: Systems with waveform generators.

Description: Specifies the decoupling sequence (without the .DEC file extension) to be used

during any period of programmable decoupling on the second decoupler under status control (i.e., dmm2='p'). The decoupling sequence must be located in the user's shapelib directory or in the VnmrJ system shapelib directory.

See also: NMR Spectroscopy User Guide

Related: dmm2 Decoupler modulation mode for second decoupler (P)

dseq Decoupler sequence for first decoupler (P)

# dseq3 Decoupler sequence for third decoupler (P)

Applicability: Systems with waveform generators.

Description: Specifies the decoupling sequence (without the .DEC file extension) to be used

during any period of programmable decoupling on the third decoupler under status control (i.e., dmm3 = 'p'). The decoupling sequence must be located in

the user's shapelib directory or in the shapelib directory.

See also: NMR Spectroscopy User Guide

Related: dmm3 Decoupler modulation mode for third decoupler (P)

dseq Decoupler sequence for first decoupler (P)

D

dseq4 Decoupler sequence for fourth decoupler (P)

Applicability: Systems with waveform generators.

Description: Specifies the decoupling sequence (without the .DEC file extension) to be used

during any period of programmable decoupling on the third decoupler under status control (i.e., dmm4 = 'p'). The decoupling sequence must be located in the user's shapelib directory or in the system's shapelib directory.

See also: NMR Spectroscopy User Guide

Related: dmm4 Decoupler modulation mode for third decoupler (P)

dseq Decoupler sequence for first decoupler (P)

dsfb Digital filter bandwidth for downsampling (P)

Description: Specifies the bandwidth of the digital filter used for downsampling. If dsfb

does not exist in the current experiment, enter addpar ('downsamp') to add it. addpar ('downsamp') creates the digital filtering and downsampling parameters downsamp, dscoef, dsfb, dslsfrq, and filtfile.

Values: Number, in Hz. A smaller value rejects frequencies at the spectrum edges; a

larger value aliases noise and signals at frequencies outside of  $\pm sw/2$ .

'n' makes dsfb default to the final sw/2.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to current experiment (M)

downsampDownsampling factor applied after digital filtering (P)dscoefDigital filter coefficients for downsampling (P)dslsfrqBandpass filter offset for downsampling (P)filtfileFile of FIR digital filter coefficients (P)

pards Create additional parameters used for downsampling (M)

sw Spectral width in directly detected dimension (P)

dshape Display pulse shape or modulation pattern (M)

Syntax: dshape<(pattern.ext)>

Description: Displays the real (X) and imaginary (Y) components of a shaped pulse. Any

type of waveform (.RF, .DEC or .GRD) can be displayed.

Arguments: pattern is the name of a shape or pattern file specified by an absolute file

name, relative file name, or a simple pattern file name. ext is a file name extension that specifies the file type. In the case of a simple file name, dshape searches for the file in the local directory, then in the user's shapelib, and finally in the directory /vnmr/shapelib. If pattern.ext is not given, dshape displays the last created waveform stored in the pbox.fid file.

Examples: dshape

dshape('Pbox.RF')

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

pshape Plot pulse shape or modulation pattern (M)

dshapef Display last generated pulse shape (M)

Description: Displays the real (X) and imaginary (Y) components of last generated shaped

pulse, stored in pbox.fid file.

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

pshapef Plot last generated pulse shape (M)

# dshapei Display pulse shape or modulation pattern interactively (M)

Syntax: dshapei<(pattern.ext)>

Description: Displays the real (X) and imaginary (Y) components of a pulse shape,

modulation pattern or gradient shape interactively. dshapei overwrites the existing data (FID) after the permission is granted by the user. It also asks for

the duration of the waveform and displays the timescale.

Arguments: pattern is the name of a shape or pattern file specified by an absolute file

name, relative file name, or a simple pattern file name. ext is a file name extension that specifies the file type. In the case of a simple file name, dshapei searches for the file in the local directory, then in the user's shapelib, and finally in the directory /vnmr/shapelib. If no file name is given, dshapei displays the last created waveform stored in the pbox.fid

file.

Examples: dshapei

dshapei('myfile.DEC')

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

# dshim Display a shim "method" string (M)

Syntax: (1) dshim<(file) >

(2) dshim('method'|'help')

Description: Looks in the user's shimmethods directory and then in the system

shimmethods directory for a file and displays the file (syntax 1) or displays

information about method strings (syntax 2).

Arguments: file is the name of a file to be searched for in the shimmethods directories.

The default is to display the contents of the shimmethods directories.

'method' is a keyword to explain the structure of method strings.

'help' is a keyword to describe the method strings in the system's

shimmethods directory.

Examples: dshim

dshim('method')
dshim('help')

See also: NMR Spectroscopy User Guide

Related: method (P)

newshm Interactively create a shim "method" with options (M) shim Submit an Autoshim experiment to acquisition (C)

stdshm Interactively create a shim "method" (M)

#### dslsfrq Bandpass filter offset for downsampling (P)

Description: For downsampling, selects a bandpass filter that is not centered about the

transmitter frequency. In this way, dslsfrq works much like lsfrq. If dslsfrq does not exist in the current experiment, add it by entering

addpar('downsamp'). The command addpar('downsamp') creates

the digital filtering and downsampling parameters downsamp, dscoef,

dsfb, dslsfrq, and filtfile.

Values: A number, in Hz. A positive value selects a region upfield from the transmitter

frequency; a negative value selects a downfield region.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to current experiment (M)

downsampDownsampling factor applied after digital filtering (P)dscoefDigital filter coefficients for downsampling (P)dsfbDigital filter bandwidth for downsampling (P)filtfileFile of FIR digital filter coefficients (P)lsfrqFrequency shift of the fn spectrum in Hz (P)

movedssw Set parameters for digital filtering and downsampling (M)

pards Create additional parameters used by downsampling (M)

# dsn Measure signal-to-noise (C)

Syntax: dsn<(low\_field,high\_field)>:signal\_to\_noise,noise

Description: Measures the signal-to-noise ratio of the spectrum by first measuring the

intensity of the largest peak in the spectral range defined by sp and wp, and then measuring the noise in the spectral region defined by the position of the two cursors. The noise value returned from dsn is not scaled by vs. The

interrelations between the signal-to-noise ratio, the noise, and peak intensities can be illustrated by comparing dsn: \$sn, \$noise and peak: \$signal. In

this case, \$sn is equal to (\$signal /\$noise)/vs.

Calculate noise by first doing a drift correction on the noise region. Noise is defined as:

$$noise = \left(\sum_{1=1}^{np} Y_{i2}/np\right)^{\frac{1}{2}}$$

 $Y_{i2}$  values are the square of the drift-corrected amplitude and np is the number

of points in the noise region.

Arguments: low\_field and high\_field are the upper and lower frequencies of the

noise region to be measured. The default is the position of the two cursors.

signal\_to\_noise is the calculated value of signal-to-noise ratio.
noise is the noise value measured within the defined spectral region.

Examples: dsn:\$ston

dsn(sp+sp,sp+wp-100) dsn(10000,8000):r1

See also: *User Programming* 

Related: dres Measure linewidth and digital resolution (C)

peak Find tallest peak in specified region (C)

sp Start of plot (P)
vs Vertical scale (P)
wp Width of plot (P)

## dsnmax Calculate maximum signal-to-noise (M)

Syntax: dsnmax<(noise region) >

Description: Finds the best signal-to-noise in a specified region.

Arguments: noise region is the size, in Hz, of the region. The default is the region

between the cursors as defined by the parameter delta.

Examples: dsnmax

dsnmax(400)

See also: User Programming

Related: delta Cursor difference in directly detected dimension (P)

# dsp Display calculated spectrum (C)

Syntax: dsp<(file<, 'nods'>)>

Description: Using the current table of transitions and intensities, dsp recalculates the

simulated spectrum (using the current value for the linewidth slw) and displays the spectrum. dsp can only be used after the spins program has been run. If only the linewidth slw or vertical scale svs have been changed, dsp can be used to redisplay the spectrum. If a chemical shift or coupling constant has been changed, however, dsp will not display a spectrum reflecting the changes in the parameter; spins must be run again to recalculate the new spectrum.

The number of points in the calculated spectrum is fn/2. To increase the number of points, change fn and rerun dsp without doing a transform.

To display a synthetic spectrum, prepare a file in the following format:

Freq1, Intens1, LineWidth1, GaussFrac1 Freq2, Intens2, LineWidth2, GaussFrac2

FreqN, IntensN, LineWidthN, GaussFracN

The units for frequency and line width are Hz. The Gaussian fraction, which is the percentage of the line shape that is Gaussian (the rest is Lorentzian) should be between 0 and 1 (i.e., 0 is pure Lorentzian, 1 is pure Gaussian). Units for intensity are not particularly important. Given numbers in a file myshape, it is only necessary to enter dsp('myshape') to display the synthetic spectrum. This approach is often preferred over deconvolution for quantifying small shoulders on large peaks.

Arguments: file is the name of a file containing spectral information that displays the

result of a spectrum deconvolution. Any file in the proper format can be used to generate a display. The default is the file spins.outdata in the experiment directory. This file contains information about frequencies, intensities, line

widths, and Gaussian/Lorentzian fractions.

'nods' is a keyword for dsp to recalculate the simulated spectrum but not to display the spectrum. The spectrum can be displayed with the ds or dss

command.

Examples: dsp

dsp('fitspec.outpar')

See also: NMR Spectroscopy User Guide

Related: ds Display a spectrum (C)

dss Display stacked spectra (C)

fn Fourier number in directly detected dimension (P)

Spin simulation linewidth (P)

spins Perform spin simulation calculation (C)
svs Spin simulation vertical scale (P)

D

# dsp Type of DSP for data acquisition (P)

Description: Selects the type of DSP (digital signal processing) for data acquisition:

- *Inline DSP* performs digital filtering and downsampling on the workstation immediately after each oversampled FID is transferred from the console. sw and at should be set to the values desired for the final spectrum. Only the digital filtered and downsampled data is written to the disk. Selective detection of a region of a spectrum is available using the moveossw macro.
- *Real-time DSP* uses optional hardware (not available on all systems) to filter the data prior to summing to memory. Real-time DSP is not compatible with pulse sequences that use explicit acquisition to acquire less than the full number of data points (np) in a single acquire statement (e.g., solids sequences such as BR24 and FLIPFLOP).

If either type is active, the filter bandwidth parameter fb is not active. The actual analog filter *is* active and is automatically set by the software to a value that matches (sw/2) \* oversamp as closely as possible.

Another type of DSP is available that allows post-processing of data. See the description of the pards macro for details.

Values:

'i' selects inline DSP and calls addpar('oversamp') to create the DSP parameters def\_osfilt, filtfile, oscoef, osfb, osfilt, oslsfrq, and oversamp. A value of oversamp greater than 1 causes the next experiment run to be oversampled, digitally filtered, and downsampled back to the selected sw prior to saving it to disk.

'r' selects real-time DSP and calls the macro addpar('oversamp') to create the DSP parameters def\_osfilt, filtfile, oscoef, osfb, osfilt, oslsfrq, and oversamp (although only oversamp and osfilt are user adjustable for real-time DSP). Use dsp='r' only if the optional DSP hardware is present in the system. Set fsq='y' to use frequency-shifted quadrature detection.

'n' (or parameter dsp is not present) disables both types of DSP. Set dsp='n' if you wish to turn off DSP on a permanent or semi-permanent basis. To turn off DSP within just a single experiment, set oversamp='n'.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to current experiment (M)

filtfile File of FIR digital filter coefficients (P)

fsq Frequency-shifted quadrature detection (P)

il Interleave arrayed and 2D experiments (P)

moveossw Set oversampling parameters for selected spectral region (M)

Number of data points (P)

oscoef
Oigital filter coefficients for oversampling (P)
Osfb
Oigital filter bandwidth for oversampling (P)
Osfilt
Oversampling filter for real-time DSP (P)
Oslsfrq
Oversampling filter offset for oversampling (P)
Oversampling factor for acquisition (P)

pards Create additional parameters used by downsampling (M)
paros Create additional parameters used by oversampling (M)
ra Resume acquisition stopped with sa command (C)

Stop acquisition (C)

Spectral width in the directly detected dimension (P)

#### dsplanes Display a series of 3D planes (M)

Syntax: dsplanes(start\_plane,stop\_plane)

Description: Produces a graphical 2D color or contour map for a subset of 3D planes. The

dconi program is used to display the planes.

Arguments: start\_plane specifies the number of the 3D plane with which display is to

begin. It must be greater than 0.

stop\_plane specifies the number of the 3D plane with which the display is to end. If start\_plane is greater than stop\_plane, only the first plane, whose number is start\_plane, is plotted. The range of stop\_plane depends on the value of the parameter plane as follows:

• If plane='f1f3', range of stop plane is between 0 and fn2/2

• If plane='f2f3', range of stop\_plane is between 0 and fn1/2

• If plane='f1f2', range of stop plane is between 0 and fn/2

Examples: dsplanes(1,3)

See also: NMR Spectroscopy User Guide

Related: dconi Interactive 2D data display (C)

dplane Display a 3D plane (M)

dproj Display a 3D plane projection (M)

getplane Extract planes from 3D spectral data set (M)

nextpl Display the next 3D plane (M)
plane Currently displayed 3D plane type (P)

plplanes Plot a series of 3D planes (M)
prevpl Display the previous 3D plane (M)

# dsptype Type of DSP (P)

Description: Indicates the existence of digital signal processing (DSP).

Values: 0 indicates no digital signal processing. 1 indicates DSP exists.

Examples: dsptype?=0 dsptype?=1 See also: NMR Spectroscopy User Guide

Related: dsp Type of DSP for data acquisition (P)

# dss Display stacked spectra (C)

Syntax: dss<(<start,finish<,step>><,options>)>

Description: Displays one or more spectra on the screen.

The display is not interactive like the command ds. Integral display is controlled by the parameter intmod when a single spectrum is displayed (see 'int' option below). The following values are accepted for intmod:

- intmod='off' turns off the integral display.
- intmod='full' displays the entire integral.
- intmod='partial' displays every other integral region.

An individual trace is displayed from and arrayed 1D spectra or 2D spectra by supplying the index number as an argument. Spectra from 2D data set are displayed from either the  $f_1$  or  $f_2$  domain by setting the parameter <code>trace</code> equal to <code>'f1'</code> or <code>'f2'</code>, respectively. Enter <code>ft1d</code>, <code>trace='f1'</code>, and <code>dss</code> to view the interferogram. Multiple spectra are displayed by supplying indexes of the first and last spectra.

The position of the first spectrum is governed by the parameters wc, sc, and vp. For 1D data, subsequent spectra are positioned relative to the preceding spectrum by the parameters vo (vertical offset) and ho (horizontal offset). For 2D data, ho defines the total horizontal offset between the first and last spectrum. Also for 2D data, vo is inactive while the parameter wc2 defines the total vertical offset between the first and last spectrum.

The parameter cutoff, if it exists and is active, defines the distance above and below the current vertical position vp at which peaks are truncated. By arraying cutoff to have two different values, the truncation limits above and below the current vertical position can be controlled independently. For example, cutoff=50 truncates peaks at vp+50 mm and vp-50 mm. cutoff=50, 10 truncates peaks at vp+50 mm and vp-10 mm.

Arguments:

start is the index of the first spectra when displaying multiple spectra. It is also the index number of a particular trace to be viewed when displaying arrayed 1D spectra or 2D spectra.

finish is the index of the last spectra when displaying multiple spectra. Since the parameter arraydim is automatically set to the total number of spectra, it can be used to set finish to include all spectra (e.g., dss(1,arraydim,3)).

step is the increment for the spectral index when displaying multiple spectra. The default is 1.

options can be any of the following:

- 'all' is a keyword to display all of the spectra.
- 'int' is a keyword to display only the integral, independently of the value of the parameter intmod
- 'top' or 'side' are keywords that cause the spectrum to be displayed either above or at the left edge, respectively, of a contour plot. This assumes that the parameters sc, wc, sc2, and wc2 are those used to position the contour plot.
- 'dodc' is a keyword for all spectra to be drift corrected independently.
- 'red', 'green', 'blue', 'cyan', 'magenta', 'yellow', 'black', and 'white' are keywords that select a color.
- 'pen1', 'pen2', 'pen2' ... are keywords that pens.
- 'nopars' prevents the display commands from drawing the parameters at the bottom of the graphics screen.
- 'custom' uses the parameters shownumx (x position) and shownumy (y position), counting from bottom left of every spectrum.
- 'reverse' rotate the text by 90° useful if the arrayed parameter values are long with respect to the width of the individual sub-spectra.
- 'value' —The values of up to two simultaneous arrays are displayed. Diagonal arrays are allowed. The second parameter is shown in different color). The name of the arrayed parameter(s) is also shown. If used on a one-dimensional array representation of a 2D spectrum, ni and phase (in case of phase sensitive 2Ds) parameters are shown.

Examples: dss(1,3)

dss(1,12,3,'green')

See also: NMR Spectroscopy User Guide

Related: cutoff Data truncation limit (P)

dssa Display stacked spectra automatically (C

dssan Display stacked spectra automatically without erasing (C)

dssh Display stacked spectra horizontally (C)

dsshn Display stacked spectra horizontally without erasing (C)

dssn Display stacked spectra without screen erase (C)

ho Horizontal offset (P)
intmod Integral display mode (P)

pl Plot spectra (C)

plww Plot spectra in whitewash mode (C)

Start of chart (P)

Start of chart in second direction (P)

shownumx x position counting from bottom left of every spectrum (P) y position counting from bottom left of every spectrum (P)

trace Mode for 2D data display (P)

vo Vertical offset (P)

vp Vertical position of spectrum (P)

wc Width of chart (P)

wc2 Width of chart in second direction (P)

# dssa Display stacked spectra automatically (C)

Syntax: dssa<(<start,finish<,step>><,options>)>

Description: Displays one or more spectra automatically.

Integral display is controlled by the parameter intmod when a single spectrum is displayed (see 'int' option below). The following values are accepted for intmod:

- intmod='off' turns off the integral display.
- intmod='full' displays the entire integral.
- intmod='partial' displays every other integral region.

An individual trace is displayed from and arrayed 1D spectra or 2D spectra by supplying the index number as an argument. Spectra from 2D data set are displayed from either the  $f_1$  or  $f_2$  domain by setting the parameter trace equal to 'f1' or 'f2', respectively. Enter ftld, trace = 'f1', and dss to view the interferogram. Multiple spectra are displayed by supplying indexes of the first and last spectra.

The position of the first spectrum is governed by the parameters wc, sc, and vp. For 1D data, subsequent spectra are positioned relative to the preceding spectrum by the parameters vo (vertical offset) and ho (horizontal offset). For 2D data, ho defines the total horizontal offset between the first and last spectrum.

Also for 2D data, vo is inactive while the parameter wc2 defines the total vertical offset between the first and last spectrum. To display spectra "automatically," the command dssa adjusts the parameters vo and ho to fill the screen in a lower left to upper right presentation (wc must be set to less than full screen width for this to work)

The parameter cutoff, if it exists and is active, defines the distance above and below the current vertical position vp at which peaks are truncated. By arraying cutoff to have two different values, the truncation limits above and below the current vertical position can be controlled independently. For example, cutoff=50 truncates peaks at vp+50 mm and vp-50 mm. cutoff=50, 10 truncates peaks at vp+50 mm and vp-10 mm.

#### Arguments:

start is the index of the first spectra when displaying multiple spectra. It is also the index number of a particular trace to be viewed when displaying arrayed 1D spectra or 2D spectra.

finish is the index of the last spectra when displaying multiple spectra.

step is the increment for the spectral index when displaying multiple spectra. The default is 1.

options can be any of the following:

- 'all' is a keyword to display all of the spectra.
- 'int' is a keyword to only display the integral, independently of the value of the parameter intmod
- 'dodc' is a keyword for all spectra to be drift corrected independently.
- 'red', 'green', 'blue', 'cyan', 'magenta', 'yellow', 'black', and 'white' are keywords that select a color.
- 'pen1', 'pen2', 'pen2' ... are keywords that pens.
- 'nopars' prevents the display commands from drawing the parameters at the bottom of the graphics screen.

Examples: dssa(1,3)

See also: NMR Spectroscopy User Guide

Related: cutoff Data truncation limit (P)

dss Display stacked spectra (

dss Display stacked spectra (C)
dssan Display stacked spectra automatically without erasing (C)

dssh Display stacked spectra horizontally (C)

dsshn Display stacked spectra horizontally without erasing (C)

dssn Display stacked spectra without screen erase (C)

dsww Display spectra in whitewash mode (C)
ftld Fourier transform along f<sub>2</sub> dimension (C)

ho Horizontal offset (P)
intmod Integral display mode (P)

pl Plot spectra (C)

plww Plot spectra in whitewash mode (C)

Start of chart (P)

Start of chart in second direction (P)

shownumx x position counting from bottom left of every spectrum (P) y position counting from bottom left of every spectrum (P)

trace Mode for 2D data display (P)

vo Vertical offset (P)

vp Vertical position of spectrum (P)

WC Width of chart (P)

wc2 Width of chart in second direction (P)

# dssan Display stacked spectra automatically without erasing (C)

Syntax: dssan<(<start,finish<,step>><,options>)>

Description: Functions the same as the command dssa except the graphics window is not

erased before starting the display. This allows composite displays of many

spectra to be created. The arguments are the same as dssa.

Examples: dssan(1,3)

See also: NMR Spectroscopy User Guide

Related: dssa Display stacked spectra automatically (C)

# dssh Display stacked spectra horizontally (C)

Syntax: dssh<(<start,finish<,step>><,options>)>

Description: Displays one or more spectra horizontally.

Integral display is controlled by the parameter intmod when a single spectrum is displayed (see 'int' option below). The following values are accepted for intmod:

- intmod='off' turns off the integral display.
- intmod='full' displays the entire integral.
- intmod='partial' displays every other integral region.

An individual trace is displayed from and arrayed 1D spectra or 2D spectra by supplying the index number as an argument. Spectra from 2D data set are displayed from either the  $f_1$  or  $f_2$  domain by setting the parameter <code>trace</code> equal to <code>'f1'</code> or <code>'f2'</code>, respectively. Enter <code>ftld</code>, <code>trace='f1'</code>, and <code>dss</code> to view the interferogram. Multiple spectra are displayed by supplying indexes of the first and last spectra.

The position of the first spectrum is governed by the parameters wc, sc, and vp. For 1D data, subsequent spectra are positioned relative to the preceding spectrum by the parameters vo (vertical offset) and ho (horizontal offset). For 2D data, ho defines the total horizontal offset between the first and last spectrum. Also for 2D data, vo is inactive while the parameter wc2 defines the total vertical offset between the first and last spectrum. To display spectra horizontally, the command dssh causes vo to be set to zero and for ho, sc, and wc to be adjusted to fill the screen from left to right with the entire array.

The parameter cutoff, if it exists and is active, defines the distance above and below the current vertical position vp at which peaks are truncated. By arraying cutoff to have two different values, the truncation limits above and below the current vertical position may be controlled independently. For example, cutoff=50 truncates peaks at vp+50 mm and vp-50 mm, and cutoff=50,10 truncates peaks at vp+50 mm and vp-10 mm.

# Arguments:

start is the index of the first spectra when displaying multiple spectra. It is also the index number of a particular trace to be viewed when displaying arrayed 1D spectra or 2D spectra.

finish is the index of the last spectra when displaying multiple spectra.

step is the increment for the spectral index when displaying multiple spectra. The default is 1.

options can be any of the following:

- 'all' is a keyword to display all of the spectra.
- 'int' is a keyword to only display the integral, independently of the value of the parameter intmod
- 'dodc' is a keyword that causes all spectra to be drift corrected independently.
- 'red', 'green', 'blue', 'cyan', 'magenta', 'yellow', 'black', and 'white' are keywords that select a color.
- 'pen1', 'pen2', 'pen2' ... are keywords that pens.
- 'nopars' prevents the display commands from drawing the parameters at the bottom of the graphics screen.

# Examples: dssh(1,3)

See also: NMR Spectroscopy User Guide

Related:	cutoff	Data truncation limit (P)
	dss	Display stacked spectra (C)
	dssa	Display stacked spectra automatically (C)
	dssan	Display stacked spectra automatically without erasing (C)
	dsshn	Display stacked spectra horizontally without erasing (C)
	dssn	Display stacked spectra without screen erase (C)

dsww Display spectra in whitewash mode (C) ftld Fourier transform along f<sub>2</sub> dimension (C)

ho Horizontal offset (P)
intmod Integral display mode (P)

pl Plot spectra (C)

plww Plot spectra in whitewash mode (C)

Start of chart (P)

Start of chart in second direction (P)

shownumx x position counting from bottom left of every spectrum (P) y position counting from bottom left of every spectrum (P)

trace Mode for 2D data display (P)

vo Vertical offset (P)

vp Vertical position of spectrum (P)

wc Width of chart (P)

wc2 Width of chart in second direction (P)

#### dsshn Display stacked spectra horizontally without erasing (C)

Syntax: dsshn<(<start,finish<,step>><,options>)>

Description: Functions the same as the command dssh except the graphics window is not

erased before starting the display. This allows composite displays of many

spectra to be created. The arguments are the same as dssh.

Examples: dssh(1,3)

See also: NMR Spectroscopy User Guide

Related: dssh Display stacked spectra horizontally (C)

#### dssl Label a display of stacked spectra (M)

Syntax: dssl(<options>)

Description: Displays a label for each element in a set of stacked spectra. The label is an

integer value from 1 up to the number of spectra in the display or the values of

parameters up to 2 dimensions.

Labels can appear at incorrect positions if wysiwyg='n'. The positions are empirically determined for a large screen display and are not guaranteed to be

correct for all displays.

Arguments: options control the display (more than one option can be entered as long as

the options do not conflict with each other):

- 'center', 'left', 'right', 'top', 'bottom', 'above', and 'below' are keywords setting the position of the displayed index relative to each spectrum.
- 'custom' uses the parameters shownumx (x position) and shownumy (y position), counting from bottom left of every spectrum.
- 'list=xxx' produces a display of the values contained in the arrayed parameter xxx.
- 'format=yyy' uses the format yyy to control the display of each label. See the write command for information about formats.
- 'reverse' rotate the text by 90° useful if the arrayed parameter values are long with respect to the width of the individual sub-spectra.
- 'value' —The values of up to two simultaneous arrays are displayed. Diagonal arrays are allowed. The second parameter is shown in different color). The name of the arrayed parameter(s) is also shown. If used on a

one-dimensional array representation of a 2D spectrum, ni and phase (in case of phase sensitive 2Ds) parameters are shown.

Examples: dssl

dssl('top','left')

dssl('value','format=%3.1f') pssl

See also: NMR Spectroscopy User Guide

Related: dss Display stacked spectra (C)

shownumx x position counting from bottom left of every spectrum (P) y position counting from bottom left of every spectrum (P)

write Write formatted text to a device (C)

# dssn Display stacked spectra without screen erase (C)

Syntax: dssn<(<start,finish<,step>><,options>)>

Description: Functions the same as the command dss except the graphics window is not

erased before starting the display. This allows composite displays of many

spectra to be created. The arguments are the same as dss.

Examples: dssn(1,3)

See also: NMR Spectroscopy User Guide

Related: dss Display stacked spectra (C)

# dsvast Display VAST data in a stacked 1D-NMR matrix format (M)

Applicability: Systems with the VAST accessory.

Syntax: dsvast<(display order,number of columns displayed)>

Description: dsvast will arrange and display the traces from a reconstructed 2D data set (see

(see vastglue)as an array of 1D spectra in a matrix of 1D spectra. If no arguments are provided, the number of rows and columns will be determined by the periodicity of the display order based on the doneQ. For example, if a block of 96 spectra (typical for a microtiter-plate) have been acquired using VAST automation, the spectra will be displayed in a matrix 8 rows and 12 columns

with the well label using the format [A->H][1->12].

The spectra can be plotted using the macro plvast.

Arguments: display order is optional and its default value is the glue order as listed in

glueorderarray. A display order can be defined using the

plate glue program.

number of columns displayed. The default value of is deduced by examining the periodicity of the requested display order. The number of columns displayed can entered as the second argument or as the first

argument if the default display order is used.

Examples: dsvast

dsvast(12)

dsvast('glue\_file', 4)

See also: NMR Spectroscopy User Guide

Related: dsast2d Display VAST data in a pseudo-2D format (M)

plvast Plot VAST data in a stacked 1D-NMR matrix (M)
plvast2d Plot VAST data in a pseudo-2D format (M)

plate glue Define a display order (U)

dsvast2d Display VAST data in a pseudo-2D format (M)

Applicability: Systems with the VAST accessory.

Syntax: dsvast2d(number)

Description: If an array of 1D spectra have been acquired (in particular if a block of 96

spectra has been acquired using VAST automation, especially in a microtiterplate format), and if these spectra have been glued into a reconstructed 2D dataset (see vastglue), this macro will arrange and display them (on the screen) in a convenient pseudo-2D format (almost like an LC-NMR chromatogram). Well labels are not attached to the spectra and spectra are

plotted with 8 spectra per row.

Arguments: The default is to display all the spectra (from 1 through arraydim) with 8

columns (spectra) and 12 rows. An optional argument dsvast2d(number) allows specifying that only spectra from l through number should be plotted. The number of spectra displayed is rounded up to the nearest multiple of 8.

Related: dsast Display VAST data in a 1D-NMR matrix format (M)

plvast Plot VAST data in a stacked 1D-NMR matrix (M)
plvast2d Plot VAST data in a pseudo-2D format (M)

dsww Display spectra in whitewash mode (C)

Syntax: dsww<(<start,finish<,step>><,'int'>)>

Description: Displays one or more spectra in whitewash mode (after the first spectra, each

spectra is blanked out in regions in which it is behind a prior spectra).

Arguments: start is the index of the first spectra when displaying multiple spectra. It is

also the index number of a particular trace to be viewed when displaying

arrayed 1D spectra or 2D spectra; default is to display all spectra.

finish is the index of the last spectra when displaying multiple spectra. step is the increment for the spectral index when displaying multiple spectra.

The default is 1.

'int' is a keyword to display only the integral, independently of the value of

the parameter intmod

Examples: dsww(1,3)

Related: dss Display stacked spectra (C)

dssa Display stacked spectra automatically (C)

dssan Display stacked spectra automatically without erasing (C)

dssh Display stacked spectra horizontally (C)

dsshn Display stacked spectra horizontally without erasing (C)

dssn Display stacked spectra without screen erase (C)

Plot spectra (C)

Plot spectra in whitewash mode (C)

dtext Display a text file in graphics window (M)

Syntax: dtext<(file,x,y)><:\$x next,\$y next,\$increment>

Description: Displays a text file in the graphics window.

Arguments: file is the name of a text file. The default is the current experiment text file.

x and y are coordinates of the first line of text. This positions the location of

the output. The default is the upper left-hand corner of the screen.

 $x_next$  and  $y_next$  are the coordinates where the start of the next line would have been displayed. This is useful for subsequent character display.

\$increment is the increment between lines.

Examples: dtext

dtext(userdir+'/exp3/text')

dtext(100,100)
dtext:\$x,\$y,\$dy

 $Related: \quad \begin{array}{ccc} \textbf{Plot a text file (M)} \end{array}$ 

Print out a text file (M)

Display text or set new text for current experiment (C)

write Write formatted text to a device (C)

# dtrig Delay to wait for another trigger or acquire a spectrum (P)

Applicability: Systems with LC-NMR accessory.

Description: If ntriq is greater than 0 after a trigger is detected, a pulse sequence waits for

dtrig seconds before either waiting for another trigger or acquiring a spectrum. Typically, after the LC has positioned the sample in the NMR probe and stopped the pump, there is a small time (30 seconds) during which conditions (pressure, etc.) in the NMR probe are still settling; better NMR performance is obtained if an appropriate delay is inserted using dtrig. If dtrig does not exist, a value of 0 is assumed. If dtrig does not exist, the

parlc macro can create it.

parlc Create LC-NMR parameters (M)

# dutyc Duty cycle for homodecoupling (optional) (P)

Applicability: VNMRS systems, 400 MR

Syntax: dutyc=<value>

Description: Sets the rf duty cycle fraction (0.0-0.4) for rf on part of homonuclear

decoupling. The duty cycle default is 0.1 (or 10% rf on) if the dutyc does not

exist. Homonuclear decoupling delay before and after the rf on period.

homorof1, homorof2, and homorof3, are equivalent to rof1, rof2 and

rof3 and all default to 2 µsec.

Values: 0.0 to 0.4 — default is 0.1

Examples: dutyc=0.2 sets a 20% duty cycle

Related: homo Homodecoupling control for observe channel (P)

hdof Frequency offset for homodecoupling (P)

hdpwr Sets the rf attenuator to control the power for homonuclear

decoupling (P)

hdmf modulation frequency for the band selective homonuclear

decoupling (P)

hdpwrf Sets the rf linear modulator fine power for homonuclear

decoupling (P)

hdres Sets the tip angle resolution (P)

hdseq Sets the decoupler waveform filename (P)
homorof1 Delay before turning on homo decoupling rf (P)

homorof2 Delay after blanking the amplifier and setting T/R switch to receive (P)

D

 $\begin{array}{ll} \mbox{homorof3} & \mbox{Delay between setting T/R switch to receive gating on the receiver (P)} \\ \mbox{tn} & \mbox{Nucleus for observe transmitter (P)} \end{array}$ 

# Ε

Eject sample (M) eaddr Display Ethernet address (M,U) Turns on eddy current compensation for Cold Probes (M) ecc\_on ecc off Turns off eddy current compensation for Cold Probes (M) echo Display strings and parameter values in text window (C) edit Edit a file with user-selectable editor (M) eject Eject sample (M) elist Display directory on remote VXR-style system (M,U) Email address (P) email enter Enter sample information for automation run (M,U) Start a dialog window using enterexp file (M) enterdialog Transfer file from remote source (M,U) eread Calculate the Ernst angle pulse (C) ernst errlog Display recent error messages (C) errloglen Number of lines in error message display (P) Transfer file to remote destination (M,U) ewrite Execute a command (C) exec execpars Set up the exec parameters (M) Execute plotting macro (P) execplot Execute prepare macro (P) execprep execprescan Execute prescan macro (P) execproc Execute processing macro (P) Execute processing macro (P) execprocess Execute setup macro (P) execsetup exists Checks if parameter, file, or macro exists and file type (C) exit Call the vnmrexit command (M) Find exponential value of a number (C) exp expactive Determine if experiment has active acquisition (C) expfit Make least-squares fit to polynomial or exponential curve (U) expl Display exponential or polynomial curves (C) expladd Add another diffusion analysis to current display (M) explib Display experiment library (M) explist Display current experiment chain and approx. time for each (M) Display log file for experiment (M) explog exptime Display experiment time (C)

## e Eject sample (M)

Description: Ejects the sample from the probe by turning on the eject air and the slow drop air. The e macro functions the same as the eject macro.

See also: NMR Spectroscopy User Guide

Related: eject Eject sample (M)

i Insert sample (M)
insert Insert sample (M)

eaddr Display Ethernet address (M,U)

Description: Displays the name of the local host and its hardware Ethernet address. The 48-

bit address is presented in octal, decimal, and hexadecimal formats.

See also: NMR Spectroscopy User Guide

Related: dnode Display list of valid limNET nodes (M,U)

ecc on Turns on eddy current compensation for Cold Probes (M)

Applicability: Systems with Varian, Inc. Cold Probes

Description: Turns on eddy current compensation

Related: ecc off Turns off eddy current compensation for Cold Probes (M)

ecc off Turns off eddy current compensation for Cold Probes (M)

Applicability: Systems with Varian, Inc. Cold Probes

Description: Turns off eddy current compensation.

Related: ecc on Turns on eddy current compensation for Cold Probes (M)

echo Display strings and parameter values in text window (C)

Syntax: echo<(<'-n',>string1,string2, ....)>

Description: Displays strings and parameter values in the text window similar to the UNIX

echo command.

Arguments: -n is a keyword that suppresses advancing to the next line. The default is to

advance to the next line.

string1, string2, ... are one or more strings (surrounded with single quote marks) or parameters. The format used for numbers is identical to the %g

format described for the write command.

Examples: echo

echo('This is a string')
echo('Pulse Width is: ',pwr)
echo('-n','No new line')

See also: User Programming

Related: write Write formatted text to a device (C)

edit Edit a file with user-selectable editor (M)

Syntax: edit(file)

Description: Opens a file for editing using a text editor. The default editor is vi. To select

another editor, set the UNIX environmental variable vnmreditor to the name of the editor (change the line setenv vnmreditor old\_editor in .login to become setenv vnmreditor new\_editor, e.g., setenv vnmreditor emacs) and make sure a script with the prefix vnmr followed

by the name of the editor (e.g., vnmr emacs) is placed in the bin

subdirectory of the system directory. The script file makes adjustments for the

type of graphic interface in use.

Scripts provided with VnmrJ include vnmr\_vi and vnmr\_textedit. To create other scripts, see the vnmr vi script for non-window editor interfaces

and the vnmr textedit script for window-based editor interfaces.

Arguments: file is the name of the file you wish to edit.

Examples: edit('myfile')
See also: User Programming

Related: paramedit Edit a parameter and its attributes with user-selected editor (C)

paramvi Edit a parameter and its attributes with vi editor (M)
macroedit Edit a user macro with user-selectable editor (C)

macrovi Edit a user macro with vi editor (C)
menuvi Edit a menu with the vi editor (M)

textvi Edit text file of current experiment with vi editor (M)

# eject Eject sample (M)

Syntax: eject

Description: Ejects the sample from the probe by turning on the eject air and the slow drop

air. The e macro functions the same as the e macro.

See also: NMR Spectroscopy User Guide

Related: e Eject sample (M)

i Insert sample (M)
insert Insert sample (M)

## elist Display directory on remote VXR-style system (M,U)

Syntax: elist(remote node, remote directory)

(From UNIX) elist remote node remote directory

Description: Lists directory contents on a remote VXR-style (Gemini, VXR-4000, or XL)

system.

Arguments: remote node is the name of the remote VXR-style system.

remote directory is the name of the directory on the remote system.

Examples: elist('gemini','fidlib')

(From UNIX) elist gemini fidlib

See also: NMR Spectroscopy User Guide

Related: dnode Display list of valid limNET nodes (M,U)

# email Email address (P)

Applicability: VnmrJ Walkup

Description: A global parameter set to the email address of an operator. It is used to send an

email message to an operator when an experiment or sample is complete. The parameter is set from the operator email field in the VnmrJ Adm interface.

See also: VnmrJ Installation and Administration, VnmrJ Walkup

Related: operatorlogin Sets workspace and parameters for the operator (M)

prescan Study queue prescan (P)

enter Enter sample information for automation run (M,U)

Applicability: Systems with an automatic sample changer.

Syntax: enter<(file<,configuration file>)>

(From UNIX) enter <file> <configuration file>

Description: Enables entry of sample information for automation runs, including the sample

location, user information, solvent used, experiment or experiments to run, and arbitrary text information. enter('abc') creates a directory named abc. In this directory is a file named abc, which contains experiment information.

Arguments: file is the name of the file to be edited. The default is that enter prompts for

this information. If the file already exists, new entries are appended to it.

configuration\_file is the name of a user-supplied file that customizes enter for local use. Several configuration files are provided:

• enter.conf is used when defining an experiment when an automation run is not currently active.

• auto.conf is used when defining an experiment for a current automation run. The walkup macro is provided for this style of entering samples.

• gilson.conf is used with the VAST accessory.

Examples: (From VnmrJ or UNIX) enter

 $\begin{array}{l} \hbox{(From $V$nmrJ)$ enter('mysamples')} \\ \hbox{(From $U$NIX)$ enter $MySamples} \end{array}$ 

(From VnmrJ) enter('mysamples', 'auto.conf')

See also: NMR Spectroscopy User Guide; User Programming,

VnmrJ Walkup

Related: auto Set up an automation directory (C)

autogo Start an automation run (C)
autoname Prefix for automation data file (P)
autora Resume a suspended automation run (C)
autosa Suspend current automation run (C)

printer
Printer device (P)

status Display status of all experiments (C)

walkup Walkup automation (M)

enterdialog Start a dialog window using enterexp file (M)

Applicability: Systems with automation.

Syntax: enterdialog

Description: Internal macro used by enter to start a dialog window using the enterexp

file in the dialoglib directory.

See also: NMR Spectroscopy User Guide; User Programming,

VnmrJ Walkup

 $\label{eq:Related:enter} Related: \quad \mbox{enter sample information for automation run } (M,U)$ 

eread Transfer file from remote source (M,U)

Applicability: Systems with limNET protocol software installed.

Syntax: (From VnmrJ) eread(local file, remote node, remote file)

(From UNIX) eread local\_file remote\_node remote\_file

Description: Copies a remote file to the local host. It will not overwrite a preexisting file.

Arguments: local\_file is the file name of the local host. If local\_file is not a dot

file (i.e., starts with "."), eread uses the "I1" and "I2" values of the remote file

to create an extension and then append it to the local file name.

remote\_node is a symbolic node name for a specified node file. Use the command dnode to list nodes defined on your system. The names of the remote computers or "nodes" available to the limNET protocol are contained in the file /vnmr/nodes. Note that this is not the same file as the name of the remote computers available to the Internet protocol (IP), which are contained in the file /etc/hosts. Each user only needs to know the "names" of relevant

nodes.

remote file is the name of file to be transferred from the remote host.

Examples: (From VnmrJ) eread('osv700','VXR4000','dsk1.osv700')

(From UNIX) eread osv700 VXR4000 dsk1.osv700

See also: NMR Spectroscopy User Guide

Related: dnode Display list of valid limNET nodes (M,U)

ewrite Transfer file to remote destination (M,U)

# ernst Calculate the Ernst angle pulse (C)

Syntax: ernst(t1 estimate<,90 pulse width>)

Description: Calculates the optimum ("Ernst") pulse width according to the formula

 $pw = cos^{-1} (exp^{-(at+d1)/t1} = estimate) \bullet (pw = 90/360)$ 

The new pw value is entered in the parameter table.

Arguments: t1 estimate is an estimate of the  $T_1$  for a peak of interest.

90 pulse width is a 90° pulse width determined by the parameter pw90.

The default is the current value of parameter pw90 if pw90 exists.

Examples: ernst(5)

ernst(3,12.6)

See also: NMR Spectroscopy User Guide
Related: pw Pulse width (P)

pw90 90° pulse width (P)

# errlog Display recent error messages (C)

Description: Displays in the text window the most recent error messages. The global

parameter errloglen controls the number of lines displayed. If errloglen

is not defined, errlog displays 10 lines by default.

See also: NMR Spectroscopy User Guide

Related: acqstatus Acquisition status (P)

errloglen Number of lines in error message display (P)

#### errloglen Number of lines in error message display (P)

Description: Sets the number of lines in the display of error messages by errlog.

Values: Integer, default is 10.

See also: NMR Spectroscopy User Guide

Related: errlog Display recent error messages (P)

ewrite Transfer file to remote destination (M,U)

Applicability: Systems with limNET protocol software installed.

Syntax: (From VnmrJ) ewrite (local file, remote node, remote file)

(From UNIX) ewrite local\_file remote\_node remote\_file

Description: Takes a preexisting local file and copies it to a remote host. The file cannot

preexist on the remote host.

Arguments: local\_file is the file name of the local host.

remote\_node is a symbolic node name for a specified node file. Use the command dnode to list nodes defined on your system. The names of the remote computers or "nodes" available to the limNET protocol are contained in the file /vnmr/nodes. Note that this is not the same file as the name of the remote computers available to the Internet Protocol (IP), which are contained in the file /etc/hosts. Each user only needs to know the "names" of relevant

iodes.

remote file is the name of file to be transferred from the remote host.

Examples: (From VnmrJ) ewrite('osv700','VXR4000','dsk1.osv700')

(From UNIX) ewrite osv700 VXR4000 dsk1.osv700

See also: NMR Spectroscopy User Guide

Related: dnode Display list of valid limNET nodes (M,U)

eread Transfer file from remote source (M,U)

exec Execute a command (C)

Syntax: exec(command string)

Description: Executes the command given by the string argument.

Arguments: command string is a character string constructed from a macro.

Examples: exec(\$cmdstr)

exec(parstyle)

See also: User Programming

execpars Set up the exec parameters (M)

Description: Set up the exec parameters as listed in /vnmr/execpars.

See also: User Programming

Related: apptype Application type (P)

execplotExecute plotting macro (P)execprepExecute prepare macro (P)execprescanExecute prescan macro (p)execprocExecute processing macro (P)execsetupExecute setup macro (P)

execplot Execute plotting macro (P)

Description: Defines which plotting macro to use to plot this experiment.

See also: User Programming

Related: apptype Application type (P)

plot Automatically plot spectra (M)

execprep Execute prepare macro (P)

Description: Defines which prepare macro to use to prescan this experiment.

See also: *User Programming* 

Related: apptype Application type (P)

acquire Acquire data (M)

plot Automatically plot spectra (M)

execprescan Execute prescan macro (P)

Description: Defines which prescan macro to use to prescan this experiment.

See also: User Programming

 $Related: \quad \begin{array}{c} \text{apptype} & \quad \\ \text{Application type (P)} \end{array}$ 

acquire Acquire data (M)

execproc Execute processing macro (P)

Description: Defines which processing macro to use to process this experiment.

See also: User Programming

Related: apptype Application type (P)

acquire Acquire data (M)

execprocess Execute processing macro (P)

Description: Defines which processing macro to use to process this experiment.

See also: User Programming

execsetup Execute setup macro (P)

Description: Defines which setup macro to use to prescan this experiment.

See also: User Programming

Related: apptype Application type (P)

cqexp Load experiment from protocol (M)
sqexp Load experiment from protocol (M)

exists Checks if parameter, file, or macro exists and file type (C

Syntax: exists(name,'keyword'):\$res1, \$res2

exists(name,'keyword'<,argument>):\$res1, \$res2

Description: Checks for the existence of a parameter, file, command, parameter file, or a

macro from within a macro. The command can be used to check if a file is an ASCII text file, a directory, or to search the application directories for a file or

directory.

Arguments: \$res1— results from exists are returned to the \$ variable.

\$res2 — optional: returns the absolute path to the file, command, macro, etc. The exists command does not pass anything to the optional second argument

if it does not find the specified file, command, macro, etc.

name — the name of a parameter, file, command, or macro.

keyword		description and returned values		
'maclib'	Macros reside in applications directories, or appdirs. Typical directories are the users vnmrsys/maclib directory and / vnmr/maclib. The appdirs are searched in order then macros are executed. Exists returns the following to \$res1:			
		acro is not found in any of the appdirs integer — indicates it was found in the first, second, pdir.		
		appdir (shapelib, manual, probes, shims) irectory within appdir and be used for the lib.		
'command'	The command keyword is similar to the maclib keyword, except that it firsts checks to see if the name represents a built-in Vnmr command.  Exists returns the following to \$res1:  0 — if the name is neither a built-in command nor a macro.  1 — if the name represents a built-in command.  2, 3, 4, or 5 — if name is a macro.			
'ascii'	Checks if the file specified by name is an ASCII text file. Exists returns the following to \$res1:  0 — if the file is not an ascii file.  1 — if the file is an ascii file.			
'parlib'	Checks for the file specified by name is in parlib using the path defined by applications directories or appdirs for parlib. A .par is appended to the name if it is not found and the search repeated if the file is not found on the first pass.  Exists returns the following to \$res1:  0 — if the file is not found.  1 — if the file is found.  Optional: result returned to \$res2.			
	Return the absolute path of the parameter set if it is found.			
'psglib'	Checks for the file specified by name is in psglib using the path defined by applications directories or appdirs for psglib. A.c is appended to the name if it is not found and the search repeated if the file is not found on the first pass.  Exists returns the following to \$res1:  0— if the file is not found.  1— if the file is found.  Optional: result returned to \$res2.  Return the absolute path of the file set if it is found.			
		g keywords accept an argument		
	argument	description and returned values		
'file'		Checks if the file specified by name exists.  Exists returns the following to \$res1:  0 — if the file does not exist.  1 — if the file exists.		
	'perm'	perm is a combination of one or more of the following:  r — read  w — write  e — execute  Access permission can be checked by passing one, two, or three characters in a single argument.		

keyword description and returned values Checks if the parameter file specified by name 'parameter' Exists returns the following to \$res1: 0 — if the parameter file does not exist. 1 — if the parameter file exists. tree is one of the following: 'tree' current (default if no argument is supplied), global, processed, or systemglobal. Checks for the existence of the specified directory 'directory' in the applications directories. Exists returns the following to \$res1: 0 — if the directory does not exist. 1 — if the directory exists. 'errval' An error value to return to \$res1 if the directory does not exist. Examples: exists('ni', 'parameter'):\$twod exists('/vnmr/conpar','file','rw') exists('wft','command'):\$num Using exists from within a macro to search the bin directory in the applications directories for the file myprog and, if found, return the path to the \$myprogPath argument: exists(\$myprog,'bin'):\$e,\$myprogPath if (\$e) then shell(\$myprogPath):\$res else write('line3','%s: Program %s has not been installed',\$0,\$myprog) Using exists from within a macro to search for files in the top-level of the appdirs. exists('pulsecal','') The search for pulsecal starts at the top-level of all appdirs. Using exists from within a macro to search multi-level directories: exists(probename, 'probes/'+probe) The first argument is set to '' which forces exists to check for directories in the appdirs. exists('nomacro','maclib',-1):\$ok Sets \$ok to -1 instead of 0 if nomacro does not exist in any of the applications directories. This feature can be applied to interface controls to make a button either not appear or appear grayed out if a macro (or file) does not exist. See also: User Programming Related:

appdirs Starts Applications Directory Editor (M)

> create Create new parameter in a parameter tree (C)

hidecommand Execute macro instead of command with same name (C)

which Display which macro or command is used (M) exit Call the vnmrexit command (M)

Description: Calls the vnmrexit command to exit from VnmrJ. As a macro, exit

provides a user some flexibility in defining other things to do when exiting.

CAUTION: When you exit from the VnmrJ user interface on your X display system,

whether you are using an X terminal or a Sun computer, and whether you are using OpenWindows, CDE, or Motif, you must first exit from any copy of VnmrJ running on your system. Failure to do this can cause current parameter values and even current data to be lost.

exp Find exponential value of a number (C)

Syntax: exp(value)<:n>

Description: Finds the exponential value (base e) of a number.

Arguments: value is a number.

n is the return value giving the exponential value of value. The default is to

display the exponential value in the status window.

Examples: exp(.5)

exp(val):exp val

See also: User Programming

Related at an Find arc tangent of a number (C)

Find cosine value of an angle (C)

Find natural logarithm of a number (C)

Find sine value of an angle (C)

Find tangent value of an angle (C)

expactive Determine if experiment has active acquisition (C)

Syntax: (1) expactive< (exp number) ><:\$answer>

(2) expactive('auto')<:\$mode>

(3) expactive('current')<:\$exp><,\$user>

Description: Determines whether an acquisition is active or pending in an experiment.

Arguments: exp number is the number, from 1 to 9999, of the experiment to be checked.

The default is the current experiment.

\$answer is a return value: -1 if an acquisition is not possible (e.g., the system is a data station), 0 if no acquisition active in the requested experiment, 1 if an acquisition active in that experiment, and 2 or larger if an acquisition is queued in the requested experiment (subtract 1 from the value to determine its position in the acquisition queue). With no return argument, the result displays on line 3.

'auto' is a keyword to check if the system is in automation mode.

mode is a return value: 1 if the system is in automation mode, or 0 if otherwise. With no return argument, the result is displayed on line 3.

'current' is a keyword that determines whether an active experiment has an active acquisition command running. An experiment is still considered active if it holds up additional acquisitions during its wexp processing by the 'wait' flag. If expactive ('current') does not have a return argument, results are displayed on line 3.

\$exp is a return value indicating the current active experiment number: -1 if no acquisition is possible, or 0 if no acquisition is active.

\$user is a return value indicating the user who started the acquisition. If the
system is running in automation mode, \$user is set to "auto." If no acquisition
is running, \$user is set to "nobody."

```
Examples: expactive
         expactive(3)
         expactive(2):$active
         expactive('auto'):$automode
```

#### expfit Make least-squares fit to polynomial or exponential curve (U)

Syntax: (From UNIX) expfit options <analyze.inp >analyze.list

Description: Makes a least-squares curve fitting to the data supplied in the file analyze.inp. For the specialized uses of analyze, VnmrJ macros (e.g., t1, t2, kind) are available that provide the correct file format and avoid the need for the user to select options.

> In the regression mode, the type of curve fitting, ( 'poly1', . . . ) must be selected. For regression (generalized curve fitting), the regression section in the manual NMR Spectroscopy User Guide shows the input file format and describes the menus that permit option choices indirectly through menu buttons.

The following text file is an example of the file analyze.inp (for options T1, T2, kinetics, contact time, and regression). (1), (2), etc. do not actually appear in the file but are used to identify lines in the description presented below the file.

```
(1) time
(2)
      <amp>
        2 4
               linear linear
(3)
(4)
         NEXT
(5)
      1
(6)
          1
             1
          2
             4
          3
             9
            16
(4)
         NEXT
(5)
      2
(6)
          2
             5
          3
             10
             17
```

This file contains the following information:

- (1) Optional *x*-axis title.
- (2) Optional y-axis title, for regression only.
- (3) Line containing an integer for the number of peaks, followed by another integer for the number of pairs per peak. If regression, the x-scale type and yscale type are also listed.
- (4) In the regression mode, a line beginning with the keyword NEXT is inserted at the start of each data set when the number of pairs per peak is variable, followed by an integer for the number of pairs for the peak.
- (5) An integer that indexes the peaks.
- (6) Data pairs, one to a line, listed by peak.

For options T1, T2, kinetics, and contact time, information from the file fp. out and from the array xarray are used to construct this file; therefore, it is necessary to run fp prior to analyze. For regression, this file is made by running expl ('regression').

For diffusion, contact\_time, and, if not in regression mode, poly1 and poly2, the analyze . inp file is slightly different:

- (1) List of n x-y data pairs
- (2) <text line>

- (3) <x-values> <y-values>
  (4) x y
  - 4) x y
- (1) Title line.
- (2) Descriptive text line.
- (3) Number of x values and y values.
- (4) Data pairs, one to a line, are listed by peak in the following order:
- x y (first peak, first pair)
- x y (first peak, second pair)

. . .

x y (second peak, first pair)

. . .

expfit also makes a file analyze.out that is used by expl to display the results of the analysis in addition to output to the standard output, which is usually directed to analyze.list.

#### Arguments: or

options can be any of the following:

T1 sets  $T_1$  analysis. This value is the default.

T2 sets  $T_2$  analysis.

kinetics sets kinetics analysis with decreasing peak height.

increment sets kinetics analysis with increasing peak height.

list sets an extended listing for each peak.

diffusion sets a special analysis for diffusion experiments.

contact\_time sets a special analysis for solids cross-polarization spin-lock
experiments.

regression sets regression mode, providing generalized curve fitting with choices poly1, poly2, poly3, and exp:

- poly0 calculates the mean.
- poly1 sets a linear fitting.
- poly2 sets a quadratic fitting.
- poly3 sets a cubic curve fitting.
- exp sets an exponential curve fitting.

#### Examples:

See also: NMR Spectroscopy User Guide

Related: analyze Generalized curve fitting (C)

expl Display exponential or polynomial curves (C)

fp Find peak heights (C)

kind Kinetics analysis, decreasing intensity (M)

t1  $T_1$  exponential analysis (M) t2  $T_2$  exponential analysis (M)

# expl

# Display exponential or polynomial curves (C)

Syntax: expl<(<options,>line1,line2,...)>

Description: Displays exponential curves resulting from  $T_1$ ,  $T_2$ , or kinetic analyses. Also

displays polynomial curves from diffusion or other types of analysis. The

parameters sc, wc, sc2, and wc2 control the size of the display.

In general, the first time expl is displayed, it calculates appropriate limits for the two axes. A subsequent call to expl, while a previous expl is displayed on the graphics screen, uses the axis scaling that displayed expl. To have the new expl recalculate its own axis limits and not use those currently displayed, call the autoscale macro before executing expl. Alternately, the axis limit for the expl display can be specified using the scalelimits macro.

Arguments: options can be any of the following:

- 'regression' is a keyword signifying the beginning of generalized curve fitting. expl displays the data in the file regression.inp as unconnected points and also uses regression.inp to create the file analyze.inp, which serves as input to analyze for curve fitting.
- 'linear', 'square', and 'log' are keywords for display of the data points against a square or logarithmic axis scale, with the exception of the results from regression. The first keyword controls the *x*-axis scale, the second the *y*-axis. The default is 'linear'.
- 'link' is a keyword to link the data points rather than a display of the theoretical curve.
- 'nocurve' is a keyword to produce a plot of data points only.
- 'tinysymbol' is a keyword to display small-scale data point symbols.
- 'nosymbol' is a keyword to produce a plot of the curve only.
- 'noclear' is a keyword to not erase the graphics screen before drawing the plot. This prevents the graphics screen from being cleared of data.
- 'oldbox' is a keyword to plot an additional curve on an existing plot. Only the first data set in the file analyze.out is plotted. The box and scale description is derived from the file expl.out in the current experiment. When the 'oldbox' option is used, a second argument is necessary to identify the curve number and data point symbol to represent the data. This second argument is a number from 1 to 6.
- 'file' is a keyword that, when followed by a file name, makes that file replace the file analyze.out as the input to expl.

line1, line2,... specify the curves to be displayed. The default is to display the first eight curves (if that many exist) along with data points.

```
Examples: expl
```

```
expl(1,3,6)
expl('oldbox',5)
expl('regression')
expl('regression',4,5)
```

See also: NMR Spectroscopy User Guide

Related: analyze Generalized curve fitting (C)

autoscale Resume autoscaling after limits set by scalelimits (M)

expfit Make least squares fit to polynomial or exponential curve (C)

pexpl Plot exponential or polynomial curves (C)

Start of chart (P)

Sc2 Start of chart in second direction (P)
Scalelimits Set limits for scales in regression (M)

WC Width of chart (P)

wc2 Width of chart in second direction (P)

# expladd Add another diffusion analysis to current display (M)

Applicability: Systems with the diffusion option.

Syntax: expladd(integral region)

Description: Adds results of another diffusion analysis to the currently displayed results.

Arguments: integral region specifies the number of the region whose results are to

be added to the existing graph.

Examples: expladd(1)

See also: NMR Spectroscopy User Guide

Related: expl Display exponential or polynomial curves (C)

pexpl Plot exponential or polynomial curves (C)
pexpladd Add another diffusion analysis to current plot (M)

# explib Display experiment library (M)

Description: Displays the currently available experiment files. For each experiment,

explib displays the name of the experiment and its subexperiments, whether an acquisition is active or its position in the acquisition queue, the current size of the experiments, the pulse sequence currently active in the experiments, and the first 50 characters of the text file in the experiment. explib also displays

a message if the system is in automation mode.

See also: NMR Spectroscopy User Guide; VnmrJ Walkup

## explist Display current experiment chain and approx. time for each (M)

See also: Displays approximate time for each experiment in a chained experiment.

Related: autotime Display approximate time for automation (M)

#### explog Display log file for experiment (M)

Description: Displays the log file for an experiment. This file includes when the experiment

started, any acquisition errors that may have occurred, and when the experiment finished. Each acquisition generates this information, which is stored in the

experiment's acqfil directory in a text file named log.

See also: NMR Spectroscopy User Guide

# exptime Display experiment time (C)

Syntax: exptime<(sequence)><:\$seconds>

Description: Estimates the acquisition time for an experiment, based on the parameters used

in the current experiment, and displays the time in the format hh: mm: ss. The

time macro uses exptime to determine the time of an experiment.

Arguments: sequence is a pulse sequence that exists in the seqlib directory. If this

argument is used, exptime estimates the acquisition time for the specified

sequence. The default is the current value of seqlib.

\$seconds is a return argument with the number of seconds estimated for the

experiment. If this argument is used, the time display is suppressed.

Examples: exptime

exptime('apt')
exptime:\$etime

exptime('noesy'):\$est time

See also: NMR Spectroscopy User Guide

Related: time Display experiment time or recalculate number of transients (M)

# F

f Set display parameters to full spectrum (C) f19 Automated fluorine acquisition (M) f19p Process 1D fluorine spectra (M) f1coef Coefficient to construct F1 interferogram (P) f2coef Coefficient to construct F2 interferogram (P) fattn Fine attenuator (P) fb Filter bandwidth (P) fbc Apply baseline correction for each spectrum in an array (M) Set, write 1D FDM parameters, run FDM (M) fdm1 fiddc3d 3D time-domain dc correction (P) fiddle Perform reference deconvolution (M) fiddled Perform reference deconvolution subtracting alternate FIDs (C) fiddleu Perform reference deconvolution subtracting successive FIDs (C) fiddle2d Perform 2D reference deconvolution (C) fiddle2D Perform 2D reference deconvolution (C) fiddle2dd 2D reference deconvolution subtracting alternate FIDs (C) fiddle2Dd 2D reference deconvolution subtracting alternate FIDs (C) fidmax Find the maximum point in an FID (C) fidpar Add parameters for FID display in current experiment (M) fidsave Save data (M) fifolpsize FIFO loop size (P) File name of parameter set (P) file files Interactively handle files (C) filesinfo Return file information for files display (C) filtfile File of FIR digital filter coefficients (P) findxmlmenu Find an xml menu (M) fitspec Perform spectrum deconvolution (C, U) Convert gauss/cm value to DAC (M) fixgrd fixpar Correct parameter characteristics in experiment (M) fixpar3rf Create parameters for third rf channel (M) fixpar4rf Create parameters for fourth rf channel (M) fixpar5rf Create parameters for fifth rf channel (M) fixup Adjust parameter values selected by setup macros (M) fixpsg Update psg libraries (M) flashc Convert compressed 2D data to standard 2D format (C) flipflop Set up parameters for FLIPFLOP pulse sequence (M) Set up parameters for 19F experiment (M) Fluorine flush Write out data in memory (C) fn Fourier number in directly detected dimension (P) fn1 Fourier number in 1st indirectly detected dimension (P) fn2 Fourier number in 2nd indirectly detected dimension (P) fn2D Fourier number to build up 2D DOSY display in freq. domain (P) Send keyboard focus to input window (C)

focus

F

fontselect Open FontSelect window (C)

Format a real number or convert a string for output (C)

fp Find peak heights or phases (C)

fpmult First point multiplier for np FID data (P)

fpmult1 First point multiplier for ni interferogram data (P)
fpmult2 First point multiplier for ni2 interferogram data (P)

**fr** Full recall of a display parameter set (M)

**fread** Read parameters from file and load them into a tree (C)

fsave Save parameters from a tree to a file (C)
fsq Frequency-shifted quadrature detection (P)

Fourier transform 1D data (C)

ft1dFourier transform along  $f_2$  dimension (C)ft1daFourier transform phase-sensitive data (M)ft1dacCombine arrayed 2D FID matrices (M)

**ft2d** Fourier transform 2D data (C)

ft2da Fourier transform phase-sensitive data (M)
ft2dac Combine arrayed 2D FID matrices (M)

Perform a 3D Fourier transform on a 3D FID data set (M,U)

full Set display limits for a full screen (C)

fullsq Display largest square 2D display (M)

**fullt** Set display limits for a full screen with room for traces (C)

# f Set display parameters to full spectrum (C)

Description: Sets up the sp and wp display parameters for a full display of a 1D spectrum.

If an FID is displayed, the parameters sf and wf are set for a full display. In multidimensional data sets, the parameters for both displayed dimensions are set up. For 2D data sets, the parameters sp, wp, sp1, and wp1 would be set. For planes of higher dimensional data sets, the appropriate two groups of sp-

wp, sp1-wp1, and sp2-wp2, parameter pairs are set.

See also: NMR Spectroscopy User Guide

Related: sf Start of FID (P)

sp Start of plot in directly detected dimension (P)
sp1 Start of plot in 1st indirectly detected dimension (P)
sp2 Start of plot in 2nd indirectly detected dimension (P)

wf Width of FID (P)

wp Width of plot in directly detected dimension (P)wp1 Width of plot in 1st indirectly detected dimension (P)wp2 Width of plot in 2nd indirectly detected dimension (P)

# £19 Automated fluorine acquisition (M)

Syntax: f19<(solvent)>

Description: Prepares parameters for automatically acquiring a standard <sup>19</sup>F spectrum. The

parameter wexp is set to 'procplot' for standard processing. If £19 is used as the command for automation via the enter program, then the macro au is

supplied automatically and should not be entered on the MACRO line of the enter program. However, it is possible to customize the standard f19 macro on the MACRO line by following it with additional commands and parameters. For example, f19 nt=1 uses the standard f19 setup but with only one

transient.

Arguments: solvent is the name of the solvent. In automation mode, the solvent is

supplied by the enter program. The default is 'CDCl3'

Examples: f19

f19('DMSO')

See also: NMR Spectroscopy User Guide

Related: au Submit experiment to acquisition and process data (M)

enter Enter sample information for automation run (C)

f19p Process 1D fluorine spectra (M)

proc1d Processing macro for simple (non-arrayed) 1D spectra (M)

procplot Automatically process FIDs (M)
wexp When experiment completes (P)

### f19p Process 1D fluorine spectra (M)

Description: Processes non-arrayed 1D fluorine spectra using a set of standard macros. f19p

is called by procld, but can also be used directly. Fully automatic processing (up to a point where a spectrum could be plotted) is provided: Fourier transformation (using preset weighting functions), automatic phasing (aphx macro), select integral regions (hregions macro), adjust integral size (integrate macro), vertical scale adjustment (vsadjc macro), avoiding excessive noise (noislm macro), threshold adjustment (if required, thadj macro), and referencing to the TMS signal, if present (tmsref macro).

See also: NMR Spectroscopy User Guide

Related: aphx Perform optimized automatic phasing (M)

f19 Automated fluorine acquisition (M)

hregions Select integral regions for proton spectra (M)
integrate Automatically integrate 1D spectrum (M)

noislm Avoids excessive noise (M)

proc1d Processing macro for simple (non-arrayed) 1D spectra (M)

thadj Adjust threshold (M)

tmsref Reference spectrum to TMS line (M)
vsadjh Adjust vertical scale for proton spectra (M)

### flcoef Coefficient to construct F1 interferogram (P)

Description: Holds the coefficient to construct an F1 interferogram for 2D and 3D

transformation. Coefficients are used by the ft2da and ft3d macros. If f1coef has a null value, ft2da uses the "standard" coefficients. f1coef is

created by the par2d macro.

Values: Series of coefficients, separated by spaces (not a comma), and stored as a string

variable. For example, the coefficient for standard States-Hypercomplex data

set is flcoef='1 0 0 0 0 0 -1 0'.

See also: NMR Spectroscopy User Guide

Related: f2coef Coefficient to construct F2 interferogram (P)

ft2da Fourier transform phase-sensitive data (M)

Perform a 3D Fourier transform on a 3D FID data set (M,U)

make3dcoef Make 3D coefficients file from 2D coefficients (M) Create 2D acquisition, processing, display parameters (M) par2d

#### f2coef Coefficient to construct F2 interferogram (P)

Description: Holds the coefficient to construct an F2 interferogram for 2D and 3D

> transformation. Coefficients are used by the ft2da ('ni2') and ft3d macros. If f2coef has a null value, ft2da ('ni2') uses the "standard"

coefficients. f2coef is created by the par3d macro.

Series of coefficients, separated by spaces (not a comma), and stored as a string

variable. For example, the coefficient for standard States-Hypercomplex data

set is f2coef='1 0 0 0 0 0 -1 0'.

#### fattn Fine attenuator (P)

Description: Configuration parameter for whether the current rf channel has a fine attenuator.

The value is set using the label Fine Attenuator in the Spectrometer

Configuration window (opened from config).

0 specifies the fine attenuator is not present on the channel (Not Present choice

in Spectrometer Configuration window).

4095 specifies the fine attenuator is present on the channel (Present choice in

Spectrometer Configuration window).

See also: VnmrJ Installation and Administration; User Guide: Solids; CP/MAS

Installation

Related: config Display current configuration and possibly change it (M)

> First decoupler fine power (P) dpwrf Observe transmitter fine power (P) tpwrf

#### fb Filter bandwidth (P)

Description: Sets the bandwidth of the audio filters, which prevents noise of higher

> frequency than the spectral limits from "folding in" to the spectrum. Because the transmitter is in the center of the spectrum, the range of audio frequencies that must be filtered out is half the spectral width sw (e.g., for a spectral width of 4000 Hz, frequencies higher than ±2000 Hz should be filtered out). The audio filters have some attenuation at frequencies lower than their nominal cutoff frequency, which is the frequency at which signals have been attenuated by 3 dB (50%). This impacts on quantitative accuracy near the edges of the spectrum so that the standard value of fb is 10% more than half of sw.

fb is automatically changed whenever the spectral width sw is changed and thus is normally not a user-entered parameter. For example, typing sw=4000 automatically sets fb=2200, which is 10% more than 2000 Hz. After changing

the value of sw, fb can be changed.

Values: if sw is 500,000 or less: 1000 to 256000 Hz, 1000-Hz steps.

if sw is greater than 500,000: 256 kHz, 1 MHz.

See also: NMR Spectroscopy User Guide

Related: Spectral width in directly detected dimension (P)

> Set the filter bandwidths for multiple receivers (P) mrfb

fbc Apply baseline correction for each spectrum in an array (M)

Description: Applies bc -type baseline correction to all the spectra in an array. The partial

integral mode should be used to set integral regions to include all significant signals, while leaving blank as large an area of baseline as is possible.

See also: NMR Spectroscopy User Guide

Related: dosy Process DOSY experiments (M)

fdm1 Set, write 1D FDM parameters, run FDM (M)

Syntax: fdm1<(filename<,n1, v1<, n2, v2<...>>>)>

or

fdm1 (i) for the i-th trace

Description: Sets 1D Filter Diagonalization Method (FDM) parameters to the default values,

writes the parameters to the curexp/datdir/fdm1.inparm file, and runs

a stand-alone C++ program (/vnmr/bin/fdmld).

Arguments: filename is the FID file; the default is curexp+'acqfil/fid'.

n1, n2... is one or more following variable names (the order is arbitrary):

axis —1 (default) to reverse the spec.

cheat No cheat if cheat=1, lines are narrower if cheat<1.

cheatmore No cheatmore if cheatmore=0.
error Error threshold for throwing away poles.

fidfmt FID format: VnmrJ or ASCII.

fdm 1 for FDM; -1 for Digital or Discrete Fourier Transform.

fn Sp1D Spectrum file; default is curexp/datadir/

fdm1.parm.

Gamm Smoothing width (line broadening).

Gcut Maximum width for a pole.

idat Data type of ASCII FID file -4 for complex data, ignored if

data is in VnmrJ format.

i fid The i-th trace of the FID.

kcoef If kcoef > 0, use 'complicated' dk(k). -1 is

always preferred.

Nb Number of basis functions in a single window.

Nbc Number of coarse basis vectors.

Npower Number of spectrum data points.

Nsig Number of points to use.

Nskip Number of points to skip.

par Line list file; default is curexp/datadir/

fdm1.parm

rho rho=1 is optimal.

specfmt Spec format: VnmrJ or ASCII.

spectyp Spectrum type: complex (default), real imag, or abs.

ssw A test parameter.

to Delay of the first point.

theta Overall phase of FID (rp in radians).

wmax Maximum spectrum frequency in hertz.

wmin Minimum spectrum frequency in hertz.

v1, v2... is the value for the variable(s).

Examples: fdm1('cheat',0.8)

fdm1('Nsig',3000,'Nb',20,1'Gamm',0.5)

See also: NMR Spectroscopy User Guide

# fiddc3d 3D time-domain dc correction (P)

Description: Sets whether a 3D time-domain dc correction occurs. If fiddc3d does not

exist, it is created by the macro par3d. The time-domain dc correction occurs immediately after any linear prediction operations and before all other

operations on time-domain data.

Values: A three-character string. The default value is 'nnn'.

- The first character refers to the f<sub>3</sub> dimension (sw, np, fn), the second character refers to the f<sub>1</sub> dimension (sw1, ni, fn1), and the third character refers to the f<sub>2</sub> dimension (sw2, ni2, fn2).
- Each character may take one of two values: 'n' for no time-domain dc correction along the relevant dimension, and 'y' for time-domain dc correction along the relevant dimension.

See also: NMR Spectroscopy User Guide

Related: fn Fourier number in directly detected dimension (P)

fn1 Fourier number in 1st indirectly detected dimension (P)
fn2 Fourier number in 2nd indirectly detected dimension (P)

ft3d Perform a 3D Fourier transform (M)

ni Number of increments in 1st indirectly detected dimension (P)
ni2 Number of increments in 2nd indirectly detected dimension (P)

np Number of data points (P)

par3d Create 3D acquisition, processing, display parameters (C)

ptspec3d Region-selective 3D processing (P) specdc3d 3D spectral dc correction (P)

Sw Spectral width in directly detected dimension (P)
Sw1 Spectral width in 1st indirectly detected dimension (P)
Sw2 Spectral width in 2nd indirectly detected dimension (P)

# fiddle Perform reference deconvolution (M)

Syntax: fiddle(option<,file><,option<,file>><,start>

<,finish><,increment>)

Description: Performs reference deconvolution using a reference signal with known

characteristics to correct instrumental errors in experimental 1D or 2D spectra.

Arguments: option can be any of the following:

- 'alternate' is a keyword specifying the alternate reference phase +- (for phase sensitive gradient 2D data).
- 'autophase' is a keyword specifying to automatically adjust the phase of the reference signal.
- 'displaycf' is a keyword specifying to stop at the display of the correction function.
- 'fittedbaseline' is a keyword specifying to use cubic spline baseline correction defined by the choice of integral regions.
- 'invert' is a keyword specifying to invert the corrected difference spectrum/spectra.

- 'noaph' is a keyword specifying not to automatically adjust zero order phase of the reference region.
- 'nodc' is a keyword specifying not to use dc correction of reference region.
- 'noextrap' is a keyword specifying not to use extrapolated dispersion mode.
- 'nohilbert' is a keyword specifying not to use Hilbert transform algorithm and to use extrapolated dispersion mode reference signal unless 'noextrap' is also used as an option.
- 'normalise' is a keyword specifying to keep corrected spectrum integrals equal to that of the first spectrum.
- 'satellites' is a keyword specifying to use satellites defined in file in ideal reference region; file should be in /vnmr/satellites, and should immediately follow 'satellites' in the argument list.
- 'stop1' is a keyword specifying to stop at display of experimental reference FID.
- 'stop2' is a keyword specifying to stop at display of correction function.
- 'stop3' is a keyword specifying to stop at display of corrected FID.
- 'stop4' is a keyword specifying to stop at display of first corrected FID.
- 'verbose' is a specifying keyword to display information about processing in the main window.
- 'writecf' is a keyword specifying to write the correction function to file; the argument file must immediately follow 'writecf'.
- 'writefid' is a keyword specifying to write out corrected FID to file; if file does not begin with /, it is assumed to be in the current working directory. In the argument list, file should immediately follow 'writefid'.

file is the name of the file used with the 'satellites' and 'writefid' options.

start and finish are the indices of the first and last array elements to be processed. increment specifies the steps in which the index is to be incremented. The default is to process all the transformed spectra in an array.

See also: NMR Spectroscopy User Guide

Related: fiddled Perform reference deconvolution subtracting alternate FIDs

fiddleu Perform reference deconvolution subtracting successive FIDs

fiddle2d Perform 2D reference deconvolution
fiddle2D Perform 2D reference deconvolution

fiddle2dd Perform 2D reference deconvolution subtracting alternate FIDs fiddle2Dd Perform 2D reference deconvolution subtracting alternate FIDs

### fiddled Perform reference deconvolution subtracting alternate FIDs (C)

Description: Produces the corrected difference between successive spectra. Refer to the

description of fiddle for details.

See also: NMR Spectroscopy User Guide

Related: fiddle Perform reference deconvolution

F

fiddleu Perform reference deconvolution subtracting successive FIDs (C)

Description: Produces corrected differences between successive FIDs and the first FID.

Refer to the description of fiddle for details.

See also: NMR Spectroscopy User Guide

Related: fiddle Perform reference deconvolution

fiddle2d Perform 2D reference deconvolution (C)

Description: Functions the same as the fiddle program except fiddle2d performs 2D

reference deconvolution. Refer to the description of fiddle for details.

See also: NMR Spectroscopy User Guide

Related: fiddle Perform reference deconvolution

fiddle2D Perform 2D reference deconvolution (C)

Description: Functions the same as the fiddle program except fiddle2D performs 2D

reference deconvolution. Refer to the description of fiddle for details.

See also: NMR Spectroscopy User Guide

Related: fiddle Perform reference deconvolution

fiddle2dd 2D reference deconvolution subtracting alternate FIDs (C)

Description: Functions the same as the fiddle program except fiddle2dd performs 2D

reference deconvolution. Refer to the description of fiddle for details.

See also: NMR Spectroscopy User Guide

Related: fiddle Perform reference deconvolution

fiddle2Dd 2D reference deconvolution subtracting alternate FIDs (C)

Description: Functions the same as the **fiddle** program except fiddle2Dd performs 2D

reference deconvolution. Refer to the description of fiddle for details.

See also: NMR Spectroscopy User Guide

Related: fiddle Perform reference deconvolution

fidmax Find the maximum point in an FID (C)

Applicability: All

Syntax: fidmax<(trace)>:\$max

fidmax:\$max
fidmax(1):\$max

fidmax(arraydim):\$max

Description: fidmax finds the absolute maximum value in an FID.

Arguments: No arguments — fidmax uses the currently active FID

FID selected by df or select.

A FID index supplied as an argument.

### fidpar Add parameters for FID display in current experiment (M)

Description: Creates the FID display parameters axisf, crf, deltaf, dotflag, vpf,

and vpfi in the current experiment. Use fidpar to define these parameters

in old parameter sets (they are already defined in new parameter sets).

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to current experiment (M)

axisf

Axis label for FID displays and plots (P)

Curf

Current time domain cursor position (P)

deltaf

Difference of two time cursors (P)

dotflag

Vpf

Current vertical position of FID (P)

vpfi Current vertical position of imaginary FID (P)

# fidsave Save data (M)

Description: Macro to save data. It uses syfdir and syfname to construct the data

filename.

### fifolpsize FIFO loop size (P)

Description: Configuration parameter for the size of the FIFO loop. The size depends on

which controller board is present on the system—the Output board, the Acquisition Controller board, or the Pulse Sequence Controller board (refer to the description of the acquire statement in the manual *User Programming* for information on identifying the boards). The value is set using the label Fifo Loop Size in the Spectrometer Configuration window (opened by config).

Values: 2048

See also: VnmrJ Installation and Administration

Related: config Display current configuration and possibly change it (M)

# file File name of parameter set (P)

Description: Contains the file name of the parameter set returned by a rt or rtp command.

This parameter is reset when the go command is issued. If the system is not in automation mode (auto='n'), file is reset to the 'exp' value. If the system is in automation mode (auto='y'), file is set to the path of the

directory where the data is stored.

See also: NMR Spectroscopy User Guide

Related: auto Automation mode active (P)

go Submit experiment to acquisition (C)

rt Retrieve FID (C)
rtp Retrieve parameters (C)

# files Interactively handle files (C)

Syntax: files<(files menu)>

Description: Brings up the interactive file handling program. With this program, the mouse

and keyboard are used to copy, delete, rename, change directories, and load and save experiment data. The files command uses the graphics window to display file names. A mouse clicked on a file name selects it and the file name is displayed in reverse video. Various operations can be conducted on one or more selected files. The menus used for the files program are placed in the

standard menulib directories. Refer to the manual *NMR Spectroscopy User Guide* for more information on using menus, and refer to the manual *User* 

*Programming* for information on programming menus.

Arguments: files menu is the files menu to control the menu buttons; the default

menu is 'files main' or the last active files menu.

Examples: files

files('files dir')

See also: User Programming

Related: filesinfo Return files display information (C)

tape Control tape options of files program (P)

# filesinfo Return file information for files display (C)

Syntax: (1) filesinfo('number'): \$number files

(2) filesinfo('name'<, file number>):\$file

(3) filesinfo('redisplay')

Description: Allows access to the list of files selected from the files interactive display.

filesinfo is normally used only by the macros that implement the menu functions of the file system and not entered from the keyboard. The command

will not execute unless the files program is active.

Arguments: 'number' is a keyword to return the number of files selected in the files

display, or 0 if no files have been selected.

\$number files is the return variable when 'number' is used.

'name' is a keyword to return a list of file names selected in the files

display.

 $\verb|file_number| is a number following the \verb|'name'| keyword to return only the$ 

file name in the list given by file number.

\$file is a string variable that returns the file name when 'name' is used.

'redisplay' is a keyword that causes the current contents of the directory to be displayed. This display is useful after making changes in the directory,

such as deleting or creating a file.

See also: User Programming

Related: files Interactively handle files (C)

# filtfile File of FIR digital filter coefficients (P)

Description: Specifies name of a file of FIR (finite impulse response) digital filter

coefficients. This file is a text file with one real filter coefficient per line (complex filters are not supported). If the parameter filtfile does not exist

in the current experiment, enter addpar ('downsamp') or

addpar('oversamp') to add it. Entering addpar('downsamp')
creates the digital filtering and downsampling parameters downsamp,
dscoef, dsfb, dslsfrq, and filtfile. Similarly, entering
addpar('oversamp') creates digital filtering and oversampling

parameters def\_osfilt, filtfile, oscoef, osfb, osfilt,

oslsfrq, and oversamp.

Values: File name. The file must be in the user's vnmrsys/filtlib directory.

Related: addpar Add selected parameters to current experiment (M)

def osfilt Default value of osfilt (P)

downsamp Downsampling factor applied after digital filtering (P)

dscoef Digital filter coefficients for downsampling (P) dsfb Digital filter bandwidth for downsampling (P) dslsfrq Bandpass filter offset for downsampling (P) oscoef Digital filter coefficients for oversampling (P) osfb Digital filter bandwidth for oversampling (P) osfilt Oversampling filter for real-time DSP (P) Bandpass filter offset for oversampling (P) oslsfrq Oversampling factor for acquisition (P) oversamp

pards Create additional parameters used for downsampling (M)

paros Create additional parameters used for oversampling (M)

# findxmlmenu Find an xml menu (M)

Description: Find an xml menu. Used by the menu system to find and display VnmrJ menus.

### fitspec Perform spectrum deconvolution (C, U)

Syntax: (From VnmrJ) fitspec<(<'usell'><,><'setsfreq'>)>

(From UNIX) fitspec

Description: Fits experimental data to Lorentzian and/or Gaussian lineshapes. fitspec

uses as a starting point data in a file fitspec.inpar, which must be prepared prior to performing the calculation. This file contains the frequency, intensity, linewidth, and (optionally) the Gaussian fraction of the lineshape. Any number followed by an asterisk (\*) is held fixed during the calculation; all other parameters are varied to obtain the best fit. fitspec creates a file

fitspec.data, which is a text representation of the spectral data (that part of the spectrum between sp and sp+wp). After the calculation is finished, the results of the fit are contained in a file fitspec.outpar, with a format identical to fitspec.inpar.

It is often useful to use the output from a deconvolution as the input to a spin simulation to ensure the most accurate possible frequencies for the spin simulation calculation. For this reason, the frequencies and amplitudes of the calculated lines in a deconvolution are automatically stored in the parameters <code>slfreq</code>, respectively, from where they can serve as input to an iterative spin simulation. If the spin system is defined *after* a deconvolution is performed, this information is lost (<code>slfreq</code> is reset). In this case,

fitspec('setslfreq') can be used to copy the information from fitspec.outpar back into slfreq. This is not necessary if you define the spin system before performing the deconvolution (you need not perform the entire spin simulation, only define the spin system).

Arguments: 'usell' is a keyword to prepare the file fitspec.inpar from the last line

listing (stored in llfrq and llamp). All lines are set to have a linewidth of slw and a fixed Gaussian fraction of 0. If another starting point is desired, this file can be edited with a text editor. Alternatively, the macro usemark may be

used.

'setslfreg' is a keyword to copy the information from the file

 $\verb|fitspec.outpar| back into the parameters \verb|slfreq|.$ 

Examples: fitspec

fitspec('usell')
fitspec('setslfreq')

See also: NMR Spectroscopy User Guide

Related: llamp List of line amplitudes (P)

llfrq List of line frequencies (P)

setgauss Set a Gaussian fraction for lineshape (M)

slfreq Measured line frequencies (P)

sp Start of plot (P)

usemark Use "mark" output as deconvolution starting point (M)

wp Width of plot (P)

fixgrd Convert gauss/cm value to DAC (M)

Syntax: fixgrd(gradient value):parameter

Description: Uses the gcal value in the probe table to return the DAC value for a specified

gradient strength.

Arguments: gradient value is the required gradient strength in gauss/cm.

parameter is any local variable or VnmrJ variable.

Examples: fixgrd(20):gzlvl

Related: gcal Gradient calibration constant (P)

### fixpar Correct parameter characteristics in experiment (M)

Description: After bringing parameters into the current experiment with convert, rt,

rtp, or rtv, fixpar is automatically executed. fixpar updates old
parameter characteristics and reconciles parameter differences due to the
hardware on the spectrometer. If a macro userfixpar exists, fixpar runs

it also. This allows an easy mechanism to customize parameter sets.

Related: convert data set from a VXR-style system (C)

fixpar3rf Create parameters for third rf channel (M)
fixpar4rf Create parameters for fourth rf channel (M)

parfix Update parameter set (M)
parversion Version of parameter set (P)

rt Retrieve FIDs (C) rtp Retrieve parameters (C)

rtv Retrieve individual parameters (C)

updatepars Update all parameter sets saved in a directory (M)

userfixpar Macro called by fixpar (M)

# fixpar3rf Create parameters for third rf channel (M)

Applicability: Systems with a second decoupler.

Description: Checks for the existence of all acquisition parameters related to the second

decoupler. Any parameters found to be absent are created, characterized, and initialized by the macro. fixpar3rf is run as a part of the standard fixpar macro if the system configuration parameter numrfch is greater than 2 (i.e.,

the number of rf channels on the system is set at 3 or more).

# fixpar4rf Create parameters for fourth rf channel (M)

Applicability: Systems with a third decoupler.

Description: Checks for the existence of all acquisition parameters related to the third

decoupler. Any parameters found to be absent are created, characterized, and initialized. fixpar4rf is run as a part of the standard fixpar macro if the system configuration parameter numrfch is greater than 3 (i.e., the number of

rf channels on the system is set at 4).

### fixpar5rf Create parameters for fifth rf channel (M)

Applicability: Systems with a deuterium decoupler channel as the fourth decoupler.

Description: Checks for the existence of all acquisition parameters related to the fourth

decoupler. Any parameters found to be absent are created, characterized, and initialized. fixpar5rf is run as a part of the standard fixpar macro if the system configuration parameter numrfch is greater than 4 (i.e., the number of

rf channels on the system is set at 5).

### fixup Adjust parameter values selected by setup macros (M)

Description: Called by the experiment setup macros h1, c13, hc, hcapt, capt, and

hcosy. As provided, the text of fixup is all in quotes so that it does nothing. It is intended to provide each user with a mechanism to make adjustments to

values selected by the setup macros.

### fixpsg Update psg libraries (M)

Description: Used by patchinstall to recompile the psg files and create new psg

libraries libpsglib.so in /vnmr/lib.

### flashc Convert compressed 2D data to standard 2D format (C)

Syntax: flashc(<'nf'>,'ms'|'mi'|'rare',ns,traces,echoes)

Description: Converts 2D FID data files from compressed formats (seqcon='nncsn',

seqcon='nccnn', seqcon='nnccn') to standard format

(seqcon='ncsnn') or from standard format to compressed format.

Compressed data is taken by using the  ${\tt nf}$  parameter; that is, compressed data

is acquired as one large uninterrupted "multiFID" acquisition.

flashc reads the file fid in the acqfil subdirectory of the current experiment.

flashc can convert a compressed-compressed multislice, multiecho, or multiimage sequence. It can also convert a "rare" type sequence with a compressed phase-encode echo train.

flashc changes the values of the following parameters:

Compressed-compressed or standard format to compressed format

- ni is set to 1 if no argument is provided.
- nf is set to the value of nf divided by the multislice, ms, or multi-image, mi, value.
- arraydim is set to the product of its original value and the value of the traces argument.
- arrayelemts is set to 1 if no parameters were arrayed during data acquisition or to 2 if any parameter was arrayed during data acquisition.

Compressed format to standard format

- nf is set to the value of the traces argument, or to 1 if no argument is provided.
- ni is set to the value of nf divided by the multislice, ms, or multi-image, mi, value.
- arraydim is set to the product of its original value and the original value of nf.
- arrayelemts is set to 1 if no parameters were arrayed during data acquisition or to 2 if any parameter was arrayed during data acquisition.

Arguments: nf is the number of FIDs in the second dimension of a 2D experiment. When converting data in the standard format to a compressed format, nf must always be the first argument.

> When converting compressed-compressed or "rare" type sequences, the first argument must be a string defining the type of compression:

- 'mi' is a keyword for the multi-image type of compression.
- 'ms' is a keyword for the multislice type of compression.
- 'rare' is a keyword for the "rare" multiecho, rare type, fast-imaging data

(Standard to compressed) ns is the number of images slices or array elements to be retained.

(Compressed-compressed or rare to standard) traces is the number of compressed traces to retain for each ni. The parameter nf is set to this number after flashc has run.

(Compressed-compressed or rare to standard) echoes is the number of compressed echoes, used with "rare" type formatting.

Examples: Compressed-compressed or standard format to compressed format

flashc('nf') (standard to compressed)

flashc('nf','ms',ns) (compressed phase-encode and multislice) flashc('nf','mi',ns) (compressed multi-image and phase-encode)

Compressed-compressed format or rare format to standard format

flashc (simple compressed phase-encode)

flashc('ms', ns) (compressed phase-encode and multislice) flashc('mi', ns) (compressed multi-image and phase-encode)

flashc('rare',ns,etl)

See also: VnmrJ Imaging NMR

Related: arraydim Dimension of experiment (P)

> ft2d Fourier transform 2D data (C) ft3d Fourier transform 3D data (C)

nf Number of FIDs (P)

Number of increments in 1st indirectly detected dimension (P) ni

#### Set up parameters for FLIPFLOP pulse sequence (M) flipflop

Applicability: Systems with solids module.

Description: Sets up a multipulse parameter set for tuning out "phase glitch" in the probe and

pulse amplifier.

User Guide: Solid-State NMR See also:

#### Fluorine Set up parameters for 19F experiment (M)

Description: Set Up parameters for <sup>19</sup>F experiment.

#### flush Write out data in memory (C)

Writes out the current data and parameters in memory buffers. Normally, this Description:

> information is not written to disk until exiting VnmrJ or joining another experiment. One reason to use flush is to be able to access experimental data

from a program separate from the VnmrJ program.

See also: User Programming

### fn Fourier number in directly detected dimension (P)

Description: Selects the Fourier number for the Fourier transformation along the directly

detected dimension. This dimension is often referred to as the  $\ensuremath{f_2}$  dimension in

2D data sets, the f<sub>3</sub> dimension in 3D data sets, etc.

Values: 'n' or a number equal to a power of 2 (minimum is 32). If fn is not entered

exactly as a power of 2, it is automatically rounded to the nearest higher power of 2 (e.g., setting fn=32000 gives fn=32768). fn can be less than, equal to, or greater than np, the number of directly detected data points:

• If fn is less than np, only fn points are transformed.

• If fn is greater than np, fn minus np zeros are added to the data table

("zero-filling").

• If fn='n', fn is automatically set to the power of 2 greater than or equal to np.

### fn1 Fourier number in 1st indirectly detected dimension (P)

Description: Selects the Fourier number for the Fourier transformation along the first

indirectly detected dimension. This dimension is often referred to as the  $\mathbf{f}_1$  dimension of a multi-dimensional data set. The number of increments along this

dimension is controlled by the parameter ni.

Values: fn1 is set in a manner analogous to the parameter fn, with np being substituted

by 2\*ni.

See also: NMR Spectroscopy User Guide

Related: fn Fourier number in directly detected dimension (P)

fn2 Fourier number in 2nd indirectly detected dimension (P)

ni Number of increments in 1st indirectly detected dimension (P)

np Number of data points (P)

### fn2 Fourier number in 2nd indirectly detected dimension (P)

Description: Selects the Fourier number for the Fourier transformation along the second

indirectly detected dimension. This dimension is often referred to as the  $f_2$  dimension of a multidimensional data set. The number of increments along this dimension is controlled by the parameter  $\tt ni2.\ fn2$  is set in a manner analogous

to the parameter fn, with np being substituted by 2\*ni2.

See also: NMR Spectroscopy User Guide

Related: fn Fourier number in directly detected dimension (P)

fn1 Fourier number in 1st indirectly detected dimension (P)

ni2 Number of increments in 2nd indirectly detected dimension (P)

Number of data points (P)

# fn2D Fourier number to build up 2D DOSY display in freq. domain (P)

Description: In 2D DOSY sequences (Dbppste, DgcsteSL, Doneshot, Dbppsteinept),

replaces fn when setting up the 2D display.

See also: NMR Spectroscopy User Guide

Related: ddif Synthesize and display DOSY plot (C)

dosy Process DOSY experiments (M)

# F

focus Send keyboard focus to input window (C)

Description: Sends keyboard focus to the input window. This is only useful for macro

programming.

See also: User Programming

foldcc Fold INADEQUATE data about two-quantum axis (C)

Syntax: foldcc

Description: Symmetrizes 2D INADEQUATE data along the P-type double-quantum axis

and applies an automatic dc baseline correction. folder functions for both

hypercomplex and complex 2D data.

See also: NMR Spectroscopy User Guide

Related: dc Calculate spectral drift correction (C)

foldj Fold J-resolved 2D spectrum about  $f_1$ =0 axis (C) foldt Fold COSY-like spectrum along diagonal axis (C)

rotate Rotate 2D data (C)

foldj Fold J-resolved 2D spectrum about  $f_1=0$  axis (C)

Description: Symmetrizes heteronuclear 2D-J, or rotated homonuclear 2D-J, experiments

about the  $f_1$ =0 axis. The foldj command functions with both complex and

hypercomplex 2D data.

Related: foldcc Fold INADEQUATE data about 2-quantum axis (C)

foldt Fold COSY-like spectrum along diagonal axis (C)

rotate Rotate 2D data (C)

foldt Fold COSY-like spectrum along diagonal axis (C)

Syntax: foldt<('symm'|'triang')>

Description: Folds COSY-like correlation spectra about the diagonal. The 2D spectrum must

exhibit a *P-type diagonal* for foldt to work properly (a P-type diagonal goes from the bottom left-hand side to the top right-hand side of the contour display.) foldt functions for both hypercomplex and complex 2D data but requires that

fn=fn1 and sw=sw1.

Arguments: 'symm' is a keyword for the folding process to perform a symmetrization of

the data by replacing every two symmetry-related points with the one point

therein that has the least magnitude. This value is the default.

'triang' is a keyword for the folding process to perform a triangularization of the data by replacing every two symmetry-related points with their geometric

mean.

Related: fn Fourier number in directly detected dimension (P)

fn1Fourier number in 1st indirectly detected dimension (P)foldceFold INADEQUATE data about 2-quantum axis (C)foldjFold J-resolved 2D spectrum about  $f_I$ =0 axis (C)

rotate Rotate 2D data (C)

Sw Spectral width in directly detected dimension (P)
Sw1 Spectral width in 1st indirectly detected dimension (P)

#### fontselect Open FontSelect window (C)

Description: Opens the FontSelect window for defining fonts in window panes created by

> setgrid. A different font can be selected for every window pane combination of rows and columns. Separate fonts can also be selected for a large or small

overall graphic window.

See also: NMR Spectroscopy User Guide

Related: curwin Current window (P)

> jwin Activate current window (M) mapwin List of experiment numbers (P) setgrid Activate selected window (M) setwin Activate selected window (C)

#### format Format a real number or convert a string for output (C)

Applicability: All

Syntax: format(realvar, 'length', 'precision'):\$rval

format(stringvar, 'isreal'):\$rval

format(stringvar,<'upper' or 'lower'>):\$sval

Description: Give the command these arguments to format the output as a real number.

Accepts arguments specifying the real number length, precision, and output variable. The realvar input must be a real number, string holding a real number, or a string variable that satisfies the rules for a real number.

Give the command two arguments, stringvar and isreal to test the string. The command returns a 1 if the first argument can represent a real number and a 0 if it cannot. The output is written to the specified output variable.

Give the command two arguments, stringvar and either 'upper' or 'lower' to write the string to the output string variable (e.g., \$sval) in either all upper case or all lower case. The temporary \$ parameter (e.g., \$sval) must

first be initialized as a string (e.g., \$sval='').

Arguments: realvar a real variable

stringvar a string variable length of the real number precision of the real number \$sval a temporary string \$ parameter \$rval a temporary real \$ parameter

Examples: format(a,5,2):sa

If a=24.1264 then string sa='24.13'

format(solvent, 'lower'):n1

If solvent='CDCl3' then string n1='cdcl3'

format(\$1,'isreal'):\$a

Sets \$a to 1 if \$1 represents a number or Sets \$a to 0 if \$1 represents a string.

Initialize \$sval to a string variable to return the value into a string

\$snum = '143.92' \$rnum = 32.75

format(\$rnum,3,1):\$sval

sets \$sval to the string '32.8'., \$sval is a string return value.

format(\$rnum,3,1):\$rval

sets Srval to the number 32.8. Srval is a real return value.

Format string value \$snum = '143.92'

format (\$snum, 3, 1):\$sval sets \$sval to the string '143.9'

```
format ($snum, 3, 1): $rval sets $rval to the number 143.9
```

See also: User Programming

Related: n1, n2, n3 Name storage for macros (P)
r1-r7 Real-value storage for macros (P)

# fp Find peak heights or phases (C)

```
Syntax: fp<(<'phase',><index1,index2,...>)>
```

Description: Following a line listing (either dll or nll), fp measures the peak height of

each peak in an array of spectra. The results of the analysis are written to a text file fp.out in the current experiment directory. If the **npoint** parameter is defined in the current parameter set and this parameter is "on," it determines the range of data points over which a maximum is searched when determining peak heights. The possible values of **npoint** are 1 to fn/4. The default is 2.

Arguments: 'phase' is a keyword to measure the phase of each peak instead of height.

index1, index2, ... restricts measuring peak heights or phases to the lines

listed.

Examples: fp

fp(1,3)
fp('phase')

See also: NMR Spectroscopy User Guide

Related: dll Display listed line frequencies and intensities (C)

fn Fourier number in directly detected dimension (P)

getll Get line frequency and intensity from line list (C)

nl Position cursor at the nearest line (C)
nll Find line frequencies and intensities (C)
npoint Number of points for fp peak search (P)

### fpmult First point multiplier for np FID data (P)

Description: Allows error correction if the first point of an FID is misadjusted. In a 1D

experiment, this adjustment influences the overall integral of the spectrum. For n-dimensional experiments, if the correction is not made, "ridges" can appear. In 2D experiments, the ridges appear as " $f_2$  ridges." In 3D experiments, the ridges appear as " $f_3$  ridges." These ridges can clearly be seen in the noise region on the top and bottom of a 2D spectrum (when <code>trace='f1'</code>) as a lowintensity profile of the diagonal. The sign and intensity of the ridges is

controlled by the magnitude of fpmult.

It has been recognized that the first point of a FID that is sampled at exactly time equal to zero must be multiplied by 0.5 for the Fourier transform to function properly. The fpmult parameter gives you a method to fine-tune the actual

correction factor.

Values: Default is 1.0, except that if the processing involves backward extension of the

time-domain data with linear prediction, the default changes to 0.5. If fpmult

is set to 'n', fpmult takes on its default value.

See also: NMR Spectroscopy User Guide

Related: fpmult1 First point multiplier for ni interferogram data (P)

fpmult2 First point multiplier for ni2 interferogram data (P)

np Number of data points (P)

trace Mode for *n*-dimensional data display (P)

wft2da Weight and Fourier transform phase-sensitive data (M)

### fpmult1 First point multiplier for ni interferogram data (P)

Description: Operates on ni hypercomplex or complex interferogram data in a manner

analogous to fpmult. In many 2D experiments, the  $t_1$  values are adjusted so there is no first-order phasing in the  $f_1$  and  $f_2$  dimensions. In this case, fpmult1 should be 0.5. If the  $t_1$  value is adjusted so that there is a 180° first-

order phase correction, fpmult1 should be 1.0.

Values: Default value is 0.5. If fpmult1 is set to 'n', it takes on its default value.

See also: NMR Spectroscopy User Guide

Related: fpmult First point multiplier for np FID data (P)

fpmult2 First point multiplier for ni2 interferogram data (P)

Number of increments in 1st indirectly detected dimension (P)

### fpmult2 First point multiplier for ni2 interferogram data (P)

Description: Operates on ni2 hypercomplex or complex interferogram data in a manner

analogous to fpmult. In many 3D experiments, the t<sub>2</sub> value is adjusted so that

there is no first-order phasing in the  $f_1$  and  $f_2$  dimensions. In this case,

fpmult2 should be 0.5. If the t<sub>2</sub> value is adjusted so that there is a 180° first-

order phase correction, fpmult2 should be 1.0.

Values: Default value is 0.5. If fpmult2 is set to 'n', it takes on its default value.

See also: NMR Spectroscopy User Guide

Related: fpmult First point multiplier for np FID data (P)

fpmult1 First point multiplier for ni interferogram data (P)

Number of increments in 2nd indirectly detected dimension (P)

### fr Full recall of a display parameter set (M)

Applicability: All

Syntax: fr(n<,noupdate>)

Description: Recall all the parameters of the specified display parameter set and set the

current display parameters to those values.

Arguments: n=1 to 9

noupdate as second argument prevents the automatic update of interactive

programs.

Related: r Recall display parameter set (M)

Save display parameters as a set (M)

### fread Read parameters from file and load them into a tree (C)

Syntax: fread(file<,tree<,'reset|'value'>>)

Description: Reads parameters from a file and loads the parameters into a tree. The tree can

be global, current, processed, or system global. fread can read from any file

that has parameters stored in the correct VnmrJ format.

Note that if parameters are read into the global tree, certain important system parameters are not loaded because these parameters should not be changed. The parameters that are not loaded are userdir, systemdir, curexp,

autodir, auto, vnmraddr, and acqaddr.

Arguments: file is the name of the file containing parameters stored in VnmrJ format.

tree is one of the keywords 'global', 'current', 'processed', or 'systemglobal'. The default is 'current'. This argument specifies the

type of tree into which the parameters are loaded. Refer to the create command for more information on types of trees.

'reset' is a keyword that causes the parameter tree to be cleared before the new parameter file is read. Without this option, parameters read from a file are added to the existing preloaded parameters. To use this option, tree must also be specified.

'value' is a keyword that causes only the values of the parameters in the file to be loaded. If a preloaded variable does not already exist, a new one is not created. Parameter attributes are not changed, and enumerated values are not changed. To use this option, tree must also be specified.

Examples: fread('/vnmr/stdpar/H1.par/procpar')

fread('sampvar','global')

fread('setvar','current','reset')
fread('var1','processed','value')

See also: User Programming

Related: auto Automation mode active (P)

autodir Automation directory absolute path (P)

create Create new parameter in a parameter tree (C)

current experiment directory (P)

destroy Destroy a parameter (C)

display Display parameters and their attributes (C)

fsave Save parameters from a tree to a file (C)

rtp Retrieve parameters (C)
systemdir System directory (P)
userdir User directory (P)

# fsave Save parameters from a tree to a file (C)

Syntax: fsave(file<,tree>)

Description: Writes parameters from a parameter tree to a file.

Arguments: file is the name of the file, which can be any valid file for which the user has

write permission. If the file already exists, it will be overwritten.

tree is one of the keywords 'global', 'current', 'processed', or 'systemglobal'. The default is 'current'. Refer to the create

command for more information on types of trees.

Examples: fsave('var1')

fsave('sampvar','global')

See also: User Programming

Related: create Create new parameter in a parameter tree (C)

destroy Destroy a parameter (C)

display Display parameters and their attributes (C)

**fread** Read parameters from file and load them into a tree (C)

Save parameters from current experiment (C)

# fsq Frequency-shifted quadrature detection (P)

Description: Selects whether to use frequency-shifted quadrature detection. When fsq is

turned on, if dsp is on, the observe frequency is offset by oslsfrq, and the digital filter is also offset by oslsfrq. The default value of oslsfrq is

1.25\*sw.

The effect of fsq is to offset only the digital filter by oslsfrq. The observe frequency must be offset by oslsfrq by modifying the pulse sequence as described in the manual *NMR Spectroscopy User Guide*.

Values: 'n' turns frequency-shifted quadrature detection off. 'y' turns it on.

See also: NMR Spectroscopy User Guide

Related: dsp Type of DSP for data acquisition (P)

oslsfrq
Bandpass filter offset for oversampling (P)
oversamp
Oversampling factor for acquisition (P)

Spectral width in directly detected dimension (P)

# ft Fourier transform 1D data (C)

Syntax: (1) ft<(<options,><'nf'><,start><,finish><,step>)>

(2) ft('inverse', exp number, expansion factor)

Description:

In syntax 1, performs a Fourier transform on one or more 1D FIDs without weighting applied to the FID. ft executes a left-shift, zero-order phase rotation, and a frequency shift (first-order phase rotation) according to the parameters <code>lsfid</code>, <code>phfid</code>, and <code>lsfrq</code>, respectively, on the time-domain data, prior to Fourier transformation. The type of Fourier transform to be performed is determined by the parameter <code>proc</code>. Solvent suppression is turned on or off with the parameters <code>ssfilter</code> and <code>ssorder</code>. For arrayed data sets, <code>ft</code> Fourier transforms all of the array elements. To Fourier transform selected array elements, <code>ft</code> can be passed numeric arguments.

In syntax 2, ft performs an inverse Fourier transform of the entire spectrum. (VnmrJ does not currently support inverse Fourier transformation of arrayed 1D or 2D data sets.)

Arguments:

options can be any of the following (all string arguments must precede the numeric arguments):

- 'acq' is a keyword to check if any elements of a multi-FID experiment have already been transformed. If so, these previously transformed elements will not be retransformed.
- 'dodc' is a keyword for all spectra to be drift corrected independently.
- 'nodc' is a keyword to not perform the usual FID drift correction.
- 'nods' is a keyword to prevent an automatic spectral display (ds) from occurring. This outcome is useful for various plotting macros.
- 'noft' is a keyword to skip the Fourier transform, thereby allowing use of all spectral manipulation and plotting commands on FIDs.
- 'zero' is a keyword to zero the imaginary channel of the FID prior to the Fourier transform. This zeroing occurs after any FID phasing. Its use is generally limited to wideline solids applications.

'nf' is a keyword that makes a single FID element containing nf traces to be transformed as if it were nf separate FID elements. If 'nf' precedes the list of numeric arguments, the rules for interpreting the numeric arguments change slightly. Passing no numeric arguments results in the transformation of all nf traces in the first FID element. Passing a single numeric argument results in the transformation of all nf traces in the requested FID element (e.g., ft('nf',3) transforms all nf traces for element 3). Regardless of the requested FID element, the resulting spectra are labeled as 1 to nf because multiple elements cannot be transformed using ft('nf'). Subsequent numeric arguments are interpreted as previously described.

start is the index of a particular element to be transformed. For an array, start is the index of the first element to be transformed.

finish is the index of the last element to be transformed for an array. step specifies the increment between successive elements that are to be transformed for an array. The default is 1.

'inverse' is a keyword specifying an inverse Fourier transform.

exp number is the number of the experiment, from 1 to 9, for storing the resulting FID from the inverse Fourier transform.

expansion factor defines the expansion of the spectrum before the inverse Fourier transform is performed. This argument is equivalent to a multiplier for the fn parameter. The multiplier is restricted to between 1 and 32 and is rounded up internally to the nearest power of 2.

Examples: ft

ft(1) ft(3,7)ft(2,10,2) ft('nf',3)

See also: NMR Spectroscopy User Guide

Related: dcrmv Remove dc offsets from FIDs in special cases (P)

> fn Fourier number in directly detected dimension (P) lsfid Number of points to left-shift the np FID (P) Frequency shift of the fn spectrum in Hz (P) lsfrq

Number of FIDs (P) nf

Zero-order phasing constant for np FID (P) phfid proc Type of processing on the np FID (P)

Full bandwidth of digital filter to yield a filtered FID (P) ssfilter ssorder Order of polynomial to fit digitally filtered FID (P)

wft Weight and Fourier transform 1D data (C)

#### ft1d Fourier transform along f<sub>2</sub> dimension (C)

Syntax: (1) ft1d(element number)

(2) ft1d<('nf',element number)</pre>

(3) ft1d<(<options,><coefficients>)>

Description: Performs the first Fourier transformation along the f2 dimension, without

weighting, and matrix transposition. ftld allows the display of t1 interferograms with the dcon and dconi commands. For arrayed 2D FID data, a single array element can be weighted and transformed using syntax 1 or 2. The keyword 'nf' is used in syntax 2 to specify that the 2D data is collected in the compressed form using 'nf'. Complex and hypercomplex interferograms can be constructed explicitly by supplying a series of options and coefficients using syntax 3.

For information on real as opposed to complex Fourier transforms, see the descriptions of the proc, proc1, and proc2 parameters. For information about Hadamard transforms, see the description of the procl parameter and the *VnmrJ NMR Liquids* user guide. For information on left-shifting, zero-order phase rotation, and frequency shifting of the FID and interferogram timedomain data during the 2D Fourier transformation, see the descriptions of the parameters lsfid, lsfid1, lsfid2, phfid, phfid1, phfid2, lsfrq, lsfrg1, and lsfrg2, as appropriate. For information on the lfs (lowfrequency suppression) and zfs (zero-frequency suppression) solvent suppression options, see the description of the parameters ssfilter and

ssorder, and the macro parfidss.

element number is a single array element to be weighted and transformed. Arguments:

> options can be the keywords 'ptype' or 'ntype' but neither serve a useful function because the differential effect of these arguments is applied only during the course of the second Fourier transformation. The default is

'ntype'.

coefficients are a series of coefficients according to the following scheme: RR1 is the coefficient used to multiply the real part (first R) of spectra set 1 before it is added to the real part (second R) of the interferogram. IR2 would thus represent the contribution from the imaginary part of spectra set 2 to the real part of the interferogram, and so on. The scheme is depicted below.

```
ftld(RR1,IR1,RR2,IR2,...,RI1,II1,RI2,II2,...)
where:
RR1*REAL(w2,element=1) -> REAL(t1)
IR1*IMAG(w2,element=1) -> + REAL(t1)
RR2*REAL(w2,element=2) -> + REAL(t1)
IR2*IMAG(w2,element=2) -> + REAL(t1)
```

RI1\*REAL(w2,element=1) -> IMAG(t1)
II1\*IMAG(w2,element=1) -> + IMAG(t1)
RI2\*REAL(w2,element=2) -> + IMAG(t1)
II2\*IMAG(w2,element=2) -> + IMAG(t1)

. . .

See also: NMR Spectroscopy User Guide

Related: dconi Interactive 2D data display (C)

ft2d Fourier transform 2D data (C)

1sfid Number of complex points to left-shift np FID (P)

lsfid1Number of complex points to left-shift ni interferogram (P)lsfid2Number of complex points to left-shift ni2 interferogram (P)

lsfrq Frequency shift of the fn spectrum (P)
lsfrq1 Frequency shift of the fn1 spectrum (P)
lsfrq2 Frequency shift of the fn2 spectrum (P)

parfidss Create parameters for time-domain solvent subtraction (M)

phfid Zero-order phasing constant for np FID (P)

phfid1 Zero-order phasing constant for ni interferogram (P)
phfid2 Zero-order phasing constant for ni interferogram (P)

proc Type of processing on np FID (P)

proc1 Type of processing on ni interferogram (P)
proc2 Type of processing on ni2 interferogram (P)

pmode Processing mode for 2D data (P)

Ssorder Order of polynomial to fit digitally filtered FID (P)

Ssfilter Full bandwidth of digital filter to yield a filtered FID (P)

wft2d Weight and Fourier transform 2D data (C)

# ft1da Fourier transform phase-sensitive data (M)

Syntax: ft1da<(options)>

Description: Performs the first (f<sub>2</sub>) transform of a 2D transform or the first part of a 3D

transform. Otherwise, ftlda has the same functionality as the ft2da command. See the description of ft2da for further information. For information about Hadamard transforms, see the description of the procl

parameter and the VnmrJ NMR Liquids user guide.

Arguments: options are the same as used with ft2da. See ft2da for details.

See also: NMR Spectroscopy User Guide

Related: ft2d Fourier transform 2D data (C)

ft2da Fourier transform phase-sensitive data (M)

wftlda Weight and Fourier transform phase-sensitive data (M) wftlda Weight and Fourier transform phase-sensitive data (M)

### ft1dac Combine arrayed 2D FID matrices (M)

Syntax: ft1dac<(<mult1><,mult2>,...<,multn>)>

Description: Allows ready combination of 2D FID matrices within the framework of the 2D

Fourier transformation program. No weighting is performed. ftldac requires that the data be acquired either without  $f_1$  quadrature or with  $f_1$  quadrature using the TPPI method. This macro is used for TOCSY (with multiple mixing times).

Arguments: mult1, mult2, ..., multn are multiplicative coefficients. The nth

argument is a real number and specifies the multiplicative coefficient for the nth

2D FID matrix.

Related: ft2dac Combine arrayed 2D FID matrices (M)

Tocsy Set up parameters for TOCSY pulse sequence (M)
wftlda Weight and Fourier transform phase-sensitive data (M)

wftldac Combine arrayed 2D FID matrices (M)

# ft2d Fourier transform 2D data (C)

Syntax: (1) ft2d(array element)

(2) ft2d('nf'<array element>)

(3) ft2d<(<options,><plane number,><coefficients>)>

(4) ft2d('ni'|'ni2',element\_number,increment)

(5) ft2d('ni'|'ni2',increment,<coefficients>)

Description:

Performs the complete 2D Fourier transformation, without weighting, in both dimensions. If the first Fourier transformation has already been done using ftld, wftld, ftlda, or wftlda, the ftld command performs only the second  $(t_1)$  transform.

For arrayed 2D FID data, a single array element can be weighted and transformed using syntax 1. If the data is collected in "compressed" form using 'nf', syntax 2 must be used. Complex and hypercomplex interferograms can be constructed explicitly by supplying a series of coefficients using syntax 3. If an arrayed 3D data set is to be selectively processed, the format of the arguments to ft2d changes to syntax 4. For example, ft2d('ni',1,2) performs a 2D transform along np and ni of the second ni2 increment and the first element within the explicit array. This command yields a 2D np-ni frequency plane.

Arrayed 3D data sets can also be subjected to 2D processing to yield 2D absorptive spectra. If the States-Haberkorn method is used along both  $f_1$  (ni dimension) and  $f_2$  (ni2 dimension), there are generally 4 spectra per (ni,ni2) 3D element. In this case, using syntax 5, entering ft2d('ni2',2,<16 coefficients>) performs a 2D transform along np and ni2 of the second ni increment using the 16 coefficients to construct the 2D  $t_1$ -interferogram from appropriate combinations of the 4 spectra per (ni,ni2) 3D element.

If there are n data sets to be transformed, as in typical phase-sensitive experiments, 4\*n coefficients must be supplied. The first 2\*n coefficients are the contributions to the real part of the interferogram, alternating between absorptive and dispersive parts of the successive data sets. The next 2\*n coefficients are the contributions to the imaginary part of the interferogram, in the same order. Thus, using the definition that the first letter refers to the source data set, the second letter refers to the interferogram, and the number identifies the source data set, we have the following cases:

	Data sets	Coefficient order							
•	1	RR1,	IR1,	RI1,	II1				
	2	RR1,	IR1,	RR2,	IR2,	RI1,	II1,	RI2,	II2
	3	RR1,	IR1,	RR2,	IR2,	RR3,	IR3,	RI1,	II1,

RI2, II2, RI3, II3

The coefficients are often 1, 0, or -1, but this is not always the case. Any nonintegral coefficient can be used, and as many coefficients can be nonzero as is desired. Up to 32 coefficients can be supplied, which at 4 per data set allows the addition, subtraction, etc., of eight 2D data sets (e.g., 8 different phase cycles).

For information on real as opposed to complex Fourier transforms, see the descriptions of the proc, proc1, and proc2 parameters. For information about Hadamard transforms, see the description of the proc1 parameter and the *VnmrJ NMR Liquids* user guide. For information on left-shifting, zero-order phase rotation, and frequency shifting of the FID and interferogram timedomain data during the 2D Fourier transformation, see the descriptions of the parameters lsfid, lsfid1, lsfid2, phfid, phfid1, phfid2, lsfrq, lsfrg1, and lsfrg2, as appropriate. For information on the lfs (lowfrequency suppression) and zfs (zero-frequency suppression) solvent suppression options, see the description of parameters ssfilter and ssorder, and macro parfidss.

Arguments: array\_element is a single array element to be transformed.

options can be any of the following (all string arguments must precede the numeric arguments):

- 'ptype' is a keyword to transform P-type data to yield a P-type contour display.
- 'ntype' is a keyword to transform N-type data to yield a P-type contour display. This is the default.
- 't2dc' is a keyword to apply a dc correction to each t2 FID prior to the first Fourier transform. The last 1/16-th of the time domain data is used to calculate the dc level.
- 'tldc' is a keyword to apply a dc correction to each t<sub>1</sub> interferogram prior to the second Fourier transform. The last 1/16-th of the time domain data is used to calculate the dc level.
- 'f2sel' is a keyword to allow only preselected f2 regions to be transformed along t<sub>1</sub>. The t<sub>1</sub> interferograms in the non-selected f<sub>2</sub> regions are zeroed but not transformed. The same mechanism used to select baseline regions for baseline correction (bc) is used to select the f2 regions to be transformed along t<sub>1</sub>. Set intmod='partial' and partition the integral of the spectrum into several regions. The even numbered  $f_2$  regions (e.g., 2, 4, 6) are transformed along  $t_1$ ; the odd numbered regions are not transformed along t<sub>1</sub>
- 'nf' is a keyword to transform arrayed or multi-slice 2D data that has been collected in the compressed form as single 2D FIDs with multiple (nf) traces.
- 'ni2' is a keyword to transform non-arrayed 2D data that have been collected with ni2 and sw2 (instead of ni and sw1). addpar ('3d') creates the necessary processing parameters for the 'ni2' operation.
- 'noop' is a keyword to not perform any operation on the FID data. This option is used mainly to allow macros, such as wft2da, to have the same flexibility as commands.

coefficients are a series of coefficients according to the following scheme: RR1 is the coefficient used to multiply the real part (first R) of spectra set 1 before it is added to the real part (second R) of the interferogram. IR2

the real part of the interferogram, and so forth. The scheme is depicted below. ft2d(RR1, IR1, RR2, IR2, ..., RI1, II1, RI2, II2, ...) where: RR1\*REAL(w2,element=1) -> REAL(t1) IR1\*IMAG(w2.element=1) -> + REAL(t1) RR2\*REAL(w2,element=2) -> + REAL(t1)  $IR2*IMAG(w2,element=2) \rightarrow + REAL(t1)$ RI1\*REAL(w2,element=1) -> IMAG(t1)  $II1*IMAG(w2,element=1) \rightarrow + IMAG(t1)$  $RI2*REAL(w2,element=2) \rightarrow + IMAG(t1)$ II2\*IMAG(w2,element=2) -> + IMAG(t1) 'ni' is a keyword to selectively transform a particular np-ni 2D plane within a non-arrayed 3D data set. To identify the plane, 'ni' is followed by the plane number argument, an integer from 1 through ni2. 'ni2' is a keyword to selectively transform a particular np-ni2 2D plane within a non-arrayed 3D data set. To identify the plane, 'ni2' is followed by the plane number argument, an integer from 1 through ni. element number is the number of an element within the explicit array when selectively processing an arrayed 3D data set; it ranges from 1 to ni2 increment is the increment within the explicit array when selectively processing an arrayed 3D data set; it ranges 1 to arraydim/(ni\*ni2). Examples: ft2d(1,0,0,0,0,0,1,0)ft2d(1) ft2d('nf',3) ft2d('ptype',...) See also: NMR Spectroscopy User Guide Related: dconi Interactive 2D data display (C) dcrmv Remove dc offsets from FIDs in special cases (P) First point multiplier for np FID data (P) fpmult fpmult1 First point multiplier for ni interferogram data (P) ft1d Fourier transform along f<sub>2</sub> dimension (C) lsfid Number of complex points to left-shift np FID (P) Number of complex points to left-shift ni interferogram (P) 1sfid1 lsfid2 Number of complex points to left-shift ni2 interferogram (P) Frequency shift of the fn spectrum (P) lsfrq lsfrq1 Frequency shift of the fn1 spectrum (P) lsfrq2 Frequency shift of the fn2 spectrum (P) parfidss Create parameters for time-domain solvent subtraction (M) phfid Zero-order phasing constant for np FID (P) Zero-order phasing constant for ni interferogram (P) phfid1 phfid2 Zero-order phasing constant for ni2 interferogram (P) Type of processing on np FID (P) proc Type of processing on ni interferogram (P) proc1 proc2 Type of processing on ni2 interferogram (P) pmode Processing mode for 2D data (P) Order of polynomial to fit digitally filtered FID (P) ssorder ssfilter Full bandwidth of digital filter to yield a filtered FID (P) Weight and Fourier transform f<sub>2</sub> for 2D data (C) wft1d wft2d Weight and Fourier transform 2D data (C)

would thus represent the contribution from the imaginary part of spectra set 2 to

#### Fourier transform phase-sensitive data (M) ft2da

Syntax: ft2da<(options)>

Description: Processes 2D FID data and 2D planes at particular t<sub>1</sub> or t<sub>2</sub> times from a 3D data set for a pure absorptive display. ft2da differs from wft2da only in that, in the case of wftlda, weighting of the time-domain data is performed prior to the FT. ft2da functions analogously to ft1da and wft1da, except that ft2da and wft2da perform only the f<sub>2</sub> Fourier transform. For information about Hadamard transforms, see the description of the proc1 parameter and the VnmrJ NMR Liquids user guide.

> Macros ftlda, wftlda, ftlda, and wftlda function for hypercomplex 2D FID data (phase=1, 2) and for TPPI 2D FID data (phase=3 or phase=1, 4) acquired either with ni or ni2. If the data were acquired with ni, no additional arguments need be used with the macros. If the data were acquired with ni2, the keyword 'ni2' must be used.

```
For phase=1, 2:
wft2da=wft2d('ptype',1,0,0,0,0,0,1,0)
For phase=3: wft2da=wft2d(1,0,0,0)
For phase=1,4:
wft2da=wft2d('ptype',1,0,0,0,0,0,1,0)
```

Macros ft1da, wft1da, ft2da, and wft2da support selective 2D processing within a 3D FID data set. All permutations of hyercomplex and TPPI modes of data acquisition in t<sub>1</sub> and t<sub>2</sub> can be handled. For selective f<sub>2</sub>f<sub>3</sub> processing, the numeric argument immediately following the 'ni2' keyword is interpreted to be the  $t_1$  increment number, which specifies the particular  $f_2f_3$ plane (plane number, see below) to be processed. For selective  $f_1f_3$ processing, the t<sub>2</sub> increment number either follows the keyword 'ni', which is optional, or is associated with the first numeric argument that does not immediately follow a 'bc' keyword.

For information on real as compared to complex Fourier transformation, see the description of proc or proc1. For information on the lfs (low-frequency suppression) and zfs (zero-frequency suppression) solvent suppression options, see the description of parameters ssfilter and ssorder, and the macro parfidss.

Arguments: options can be any of the following (the order is not important):

- 'ntype', 't2dc', 't1dc', and 'f2sel' are keywords that function the same as when supplied to the ft2d and wft2d commands. Refer to the ft2d command for a description of these options.
- 'bc' is a keyword for a baseline correction of the phase-corrected f<sub>2</sub> spectra prior to the f<sub>1</sub> Fourier transform. The baseline regions must have been previously determined. The default polynomial order is 1, which leads to a spline fit. A different polynomial order can be specified by inserting a numerical argument following 'bc'.
- 'dc' is a keyword for a drift correction (dc) of the f<sub>2</sub> spectra prior to the f<sub>1</sub> Fourier transformation.
- 'ni' is a keyword to selectively transform a particular np-ni 2D plane within a non-arrayed 3D data set. To identify the plane, 'ni' is followed by plane number, an integer from 1 through ni2.
- 'ni2' is a keyword to selectively transform a particular np-ni2 2D plane within a non-arrayed 3D data set. To identify the plane, 'ni2' is followed by plane number, an integer from 1 through ni.
- 'old' is a keyword to allow data acquired before the February 25, 1988, software release to be processed correctly. 'old' does not function for

selective 2D processing within 3D data sets. If no ni2 or ni plane\_number is given, it is assumed that the data set is only 2D in either ni2 or ni, respectively.

See also: NMR Spectroscopy User Guide

Related: flcoef Coefficient to construct F1 interferogram (P)

f2coef Coefficient to construct F2 interferogram (P)
ft1da Fourier transform phase-sensitive data (M)

parfidss Create parameters for time-domain solvent subtraction (M)

phase Phase selection (P)

Type of processing on the np FID (P)

proc1 Type of processing on the ni interferogram (P)
ssorder Order of polynomial to fit digitally filtered FID (P)
ssfilter Full bandwidth of digital filter to yield a filtered FID (P)
wftlda Weight and Fourier transform phase-sensitive data (M)
wftlda Weight and Fourier transform phase-sensitive data (M)

### ft2dac Combine arrayed 2D FID matrices (M)

Syntax: ft2dac<(<mult1><,mult2>,...<,multn>)>

Description: Allows ready combination of 2D FID matrices within the framework of the 2D

FT program. No weighting is performed. Data must be acquired either without  $f_1$  quadrature or with  $f_1$  quadrature using the TPPI method. ft2dac is used

with TOCSY (with multiple mixing times).

Arguments: mult1, mult2,...,multn are multiplicative coefficients. The nth argument is

a real number and specifies the coefficient for the nth 2D FID matrix.

Related: ftldac Combine arrayed 2D FID matrices (M)

Tocsy Set up parameters for a TOCSY pulse sequence (M)

wft1dac Combine arrayed 2D FID matrices (M)
wft2dac Combine arrayed 2D FID matrices (M)

### ft3d Perform a 3D Fourier transform on a 3D FID data set (M,U)

Description: Transforms 3D FID data into 3D spectral data. ft3d can be entered from a

macro or directly from UNIX. Each type of entry is described below. A final

section explains the ft3d coefficient file.

Additional parameter control for the operation of ft3d is available. This allows drift corrections and partial Fourier transformation. See the descriptions of specdc3d, fiddc3d, and ptspec3d for information.

The 3D FID data must be loaded into the experiment in which the ft3d macro is to be run. ft3d is started up in background mode by this macro so that VnmrJ remains free for interactive processing. You can start a 3D transform from within exp4 and, at the same time, continue with any 1D or 2D processing of the 3D FID data within the same experiment using VnmrJ.

Distributed f<sub>1</sub>f<sub>2</sub> processing has the following system and network requirements:

 The system on which the macro ft3d is executed from within VnmrJ must define the names of the networked computers that are to participate in the distributed processing. The file /etc/hosts.3D must contain these names in the following format:

```
unity1
unity2
datastation1
datastation2
```

- Each participating computer must recognize the name of the user that started up the master ft3d program as a valid user name on its system. For example, if user steve issues the ft3d command within VnmrJ running on computer unity0, steve must be a valid user on all other computer systems that are to be used in the distributed  $f_1f_2$  processing.
- Each computer system must have NFS access to the 3D data directory.

Arguments: The order of the arguments is not important.

data directory (without the /data subdirectory appended) specifies the output directory for the 3D spectral data file(s). The default directory for the 3D spectral data is curexp/datadir3d.

number files sets the number of 3D data files (data1, data2, ... datan, where n is number files) used to store the transformed 3D data. number files must be an integer and be 32 or less. When number files is entered, distributed  $f_1f_2$  processing is performed by ft3d if possible.

'nocoef' is a keyword for the set3dproc command within the ft3d macro to not create a 3D coefficient file prior to invoking the ft3d program. This option is useful if you have modified an existing 3D coefficient file and do not want it to be overwritten prior to the 3D transform. See below for information on coefficient files. By default, ft3d calls the make3dcoef macro to create a coefficient file using the flcoef and flcoef string parameter values.

't1t2' and 't2t1' are keywords to explicitly define the order of the t1 and t2 arrays (other than ni and ni2). By default, ft3d looks at the array parameter and if any parameter other than phase and phase 2 are arrayed, the macro aborts.

'fdf' indicates that the output of ft3d is to be an FDF (Flexible Data Format) file named data.fdf. This is the default if the parameter appmode is set to 'imaging'. Distributed processing can still be performed if number files is set appropriately. 3D FDF files can be viewed with the ImageBrowser.

'nofdf' indicates that the final output is the group of data1, data2, ... files, and that no FDF format file should be produced. This is the default if the parameter appmode is not set to 'imaging'.

plane type sets plane extraction following the complete 3D FT with the following keywords:

- 'xall' indicates that all three 2D plane types,  $f_1f_3$ ,  $f_2f_3$ , and  $f_1f_2$ , are to be automatically extracted at the end of the 3D Fourier transform.
- 'f1f3', 'f2f3', and 'f1f2' can be used to select any combination of plane types to be extracted.

Any of these options can be submitted more than once to the ft3d program, but the getplane program will display an error and abort if any one plane type is defined for extraction more than once.

```
Examples: ft3d
           ft3d('nocoef','f1f3','f2f3')
           ft3d Entered from UNIX
           (From UNIX) ft3d -e exp number -f -r <options>
```

The ft3d program can also be run directly from the UNIX environment on the host computer. An information file must be present before ft3d can execute successfully but it need contain only valid processing information for the  $t_3$  dimension and valid Fourier numbers for the  $t_1$  and  $t_2$  transforms. Valid weighting and phasing parameters for the  $f_1$  and  $f_2$  dimension do not need to be set while wftt3 executes. After several FIDs have been collected, you can determine acceptable  $f_3$  weighting and phasing parameters. After setting fn1 and fn2 to the desired values, the 3D processing information file can be created by typing set3dproc in the VnmrJ command line. At that point, the next invocation of ft3d by the macro wftt3 causes all  $(t_1,t_2)$  increment sets up to and including the current increment in  $t_3$  to be processed.

To start ft3d on a remote computer running as a data station for the system, log in as root and enter one of the following commands so that the master ft3d program can properly communicate with the computer:

• Enter /vnmr/acqbin/Infoprc &

With the Infopro or acqinfo\_svc program running, enter ft3d with the -h option and the necessary arguments. The ft3d program invoked with the -h option is considered to be the master program and is responsible for spawning additional remote ft3d processes.

Each remote computer must be able to access the 3D data directory as if it were stored on a local disk, must recognize the user name under which the master ft3d program is being run, and must also have permission to read from and write to that directory. If the 3D data directory contains four  $f_3$  transformed data files (data1-data4), the master ft3d program uses the first three remote computer systems listed in file hosts. 3D that respond.

If the multihost processing option is selected, the number of computers involved will be no more than the number of sets the  $f_3$  spectral data is partitioned into. This number is selected with the -m option (see below).

If you are unsure of whether to use Inforco or acqinfo\_svc on the remote computer, change directories to /vnmr/acqbin, enter lf, and check which program is present.

Note that if the host computer is rebooted, the background command (Infoprc or acqinfo\_svc) has to be entered again.

Arguments:

Note that entering ft3d with an ampersand (&) after the arguments makes the command execute in the background. As a result, the UNIX prompt reappears after the command is entered and further commands can be entered and executed while the ft3d command is processing.

• -e exp\_number is the experiment number where 3D processing is to occur. This argument is required. It must be written as a minus sign, the letter e, a space, and a valid experiment number from 1 to 9 (e.g., -e 3 sets experiment 3). The experiment must already exist.

The following two options should always be set for reliable operation:

- -f specifies that any existing 3D data sets in the experiment should be deleted. This option requires no additional value.
- -r calls for explicit data reduction after the 3D Fourier transform. Data reduction consists of retaining only the "real-real" part of the completely transformed 3D data set. The -r option is mandatory and is enforced within ft3d regardless of the user command line input.

options can be any of the following:

• -F header\_file indicates that an FDF (Flexible Data Format) output file should be produced, using the FDF header found in header\_file. The output file will be named data.fdf, and the data1, data2, ... files will not be produced.

- -h selects the multihost processing option. The /etc/hosts.3D file must exist and contain the names of the remote hosts, one host name per line. Each remote host must also have either the program Inforc or the program acqinfo\_svc running in the background (one of these programs is already running on any computer being used as a spectrometer host).
- -1 specifies that a log file be generated in the data subdirectory of the datadir3d directory.
- -m partitions the f<sub>3</sub> transformed spectral data over more than one data file. This partitioning is necessary if the distributed processing capability of ft3d is to be used in performing the remaining f<sub>1</sub> and f<sub>2</sub> transforms. The syntax -mnfiles is used to specify nfiles, the number of data files into which the 3D spectral data is to be divided (e.g., -m4 specifies 4 data files). Each such data file contains an f<sub>3</sub> subset of the f<sub>1</sub>f<sub>2</sub> spectral planes. If nfiles is not specified, ft3d reports an error and aborts. If nfiles is less than an internally calculated value (based on memsize and the maximum size for a single 2D transform), the number of data files is set to the internally calculated value; otherwise, nfiles determines the number of data files to be used. The maximum number of such files is currently defined to be 32. These 3D data files are labeled data1, deta2,..., datan.
- -o specifies an alternative output directory for the processed 3D data. The default directory is datadir3d within the current experiment. A full UNIX path must follow the -o option.
- -p specifies the time-domain dimensions to be processed. If -p is used, the processed dimensions can be specified as f3f2f1, f3f2, f2f3, f2f1, f1f2, f3, f2, and f1. The values f3f1 and f1f3 are not allowed because processing must be done sequentially in the order f3, then f2, and then f1. If the -p option is not invoked, ft3d defaults to f3f2f1, resulting in a completely transformed 3D data set.
- -s specifies processing of the  $f_3$  dimension of the 3D FID data concurrently with data acquisition. In practice, concurrent  $f_3$  processing is realized by setting wnt='wftt3' in the VnmrJ parameter set and starting the 3D acquisition by entering au. The macro wftt3 handles the call to ft3d at the appropriate times during data collection.
- -x specifies that plane extractions be performed at the end of 3D processing. The available planes are defined as f1f2, f1f3, and f2f3. If more than one plane extraction is desired, the planes are separated by a colon. For example, -x f1f2:f1f3:f2f3 would extract all three planes. The planes are placed in the extr subdirectory of datadir3d.

```
Examples: (From UNIX) ft3d -r -f -l -e 2 &
```

(From UNIX) ft3d -r -f -l -e 2 -x f1f2:f1f3:f2f3 &

See also: NMR Spectroscopy User Guide

Related: appmode Application mode (P)

dconi Interactive 2D data display (C)
fiddc3d 3D time-domain dc correction (P)

flcoef Coefficient to construct F1 interferogram (P)
flooef Coefficient to construct F2 interferogram (P)
getplane Extract planes from a 3D spectral data set (M)

killft3d Terminate any ft3d process started in an experiment (M,U)

make3dcoef Make 3D coefficients file from 2D coefficients (M)

ptspec3d Region-selective 3D processing (P)

set3dproc Set 3D processing (C)
specdc3d 3D spectral dc correction (P)

wftt3 Process f<sub>3</sub> dimension during 3D acquisition (M)

### full Set display limits for a full screen (C)

Description: Sets the horizontal control parameters (sc and wc) and the vertical control

parameters (sc2 and wc2) to produce a display (and subsequent plot) on the

entire screen (and page). For 2D data, space is left for the scales.

Related: center Set display limits for center of screen (C)

fullt Set display limits for full screen with room for traces (C)

left Set display limits for left half of screen (C)
right Set display limits for right half of screen (C)

Start of chart (P)

Start of chart in second direction (P)

wc Width of chart (P)

wc2 Width of chart in second direction (P)

### fullsq Display largest square 2D display (M)

Description: Adjusts sc, sc2, wc, and wc2 parameters to show the largest possible square

2D display.

Related: full Set display limits for a full screen (C)

fullt Set display limits for a full screen with room for traces (C)

Start of chart (P)

Start of chart in second direction (P)

WC Width of chart (P)

wc2 Width of chart in second direction (P)

### fullt Set display limits for a full screen with room for traces (C)

Description: Sets the horizontal control parameters (sc and wc) and the vertical control

parameters (sc2 and wc2) to produce a display (and subsequent plot) in the entire screen (and page) with room for traces (dconi). For 2D data, space is

left for the scales.

Related: center Set display limits for center of screen (C)

full Set display limits for a full screen (C)

left Set display limits for left half of screen (C)

right Set display limits for right half of screen (C)

# G

g2pul ecc Setup macro for eddy current compensation parameters (M) Submit experiment to acquisition and FT the result (M) σa Receiver gain (P) gain gap Find gap in the current spectrum (M) Set up unshifted Gaussian window function (M) gaussian gcal Gradient calibration constant (P) gcoil Current gradient coil (P) Gcosy Convert the parameter to a gradient COSY experiment (M) gdiff Diffusion gradient level (P) Gdqcosy Convert the parameter to a gradient DQCOSY experiment (M) Select a 1D experiment for processing (M) get1d get2d Select a 2D experiment for processing (M) Return dimensionality of experiment (M) getdim getfile Get information about directories and files (C) getlimit Get the limits of a variable in a tree (C) Get intensity and line frequency of line (C) getll Retrieve parameter from probe file (M) getparam getplane Extract planes from a 3D spectral data set (M) Get frequency limits of a specified region (C) getreg Get signal-to-noise estimate of a spectrum (M) getsn gettoken Utility macro to separate a string into tokens (M) Get text file from VnmrJ data file (C) gettxt Get the type of a variable (C) gettype Get value of parameter in a tree (C) getvalue gf Prepare parameters for FID/spectrum display in acqi (M) Gaussian function in directly detected dimension (P) gf Gaussian function in 1st indirectly detected dimension (P) gf1 gf2 Gaussian function in 2nd indirectly detected dimension (P) gflow Flow encoding gradient level (P) Gaussian shift const. in directly detected dimension (P) gfs gfs1 Gaussian shift const. in 1st indirectly detected dimension (P) qfs2 Gaussian shift const. in 2nd indirectly detected dimension (P) Ghmbc Convert the parameter to a gradient HMBC experiment (M) Set up a PFG HMQC pulse sequence (M) ghmqc Convert the parameter to a gradient HMQC experiment (M) Ghmqc Set up parameters for <sup>15</sup>N gHMQC experiment (M) gHMQC15 Set up parameters for <sup>15</sup>N gHMQC experiment using dec. 2 (M) gHMQC d2 Set up parameters for <sup>13</sup>C gHMQC experiment using dec. 2 (M) gHMQC\_d213 ghmqcps Set up a PFG HMQC phase-sensitive pulse sequence (M) Set up a PFG HSQC pulse sequence (M) ghsqc Ghsqc Convert the parameter to a gradient HSQC experiment (M) Set up parameters for <sup>15</sup>N gHSQC experiment (M) qHSQC15 gHSQC d2 Set up parameters for <sup>15</sup>N gHSQC experiment using dec. 2 (M)

G

gHSQC\_d213 Set up parameters for <sup>13</sup>C gHSQC experiment using dec. 2 (M)

Ghsqctoxy Convert parameters for gradient HSQCTOXY experiment (M)

Gilson Liquid Handler window (C)

gin Return current mouse position and button values (C)

globalautoAutomation directory name (P)glueCreate a pseudo-2D dataset (M)gmapshimStart gradient autoshimming (M)

gmapshim\_au Start acquisition with gradient shimming (M)

gmapspinEnable or disable spinning during gradient shimming (P)gmapsysRun gradient autoshimming, set parameters, map shims (M)gmapzGet parameters and files for gmapz pulse sequence (M)

gmap\_findtof Gradient shimming flag to first find tof (P)
gmap\_zlz4 Gradient shimming flag to first shim zl-z4 (P)

gmax Maximum gradient strength (P)

gmqcosy Set up PFG absolute-value MQF COSY parameter set (M)

gnoesy

Set up a PFG NOESY parameter set (M)
go

Submit experiment to acquisition (M)

go\_ Pulse sequence setup macro called by go, ga, and au (M)

gplan Start interactive image planning (C)

gradientdisable Disable PFG gradients (P)

gradientshaping Activate shaping on the gradient pulses (P)

gradstepsz Gradient step size (P)

**gradtype** Gradients for X, Y, and Z axes (P)

graphis Return the current graphics display status (C)

**grayctr** Gray level window adjustment (P)

Gray level slope (contrast) adjustment (P)

grid Draw a grid on a 2D display (M)

groupcopy Copy parameters of group from one tree to another (C)

gspoil Spoiler gradient level (P)
gsspat Slice-select gradient shape (P)

gtnnoesy Set up a PFG TNNOESY parameter set (M)

**Set up a PFG absolute-value ROESY parameter set (M)** 

gtotlimit Gradient total limit (P)
gtrim Trim gradient level (P)

gxmax, gymax, gzmax Maximum gradient strength for each axis (P)

gzlvl Pulsed field gradient strength (P)

gzsize

Number of z-axis shims used by gradient shimming (P)
gzwin

Spectral width percentage used for gradient shimming (P)

### g2pu1 ecc Setup macro for eddy current compensation parameters (M)

Applicability: Systems with Varian, Inc. Cold Probes

Description: Setup macro for pulse sequence used to determine the eddy current

compensation parameters.

### ga Submit experiment to acquisition and FT the result (M)

Syntax: ga<(<'nocheck'><,'next'><,'wait'>)>

Description:

Performs experiment described by the current acquisition parameters, checking parameters loc, spin, gain, wshim, load, and method to determine the necessity to perform various actions in addition to simple data acquisition. This may involve a single FID or multiple FIDs, as in the case of arrays or 2D experiments. ga causes the data to be automatically weighted and Fourier transformed (wft) at the end of each FID data acquisition.

Before starting the experiment, ga executes two user-created macros if they exist. The first is usergo, a macro that allows the user to set up general conditions for the experiment. The second is a macro whose name is formed by go\_followed by the name of the pulse sequence (from seqfil) to be used (e.g., go\_s2pul, go\_dept). The second macro allows a user to set up experiment conditions suited to a particular sequence.

Arguments:

'nocheck' is a keyword to override checking if there is insufficient free disk space for the complete 1D or 2D FID data set to be acquired.

'next' is a keyword to put the experiment started with ga('next') at the head of the queue of experiments to be submitted to acquisition.

'wait' is a keyword to stop submission of experiments to acquisition until wexp processing of the experiment, started with ga('wait'), is finished.

See also: NMR Spectroscopy User Guide

Related:

au Submit experiment to acquisition and process data (M) Change Submit a change sample experiment to acquisition (M)

gain Receiver gain (P)

go Submit experiment to acquisition (M)

go\_ Pulse sequence setup macro called by go, ga, and au (M)

Load status of displayed shims (P)
Location of sample in tray (P)

lock Submit an Autolock experiment to acquisition (C)

method Autoshim method (P)

Submit change sample, Autoshim experiment to acquisition (M)

seqfil Pulse sequence name (P)

shimSubmit an Autoshim experiment to acquisition (C)spinSubmit a spin setup experiment to acquisition (C)

spin Sample spin rate (P)

su Submit a setup experiment to acquisition (M)

usergo Experiment setup macro called by go, ga, and au (M)

wft Weight and Fourier transform 1D data (C)
wshim Conditions when shimming is performed (P)

### gain Receiver gain (P)

Description:

Sets receiver gain or, by setting gain='n', enables Autogain for automatic adjustment of gain. Low gain in multiline, high-dynamic-range samples can cause a number of problems, including intermodulation distortions and extra lines in the spectrum. Too high a gain, on the other hand, can cause receiver overload and consequent baseline distortions. Autogain capability allows the observe channel to be set optimally for detecting and digitizing NMR signals from a wide variety of samples.

Autogain adjusts the observe channel gain such that the NMR signal takes about 50 percent of the maximum range of the ADC. This setting allows a comfortable leeway for variations in signal. The program begins acquisition in the normal manner but the first transient (after any requested steady state transients) is

examined for signal level. If the intensity is too low or too high, the gain is changed and the process is repeated until the intensity is within the proper range, and then normal acquisition commences. The final gain value used for the experiment is stored and when the experiment is finished, setting gain='y' results in the value being displayed in the dgs parameter group.

If the gain is reduced by the Autogain procedure such that the noise does not trigger the least significant 1 or 2 bits in the ADC and the signal still overloads either the receiver or ADC, the system stops and displays a message indicating Autogain failure.

Values: 0 to 60, in steps of 2 dB (60 represents highest possible receiver gain and 0

lowest). On 500-750-MHz systems, low-band gain is limited from 18 to 60.

'n' enables Autogain, in which the gain is automatically adjusted at the start of acquisition for an optimum value. After the acquisition is finished, setting gain='y' then allows the value of gain to be read. gain='n' may not be

used for arrayed experiments.

See also: NMR Spectroscopy User Guide

Related: dgs Display group of special/automation parameters (M)

### gap Find gap in the current spectrum (M)

Syntax: gap(gap, height): found, position, width

Description: Looks for a gap between the lines of the currently displayed spectrum. It can be

used to automatically place inserts, parameter printouts, trace labels, etc. The

search starts on the left side (low-field end) of the spectrum.

Arguments: gap is the width of the desired gap.

height is the starting height (same as the lower limit for the insert).

found is a return value that is set to 1 if the search is successful, or set to 0 if

unsuccessful.

position is a return value that is set to the distance from the left edge of the chart (not the plot) to the left end of the gap (3 mm from the nearest peak to the left, positioning with "left gravity") if the search is successful, or set to the position (no spacing to the nearest line) of the largest gap found if unsuccessful.

width is a return value set to the total width of the first gap if the search is

successful, or set to the width of largest gap found if unsuccessful.

Examples: gap(120,80);\$1,\$2,\$3

See also: *User Programming* 

### gaussian Set up unshifted Gaussian window function (M)

Syntax: gaussian<(<t1\_inc><,t2\_inc>)>

Description: Sets up an unshifted Gaussian window function in 1, 2, or 3 dimensions. The

macro checks whether the data is 1D, 2D, and 3D.

Arguments: t1 inc is the number of t1 increments. The default is ni.

t2 inc is the number of t2 increments. The default is ni2.

See also: NMR Spectroscopy User Guide

Related: ni Number of increments in 1st indirectly detected dimension (P)

ni2Number of increments in 2nd indirectly detected dimension (P)pi3ssbsqSet up pi/3 shifted sinebell-squared window function (M)pi4ssbsqSet up pi/4 shifted sinebell-squared window function (M)

sqcosine Set up unshifted cosine-squared window function (M)
sqsinebell Set up unshifted sinebell-squared window function (M)

### gcal Gradient calibration constant (P)

Applicability: Systems with the pulsed field gradient or the imaging module.

Description: Stores the proportionality constant between the parameter values (DAC units)

controlling the desired gradient and the intensity of the gradient expressed in gauss/cm. The gradients generated in the magnet require calibration of the gain on the gradient compensation board so that coordinate data, slice positions, and the field of view can be set up accurately. gcal should be located in each user's

vnmrsys/global file.

Values: Number that is probe dependent, in gauss/cm-DAC unit. On the Performa I PFG

module, 0.00028 to 0.00055 gauss/cm-DAC unit is nominal; On the Performa

II, 0.0014 to 0.0025 gauss/cm-DAC unit is nominal.

See also: VnmrJ Imaging NMR

Related setgcal Set gradient calibration constant (M)

# gcoil Current gradient coil (P)

Description:

Reserved parameter that specifies which physical gradient set is currently installed. This allows convenient updating of important gradient characteristics when one gradient set is interchanged for another. When set, gcoil reads the gradient table file of the same name in /vnmr/imaging/gradtables and sets the gradient calibration parameters.

gcoil is local to each individual experiment. It is normally set the same as sysgcoil for acquiring new data, but can be set to other gradient names when working with saved data or data from another instrument. Each possible gradient name should have an associated file of that name located in the directory /vnmr/imaging/gradtables. Look at any file in this directory for an example of the proper gradtable format, or use the macro creategtable to make new gradtables entries.

If the parameter gooil does not exist in a parameter set and a user wants to create it, you must set the protection bit that causes the macro \_gooil to be executed when the value for gooil is changed. There are two ways to create gooil:

- Use the macro updtgcoil, which will create the gcoil parameter if it does not exist and set the correct protection bits.
- Enter the following commands:

```
create('gcoil','string')
setprotect('gcoil','set',9)
```

gradient calibration parameter gmax is updated with the values listed in the table on the right each time a parameter set is retrieved, or when an experiment is joined. In the rare case that a gradtables file is modified, but the value of gooil is not changed, manually

Variable Name	Value
boresize	22.50 cm
gmax	5.00 gauss/cm
trise	0.000500 sec

force an update of the calibration parameters. Updating may be accomplished either by setting gcoil to itself, for example, gcoil=gcoil, or by using the macro gcoil.

Be aware that if an old dataset is returned and processed, gradient parameters associated with that dataset will replace any new qcoil parameters.

The table is a gradient table (gradient coil name: asg33) for a horizontal imaging system with all three axes set to the same maximum gradient strength.

On the right is a gradient table (gradient coil name: tc203) for a three-axis gradient set with unequal maximum gradient strength.

Variable Name	Value
boresize	5.10 cm
trise	0.000200 sec
gxmax	29.00 gauss/cm
gymax	27.00 gauss/cm
gzmax	70.00 gauss/cm

See also: User Programming

Related: gmax Maximum gradient strength (P)

setgcoil Assign sysgcoil configuration parameter (M)

sysgcoil System gradient coil (P)
updtgcoil Update gradient coil (M)

Gcosy Convert the parameter to a gradient COSY experiment (M)

Applicability: Systems with the pulsed field gradient or the imaging module.

Description: Converts a 1D standard two-pulse sequence parameter set into a set ready to run

a PFG (pulsed field gradient) absolute-value COSY experiment.

See also: NMR Spectroscopy User Guide

gdiff Diffusion gradient level (P)

Description: Predefined parameter available for use in setting a diffusion gradient level, often

paired with the timing parameters tdiff or tdelta.

Gdqcosy Convert the parameter to a gradient DQCOSY experiment (M)

Description: Convert the parameter to a gradient Dqcosy experiment

get1d Select a 1D experiment for processing (M)

Syntax: get1d<(experiment)>

Description: In nonautomation mode, the macros hoosy, hoapt, capt, hodept, and

cdept all acquire two or more data sets in the experiment in which the macro was executed. These data sets are stored, complete with Fourier transformed data. The data sets are also stored directly in the experiment. The getld macro

is used to select which data set should be active for processing in that

experiment. After getld is executed, data can be stored in the conventional way with the svf command (e.g., when hcosy completes, getld can be used

to process the 1D data set).

Arguments: experiment is the 1D data set to be used for processing. The default is the

'H1' experiment.

Examples: get1d

get1d('apt')

See also: NMR Spectroscopy User Guide

Related: capt Automated carbon and APT acquisition (M)

cdept Automated carbon and DEPT acquisition (M) get2d Select a 2D experiment for processing (M)

hcapt Automated proton, carbon, and APT acquisition (M)
hcdept Automated proton, carbon, and DEPT acquisition (M)

hcosy Automated proton and COSY acquisition (M)

Save FIDs in current experiment (C)

## get2d Select a 2D experiment for processing (M)

Syntax: get2d<(experiment)>

Description: In nonautomation mode, the macros hcosy, hcapt, capt, hcdept, and

cdept all acquire two or more data sets in the experiment in which the macro was executed. These data sets are stored complete with Fourier transformed data. The data sets are also stored directly in the experiment. The get2d macro

is used to select which data set should be active for processing in that

experiment. After entering get2d, data may be stored in the conventional way with the <a href="mailto:svf">svf</a> command. For example, following completion of <a href="mailto:hcosy">hcosy</a>, get2d

can be used to process the 2D data set.

Arguments: experiment is the 2D data set that should be used for processing. The default

is the 'relayh' experiment.

Examples: get2d('hetcor')

See also: NMR Spectroscopy User Guide

Related: get1d Select a 1D experiment for processing (M)

Save FIDs in current experiment (C)

#### getdim Return dimensionality of experiment (M)

Syntax: getdim:dimensions

Description: Used in other macros to determine the number of dimensions of the current data

set. Many macros make decisions based on whether a data set is

multidimensional or 1D. getdim makes it easier to access this information.

Arguments: dimensions is a return variable giving the number of dimensions of the data.

If ni3 is 2 or greater, dimensions is set to 4; if ni2 is 2 or greater,

dimensions is set to 3; if ni is 2 or greater, dimensions is set to 2; and if

ni is less than 2 or undefined, dimensions is 1.

Examples: getdim:r1

See also: NMR Spectroscopy User Guide

Related: ni Number of increments in 1st indirectly detected dimension (P)

Number of increments in 2nd indirectly detected dimension (P)

Number of increments in 3rd indirectly detected dimension (P)

#### getfile Get information about directories and files (C)

Syntax: (1) getfile(directory): \$number files

(2) getfile(directory, file index):\$file,\$extension

Description: Returns information about the number of files in a directory or about a particular

file in a directory.

Arguments: directory is the name of the directory for which information is desired.

number\_files is the number of files in the directory, with dot files

(e.g., .login) ignored.

file index is the number of file for which information is desired (the order

is UNIX-dependent).

file is the name of the file, excluding any extension, identified by the index

(see examples below).

extension is the extension of the file name identified by the file\_index. For example, if file\_index points to the file named s2pul.fid, getfile returns the string s2pul to \$file and the string fid to \$extension. If the file name pointed to has no extension (e.g., dummy), no value is returned to \$extension. If the file name has more than one extension, only the last extension is returned to \$extension (e.g., the file fid.tmp.par returns fid.tmp to \$file and par to \$extension).

Complete paths (full file names) can be reconstructed like this:

```
getfile('dir',i):$filename,$ext
if ($ext='') then $path='dir'+'/'+$filename
else $path='dir'+'/'+$filename+'.'+$ext
endif
```

Paths for the rt command can be reconstructed like this:

\$path='dir'+'/'+\$filename.

Examples: getfile('dir'):\$entries \$temp = 0

while (\$temp < \$entries)
 \$temp = \$temp + 1
 getfile('dir',\$temp):\$filename,\$ext</pre>

endwhile

See also: User Programming

# getlimit get the limits of a variable in a tree (C)

Syntax: getlimit(name[,tree]):\$max,\$min,\$step,\$index Description: getlimit displays or returns the limits of a variable in a tree.

The returned values are the max value, min. value, step size, and index. The fourth argument will return a 0 if the parameter is not using an indexed table lookup for the maximum, minimum, and step size. If the parameter is using the table lookup mechanism, the fourth argument will be set to the index for that

table.

The variable trees are current (the default), global, processed, or

systemglobal.

Arguments: name — the name of the variable

tree — the variable tree: current (the default), global, processed, or

systemglobal.

Examples: getlimit('np'):\$max,\$min,\$step,\$index

sets \$max to 128000, \$min to 32, \$step to 2 and \$index to 0 getlimit('lockfreg','systemglobal'):\$max

sets \$max to 160

getlimit('dpwr'): \$max, \$min, \$step, \$index sets \$max to 49, \$min to 0 \$step to 1 and \$index to 9

Related: setlimit Set limits of a parameter in a tree (C)

setprotect Set protection mode of a parameter (C)

# get11 Get intensity and line frequency of line (C)

Syntax: getll(line number)<:height,frequency>

Description: Finds the height and frequency of line from a line listing. It assumes a previous

line list using dll.

Arguments: line number is the number of the line in the line list.

height is the intensity of the specified line.

frequency is the line frequency with units defined by the parameter axis.

See also: User Programming

Related: axis Axis label for displays and plots (P)

dll Display listed line frequencies and intensities (C)

fp Find peak heights (C)

nll Find line frequencies and intensities (C)

# getparam Retrieve parameter from probe file (M)

Syntax: getparam(param<,nucleus>):\$value

Description: Retrieves the value of a parameter from the current probe file. The name of the

probe file is referenced from the parameter probe.

Arguments: param is the name of the parameter to be retrieved.

nucleus is the nucleus to be retrieved from the probe file. The default is the

current value of the parameter tn

value is a return variable with the value of the retrieved parameter.

Examples: getparam('tpwr'):tpwr

getparam('dmf','H1'):\$dmf

See also: NMR Spectroscopy User Guide

Related: addnucleus Add new nucleus to existing probe file (M)

addparams Add parameter to current probe file (M)

addprobe Create new probe directory and probe file (M)

probe Probe type (P)

setparams Write parameter to current probe file (M)
tn Nucleus for the observe transmitter (P)

updateprobe Update probe file (M)

#### getplane Extract planes from a 3D spectral data set (M)

Syntax: getplane<(<data\_dir><,plane\_dir><,plane\_type>)>

Description: Executes the program getplane in the VnmrJ system bin directory

(\$vnmrsystem/bin). getplane checks whether there is sufficient file space on the disk partition to accommodate the extracted planes. If space is insufficient, getplane writes an error to the VnmrJ text window and aborts. getplane does not delete the output plane directory if it is run multiple times

to individually extract different plane types.

Arguments: data dir specifies the directory (without the /data subdirectory)

containing the input 3D spectral data. The first non-keyword argument to

getplane is always taken to be data dir.

plane\_dir specifies the directory (without the /extr subdirectory) in which the extracted planes are to be stored. The second non-keyword argument to getplane is always taken to be plane\_dir. If plane\_dir is not specified, data\_dir also specifies the output plane directory. If both data\_dir and plane\_dir are not specified, the input data directory and the output plane directory are set to curexp/datadir3d. The parameter

plane is always set equal to the output plane directory.

plane type can be any of the following keywords:

• 'xall' is a keyword to extract all three 2D plane types: f1f3, f2f3, f1f2.

- 'f1f3', 'f2f3', 'f1f2' are keywords to extract their respective 2D planes.
- Any of these keywords can be submitted more than once to the getplane macro, but the getplane program displays an error and aborts if any one plane type is defined for extraction more than once.

Examples: getplane

getplane('data3d.inp,'data3d.planes','f1f3','f2f3')

See also: NMR Spectroscopy User Guide

Related: dplane Display a 3D plane (M)

dproj Display a 3D plane projection (M)
dsplanes Display a series of 3D planes (M)
ft3d Perform a 3D Fourier transform (M)
nextpl Display the next 3D plane (M)

path3d Path to currently displayed 2D planes from a 3D data set (P)

plane Currently displayed 3D plane type (P)
plplanes Plot a series of 3D planes (M)
prevpl Display the previous 3D plane (M)

# getreg Get frequency limits of a specified region (C)

Syntax: getreg(region number) <: minimum, maximum>

Description: Returns the frequency limits of a region. The spectrum should have been

previously divided into regions with the region command.

Arguments: region number specifies the number of the region.

minimum, maximum are return values set to the frequency limits, in Hz, of the

specified region.

Examples: getreg(1):\$a,\$b

getreg(\$4):cr,\$lo
getreg(R1-1):r2,r3

See also: User Programming

Related: Cz Clear integral reset points (C)

ds Display a spectrum (C)

numreg Return the number of regions in a spectrum (C)

region Divide spectrum into regions (C)

Add integral reset point at cursor position (C)

## getsn Get signal-to-noise estimate of a spectrum (M)

Syntax: getsn:current\_sn,predicted\_sn

Description: Estimates spectrum signal-to-noise using the following algorithm:

- Measures four adjacent 5-percent portions at the left edge of the spectrum, finding the root-mean-square noise, and taking the smallest of the four values. By measuring four different values and finding root-mean-square noise instead of peak noise, the result should be reliable even if several signals are present in the selected regions.
- Next, estimates the signal level using the vertical scale adjustment macros: vsadjh for proton, vsadjc for carbon, and vsadj for other nuclei. For carbon spectra, this algorithm ignores solvent lines and TMS. For proton spectra, in addition to ignoring the largest line in the spectrum, if the tallest line is greater than three times the height of the second tallest line, the

second highest line is be used instead. For other nuclei, getsn uses the tallest line in the spectrum.

• Finally, estimates the signal-to-noise at the end of the experiment by a simple extrapolation (multiplying by the square root of nt/ct).

Arguments: current\_sn is a return value set to the current signal-to-noise level.

predicted\_sn is a return value set to the predicted signal-to-noise level at

the end of the experiment.

See also: NMR Spectroscopy User Guide

Related: ct Completed transients (P)

nt Number of transients (P)
testsn Test signal-to-noise ratio (M)
vsadj Adjust vertical scale (M)

vsadjc Adjust vertical scale for carbon spectra (M)
vsadjh Adjust vertical scale for proton spectra (M)

## gettoken Utility macro to separate a string into tokens (M)

Syntax: gettoken(input\_string<,delimiter>):output\_string,

next\_location

Description: Gets the first occurrence of a substring in input\_string which is delimited

by delimiter, or by the default delimiter '\$'. The substring is returned in output\_string. The next location in the string after the second delimiter is returned as a real in next\_location. If there are not both one occurrence of each of the beginning delimiter and the second delimiter - in other words, if the delimiters are not paired - an empty string is returned in output\_string, and -1 is returned in next\_location. If the delimited substring is the last substring in input\_string, then the substring is returned as expected, but

next location returns -1.

Arguments: input\_string

The string to be tokenized delimiter is the delimiter for the tokens (default is \$)

Examples: gettoken(\$mydirname):\$mytoken, \$next\_location

gettoken(\$mydirname,'%'):\$mytoken, \$next\_location

Related: regpartest Tests whether required parameters are set (M)

## gettxt Get text file from VnmrJ data file (C)

Syntax: gettxt(file)

Description: Copies text from a data file to the current experiment.

Arguments: file is the name of a VnmrJ data file saved from an experiment (i.e., a

directory with a .fid or .par suffix). Do not include the file name suffix.

Examples: gettxt('/vnmr/fidlib/fid1d')

See also: NMR Spectroscopy User Guide

Related: puttxt Put text file into another file (C)

## gettype Get the type of a variable (C)

Syntax: gettype(name[, tree])<:index, name>
Description: Displays or returns the type of an existing variable.

Arguments: A "string" variable can return type 'string' or `flag'. A "real" variable

can return type 'real', 'delay', 'frequency', 'pulse', or

'integer'. gettype returns one or two values to a macro. The first value is an integer corresponding to the parameter type. The second value is the name of the parameter type. name can be used in commands such as settype and

create.

An optional tree argument can be given. Variables are 'current',

'global', 'processed', and 'systemglobal'.

The default is to search for the parameter in the 'current', 'global', and

'systemglobal' trees, in that order.

Examples: gettype('dmm'):\$int,\$name sets \$int to 4 and \$name to 'flag'.

See also: gettype('pw'):\$int,\$name sets \$int to 6 and \$name to 'pulse'.

## getvalue Get value of parameter in a tree (C)

Syntax: getvalue(parameter<,index><,tree>)

Description: Gets the value of any parameter in a tree. The value of most parameters can be

accessed simply by using their name in an expression. For example, sw? or r1=np accesses the value of sw and np, respectively. However, parameters in the processed tree cannot be accessed that way; getvalue can be used to get

the value of a parameter in the processed tree.

Arguments: parameter is the name of an existing parameter.

index is the number of a single element in an arrayed parameter. Default is 1.
tree is one of the keywords 'global', 'current', 'processed', or
'systemglobal'. The default is 'processed'. Refer to the create

command for more information on the types of parameter trees.

Examples: getvalue('arraydim')

See also: User Programming

Related: create Create new parameter in a parameter tree (C)

displayDisplay parameters and their attributes (C)setgroupSet group of a parameter in a tree (C)setlimitSet limits of a parameter in a tree (C)setprotectSet protection mode of a parameter (C)settypeChange type of a parameter (C)setvalueSet value of any parameter in a tree (C)

## gf Prepare parameters for FID/spectrum display in acqi (M)

Description: Provided as a model for preparing parameters for the FID and spectrum display

in acqi. The unmodified version of this macro turns off phase cycling, autoshimming, autolocking, spin control, temperature control, sample changer control, and autogain. It also selects the current pulse sequence and parameter set by issuing the command go('acqi') and the command acqi('par'). The automation parameters cp, wshim, alock, spin, temp, loc, and gain are then reset to their original values. Users can customize gf by copying it into their private maclib directory and editing that version to suit their needs.

See also: NMR Spectroscopy User Guide

Related: acqi Interactive acquisition display process (C)

alock Automatic lock status (P)

Cycle phase (P)

dmgf Absolute-value display of FID data and spectrum in acqi (P)

gain Receiver gain (P)

go Submit an experiment to acquisition (C)

Location of sample in tray (P)

spin Sample spin rate (P)
temp Sample temperature (P)

wshim Conditions when shimming performed (P)

gf Gaussian function in directly detected dimension (P)

Description: Defines a Gaussian time constant of the form  $\exp(-(t/gf) 2)$  along the

directly detected dimension. This dimension is referred to as the f<sub>2</sub> dimension

in 2D data sets, the  $f_3$  dimension in 3D data sets, etc.

Values: Number, in seconds. Typical value is gf='n'.

See also: NMR Spectroscopy User Guide

Related: gf1 Gaussian function in 1st indirectly detected dimension (P)

gf2 Gaussian function in 2nd indirectly detected dimension (P)
gfs Gaussian shift constant in directly detected dimension (P)

gf1 Gaussian function in 1st indirectly detected dimension (P)

Description: Defines a Gaussian time constant of the form exp(-(t/gf1)2) along the

first indirectly detected dimension. This dimension is referred to as the  $f_1$  dimension of a multidimensional data set. gf1 works analogously to the parameter gf. The "conventional" parameters, such as lb and gf, operate on the detected FIDs, while this "2D" parameter is used during processing of the

interferograms.

Values: Number, in seconds.

See also: NMR Spectroscopy User Guide

Related: gf Gaussian function in directly detected dimension (P)

gf2 Gaussian function in 2nd indirectly detected dimension (P)

Description: Defines a Gaussian time constant of the form exp(-(t/gf2)2) along the

second indirectly detected dimension. This dimension is referred to as the  $f_2$  dimension of a multidimensional data set. gf2 works analogously to the parameter gf. The wti program can be used to set gf2 on the 2D

interferogram data.

Values: Number, in seconds.

See also: NMR Spectroscopy User Guide

Related: gf Gaussian function in directly detected dimension (P)

wti Interactive weighting (C)

gflow Flow encoding gradient level (P)

Description: Predefined parameter available for use in setting a flow encoding gradient level,

often paired with the timing parameter tflow.

See also: VnmrJ Imaging NMR

gfs Gaussian shift const. in directly detected dimension (P)

Description: Working in combination with the gf parameter, gfs allows shifting the center

of the Gaussian function exp(-((t-gfs)/gf)2) along the directly

detected dimension. This dimension is referred to as the  $f_2$  dimension in 2D data

sets, the  $f_3$  dimension in 3D data sets, etc. Typical value is gfs='n'.

See also: NMR Spectroscopy User Guide

Related: gf Gaussian function in directly detected dimension (P)

gfs1 Gaussian shift const. in 1st indirectly detected dimension (P)
gfs2 Gaussian shift const. in 2nd indirectly detected dimension (P)

## gfs1 Gaussian shift const. in 1st indirectly detected dimension (P)

Description: Working in combination with the gf1 parameter, gfs1 allows shifting the

center of the Gaussian function  $\exp(-((t-gfs1)/gf1)2)$  along the first indirectly detected dimension. This dimension is referred to as the  $f_1$  dimension in multidimensional data sets. gfs1 works analogously to the parameter gfs. The "conventional" parameters (i.e., lb, gf, etc.) operate on the detected FIDs, while this "2D" parameter is used during processing of the interferograms.

See also: NMR Spectroscopy User Guide

Related: qf Gaussian function in directly detected dimension (P)

gf1 Gaussian function in 1st indirectly detected dimension (P)
gfs Gaussian shift const. in directly detected dimension (P)

## gfs2 Gaussian shift const. in 2nd indirectly detected dimension (P)

Description: Working in combination with the gf2 parameter, gfs2 allows shifting the

center of the Gaussian function  $\exp(-((t-gfs2)/gf2)2)$  along the second indirectly detected dimension. This dimension is referred to as the  $f_2$  dimension in multidimensional data sets. gfs2 works analogously to the parameter gfs. The wti program can be used to set gfs2 on the 2D

interferogram data.

See also: NMR Spectroscopy User Guide

Related: gf Gaussian function in directly detected dimension (P)

gf2 Gaussian function in 2nd indirectly detected dimension (P)
gfs Gaussian shift const. in directly detected dimension (P)

wti Interactive weighting (C)

### Ghmbc Convert the parameter to a gradient HMBC experiment (M)

Applicability: Systems with a pulsed field gradient module.

Description: Prepares an experiment for a PFG (pulsed field gradient) HMQC.

Arguments: NMR Spectroscopy User Guide

## ghmqc Set up a PFG HMQC pulse sequence (M)

Applicability: Systems with a pulsed field gradient module.

Description: Prepares an experiment for a PFG (pulsed field gradient) HMQC using the

sequence GHMQC. The sequence sets three gradients, all separately.

Arguments: NMR Spectroscopy User Guide

## Ghmqc Convert the parameter to a gradient HMQC experiment (M)

Description: Convert the parameter to a gradient HMQC experiment

gHMQC15 Set up parameters for <sup>15</sup>N gHMQC experiment (M)

Description: Converts the current parameter set to a gHMQC experiment for <sup>15</sup>N.

gHMQC d2 Set up parameters for <sup>15</sup>N gHMQC experiment using dec. 2 (M)

Description: Converts the current parameter set to a gHMQC experiment for <sup>15</sup>N with

decoupler 2 as <sup>15</sup>N.

gHMQC d213 Set up parameters for <sup>13</sup>C gHMQC experiment using dec. 2 (M)

Description: Converts the current parameter set to a gHMQC experiment for <sup>13</sup>C with

decoupler 2 as <sup>13</sup>C.

ghmqcps Set up a PFG HMQC phase-sensitive pulse sequence (M)

Applicability: Systems with a pulsed field gradient module.

Description: Prepares an experiment for a PFG (pulsed field gradient) HMQC, phase-

sensitive version.

See also: NMR Spectroscopy User Guide

ghsqc Set up a PFG HSQC pulse sequence (M)

Applicability: Systems with a pulsed field gradient module.

Syntax: ghsqc<(nucleus) >

Description: Converts a 1D standard two-pulse sequence parameter set into a parameter set

ready to run a PFG (pulsed field gradient) HSQC experiment, either absolute

value or phase sensitive.

Arguments: nucleus is 13C or 15N. The default is 13C.

See also: NMR Spectroscopy User Guide

Ghsqc Convert the parameter to a gradient HSQC experiment (M)

Description: Convert the parameter to a gradient HSQC experiment.

gHSQC15 Set up parameters for <sup>15</sup>N gHSQC experiment (M)

Description: Converts the current parameter set to a gHSQC experiment for <sup>15</sup>N.

gHSQC d2 Set up parameters for <sup>15</sup>N gHSQC experiment using dec. 2 (M)

Description: Converts the current parameter set to a gHSQC experiment for <sup>15</sup>N with

decoupler 2 as <sup>15</sup>N.

gHSQC d213 Set up parameters for <sup>13</sup>C gHSQC experiment using dec. 2 (M)

Description: Converts the current parameter set to a gHSQC experiment for <sup>13</sup>C with

decoupler 2 as <sup>13</sup>C.

Ghsqctoxy Convert parameters for gradient HSQCTOXY experiment (M)

Description: Convert the parameter to a gradient HSQCTOXY experiment

G

gilson Open the Gilson Liquid Handler window (C)

Syntax: gilson

Description: Opens the Gilson Liquid Handler window, which enables setup, configuration,

and operation of the VAST automatic sampler changer accessory.

See also: NMR Spectroscopy User Guide

gin Return current mouse position and button values (C)

Applicability: All

 $Syntax: gin<(Bn_<pre>press><release>)>:$x,$y,$b1,$b2,$b3$ 

Description: The gin command reports the pointer position in relationship to the graphics

window and is often used with the move and draw commands. The variables x and y are the x and y positions hold the pointer in millimeters. The variables b1, b2, and b3 hold the values for the state of the left, middle, and right

mouse buttons.

Values: x = x is the value in the x direction, in millimeters, of the pointer. The range of x

is 0 at the left edge of the chart and wcmax at the right edge. A value of -1 is returned if the pointer position is outside the graphics window along the x axis.

y is the position of the pointer along the y axis. The range of y is -20 at the bottom of the chart to wc2max at the top. A value of 10000 is returned if the

pointer position is outside the graphics window along the y axis.

\$b1 is the state of left button; returns the value 0 if released and 1 if pressed. \$b2 is the of middle button; returns the value 0 if released and 1 if pressed. \$b3 is the of right button; returns the value 0 if released and 1 if pressed.

Arguments: no argument, returns current mouse positions and button values.

Bn press, n=a, 1, 2, or 3. Wait for mouse button (any, 1, 2, or 3) or any key

to be pressed.

Bn release, n=a, 1, 2, or 3. Wait for mouse button (any, 1, 2, or 3) to be

released or any key to be pressed.

Examples: gin('B3 press'):\$x,\$y,\$b1,\$b2,\$b3

wait until button 3or any key is pressed

gin('Ba\_press'):\$x,\$y,\$b1,\$b2,\$b3 wait until any button or any key is pressed

gin('B1\_release'):\$x,\$y,\$b1,\$b2,\$b3
wait until button 1 is released or any key pressed
gin('B2\_release'):\$x,\$y,\$b1,\$b2,\$b3
wait until button 2 is released or any key pressed

See also: User Programming

Related: box Draw a box on a plotter or graphics display (C)

draw Draw line from current location to another location (C)

move Move to an absolute location to start a line (C)

globalauto Automation directory name (P)

Applicability: *VnmrJ Walkup* and systems with automation such as sample handling.

Description: A global parameter that specifies the name of a directory in which the daily

automation directories or study directories are saved. This parameter is created

and used by the walkup macro and the VnmrJ Walkup interface.

See also: NMR Spectroscopy User Guide; VnmrJ Walkup

Related: cqinit Initialize liquids study queue (M)

walkup Walkup automation (M)

glue Create a pseudo-2D dataset (M)

Applicability: Systems with the LC-NMR accessory.

Syntax: glue<(num\_scans)>

Description: Steps through the series of FIDs, putting them into exp5 one by one as an array,

and then jumps to exp5 and changes the parameters arraydim, ni, and fn1, so that the data appear to the user to be a 2D experiment, which can then be processed and displayed with standard 2D commands (wft2d, dconi, etc.). The parameter savefile should exist and should contain the base file name

to which a series of FIDs have been saved as savefile.001,

savefile.002, etc.

Arguments: num scans is the number of FIDs copied into the exp5 array. Typically,

num\_scans is used if the experiment was aborted prematurely, so that the

complete num scans worth of FIDs were not actually acquired.

See also: NMR Spectroscopy User Guide

Related: savefile Base file name for saving FIDs or data sets (P)

gmapshim Start gradient autoshimming (M)

Applicability: Systems with gradient shimming installed.

Syntax: gmapshim<('files'|'mapname'|'quit')>

Description: Starts gradient autoshimming if no arguments are used. It can also retrieve a

shimmap file or quit gradient autoshimming. When the gmapshim macro is

done, it automatically exits, and the previous data set is retrieved.

Arguments: 'files' is a keyword to enter the gradient autoshimming files menu.

'mapname' is a keyword to display the current mapname.

'quit' is a keyword to exit from gradient autoshimming and retrieve the

previous data set.

See also: NMR Spectroscopy User Guide

Related: gmapsys Run gradient autoshimming, set parameters, map shims (M)

 $\begin{tabular}{ll} $\tt gmapz$ & Get parameters and files for $\tt gmapz$ pulse sequence (M) \\ \end{tabular}$ 

gmapshim au Start acquisition with gradient shimming (M)

Applicability: Systems with gradient shimming installed.

Description: If wshim is not set to 'n', gmapshim au checks the probe file for a lock

gradient map name. If the name exists, gmapshim\_au executes gmapshim('qlideau') to start gradient shimming followed by

acquisition. If the map name does not exist, gmapshim au starts acquisition

by running au ('wait').

gmapspin Enable or disable spinning during gradient shimming (P)

Description: Specifies whether or not sample spinning during gradient shimming is enabled.

If spinning is enabled during gradient shimming, the pulses and delays must also

be synchronized with the rotor period.

Values: 'n' disable spinning during gradient shimming.

'y' enable spinning during gradient shimming.

Related: gmapz Get parameters and files for gmapz pulse sequence (M)

gmapsys Run gradient autoshimming, set parameters, map shims (M)
gzsize Number of z-axis shims used by gradient shimming (P)

spin Sample spin rate (P)

## gmapsys Run gradient autoshimming, set parameters, map shims (M)

Applicability: Systems with gradient shimming installed.

Syntax: (1) gmapsys< (option) >

(2) gmapsys('shimmap'<, shimmap option>)

Description: Enters the Gradient Shimming Setup panel for setting parameters, mapping the

shims, and performing autoshimming. This is the only entry point to the

gradient shimming Setup panel.

If the gmapz pulse sequence is not loaded, retrieve parameters from the last shimmap used (or current mapname) or from gmapz.par if no shimmap

exists.

Arguments: option is one of the following keywords:

 'addpar' adds gradient shimming parameters to the current parameter set.

- 'findgzlvl' runs an experiment to calibrate gzlvl, gzwin, and tof to optimize the spectral window.
- 'findgzwin' runs an experiment to calibrate gzwin and tof to optimize the spectral window.
- 'findtof' runs an experiment to center tof to optimize the spectral window.
- 'rec' displays the record of shim adjustments from the previous gradient shimming run.
- 'shim' start autoshimming (same as Gradient Autoshim on Z button).
- 'vi' edits the file gshim.list, which is used for editing shim offsets, mapname, or selecting coarse and fine shims.
- 'writeb0' displays the b0 plot calculated from the first two array elements.

'shimmap' is a keyword to run a shim mapping experiment and save the results (same as Make Shimmap button).

shimmap option is one of the following values:

- 'auto' is a keyword to calibrate gzwin and then make a shimmap (same as Automake Shimmap button).
- 'manual' is a keyword to use shim offset values set manually from the file gshim.list and not the default values to make a shimmap.
- 'overwrite' is a keyword to make a shimmap and overwrite the current mapname if it exists.
- mapname is the prefix of the shimmap file name. The default is the user is queried for mapname before running the experiment.

See also: NMR Spectroscopy User Guide

Related: gmapshim
Start gradient autoshimming (M)

gmapz Get parameters and files for gmapz pulse sequence (M)

gradtype Gradients for X, Y, Z axes (P)

gzwin Spectral width percentage used for gradient shimming (P)

seqfil Pulse sequence name (P)

gmap\_z1z4 Gradient shimming flag to first shim z1-z4 (P)

Number of z-axis shims used by gradient shimming (P)

## gmapz Get parameters and files for gmapz pulse sequence (M)

Applicability: Systems with gradient shimming installed.

Syntax: gmapz<(mapname)>

Description: Retrieves gradient shimming parameters to set up a gradient shimming

experiment.

Arguments: mapname is the name of a gradient shimmap file that must exist in the

shimmaps directory. gmapz retrieves parameters and loads the shimmap file

from mapname. The default is to retrieve standard gradient shimming

parameters from the file gmapz.par.

See also: NMR Spectroscopy User Guide

Related: gmapshim Start gradient autoshimming (M)

gmapsys Run gradient autoshimming, set parameters, map shims (M)

gmap\_z1z4 Gradient shimming flag to first shim z1-z4 (P)

#### gmap findtof Gradient shimming flag to first find tof (P)

Applicability: Systems with gradient shimming installed.

Description: When the flag is set to 'y', gradient shimming first performs a calibration to

find tof before the start of shimming. This action is recommended for only homospoil deuterium gradient shimming with different solvents. The default

value is 'n'.

Values: 'y' turns on the flag.

'n' turns off the flag.

See also: NMR Spectroscopy User Guide

Related: gmapshim Start gradient autoshimming (M)

gmapsys Run gradient autoshimming, set parameters, map shims (M)
gmapz Get parameters and files for gmapz pulse sequence (M)

Frequency offset for observe transmitter (P)

## gmap z1z4 Gradient shimming flag to first shim z1-z4 (P)

Applicability: Systems with gradient shimming installed.

Description: When the flag is set to 'y', if gzsize is greater than 4, gradient shimming

first shims on z1-z4, and then uses all shims specified by gzsize. When the

flag is set to 'n' (default), all shims specified by gzsize are used.

Values: 'y' turns on the flag.

'n' turns off the flag.

See also: NMR Spectroscopy User Guide

Related: gmapshim Start gradient autoshimming (M)

gmapsys Run gradient autoshimming, set parameters, map shims (M)
gmapz Get parameters and files for gmapz pulse sequence (M)
gzsize Number of z-axis shims used by gradient shimming (P)

G

gmax Maximum gradient strength (P)

Description: The allowed maximum gradient level (absolute value) in gauss/cm. gmax is one

of the calibration entries in a gradtables file. gxmax, gymax, and gzmax are used when the maximum gradient level is different for each axis in gauss/

cm, which is the case for triple-axis PFG coils.

See also: VnmrJ Installation and Administration; VnmrJ Imaging NMR

Related: gcoil Current gradient coil (P)

gxmax,gymax,gzmax Maximum gradient strength for each axis (P)

sysgcoil System gradient coil (P)

gmqcosy Set up PFG absolute-value MQF COSY parameter set (M)

Applicability: Systems with the pulsed field gradient module.

Description: Converts a 1D standard two-pulse sequence parameter set into a parameter set

ready to run a PFG (pulsed field gradient) absolute-value MQF COSY

experiment.

See also: NMR Spectroscopy User Guide

gnoesy Set up a PFG NOESY parameter set (M)

Applicability: Systems with the pulsed field gradient module.

Description: Converts a 1D standard two-pulse sequence parameter set into a parameter set

ready to run a PFG (pulsed field gradient) NOESY experiment, either absolute

value or phase sensitive.

See also: NMR Spectroscopy User Guide

go Submit experiment to acquisition (M)

Syntax: go<(<'acqi'><,'nocheck'><,'nosafe'><,'next'>

<,'sync'><,'wait'>)>

Description: Performs the experiment described by the current acquisition parameters,

checking parameters loc, spin, gain, wshim, load, and method to determine the necessity to perform various actions in addition to data acquisition. This may involve a single FID or multiple FIDs, as in the case of arrays or 2D experiments. go acquires the FID and performs no processing. If free disk space is insufficient for the complete 1D or 2D FID data set to be acquired, go prompts the user with an appropriate message and aborts the

acquisition initiation process.

Before starting the experiment, go executes two user-created macros if they exist. The first is usergo, a macro that allows the user to set up general conditions for the experiment. The second is a macro whose name is formed by go\_ followed by the name of the pulse sequence (from seqfil) to be used (e.g., go\_s2pul, go\_dept). The second macro allows a user to set up

experiment conditions suited to a particular sequence.

Arguments: 'acqi' is a keyword to submit an experiment for display by the acqi

program. All operations explained above are performed, except acquisition of data is not initiated. The instructions to control data acquisition are stored so that acqi can acquire the data when the FID button is clicked. The gf macro is recommended instead of running go ('acqi') directly. Using gf prevents certain acquisition events from occurring, such as spin control and temperature

change. See the description of gf for more information.

'nocheck' is a keyword to override checking if there is not enough free disk space for the complete 1D or 2D FID data set to be acquired.

'nosafe' is a keyword to disable probe protection during the experiment.

'next' is a keyword to put the experiment started with go ('next') at the head of the queue of experiments to be submitted to the acquisition system. If go ('next') is entered, the go macro remains active until the experiment is submitted to the acquisition system, and no other VnmrJ commands are processed until the go macro finishes.

'sync' is a keyword in nonautomation mode that accomplishes the same effect as go ('next') in synchronizing VnmrJ command execution with the submission of experiments to the acquisition system. The difference is that 'sync' does not put the experiment at the head of the queue.

'wait' is a keyword to stop submission of experiments to acquisition until wexp processing of the experiment, started with 90 ('wait'), is finished.

Examples: 90

See also:

go('nosafe')
go('next')

NMR Spectroscopy User Guide

Related: acqi Interactive acquisition display process (C)

au Submit experiment to acquisition and process data change Submit a change sample experiment to acquisition (M)

gain Receiver gain (P)

ga Submit experiment to acquisition and FT the result (C)
gf Prepare parameters for FID/spectrum display in acqi (M)
go Pulse sequence setup macro called by go, ga, and au (M)

Load status of displayed shims (P)
Location of sample in tray (P)

lock Submit an Autolock experiment to acquisition (C)

method Autoshim method (P)
probe protection Probe protection control (P)

Submit change sample, Autoshim exp. to acquisition (M)

seqfil Pulse sequence name (P)

shimSubmit an Autoshim experiment to acquisition (C)spinSubmit a spin setup experiment to acquisition (C)

spin Sample spin rate (P)

Submit a setup experiment to acquisition (M)

usergo Experiment setup macro called by go, ga, and au (M)

vnmrj cmd () Commands to invoke the GUI popup (C) wshim Conditions when shimming is performed (P)

# go\_ Pulse sequence setup macro called by go, ga, and au (M)

Syntax: go macro

Description: Called by the macros qo, qa, or au before starting an experiment. The user

typically creates this macro to set up general experiment conditions. The name of the macro is formed by combining go\_ with the name of the pulse sequence

macro (from seqfil) to be used.

Examples: go dept

go\_noesy
go\_s2pul

See also: NMR Spectroscopy User Guide

Related: au Submit experiment to acquisition and process data (M)

ga Submit experiment to acquisition and FT the result (M)

go Submit experiment to acquisition (M)

seqfil Pulse sequence name (P)

usergo Experimental setup macro called by go, ga, and au (M)

gpat-gpat3 Gradient shape (P)

Description: Predefined string parameters available to specify gradient shapes.

See also: VnmrJ Imaging NMR

gplan Start interactive image planning (C)

Syntax: gplan(function name, arg1, arg2,...)

Description: In VnmrJ, starts an image planning session.

Arguments: 'function name', path is the name of an image planning function

surrounded by single quotation marks.

arg1, arg2, ... are arguments for the function, if relevant.

Examples: gplan 'clearStacks()'

get 'PrevStacks()'

See also: NMR Spectroscopy User Guide

## gradientdisable Disable PFG gradients (P)

Description: gradientdisable is an optional global parameter for disabling the gradient

pulses. If gradientdisable parameter is set to 'y', the psg software sets the gradient dac values to 0. The gradient parameters in VnmrJ and pulse sequence are not altered. This feature works in both C psg and SpinCAD Jpsg.

To use this feature, create gradientdisable as a global parameter of type 'flag'. If gradientdisable is set to 'y', the gradient amplitude values will be set to 0; if set to 'n' the gradient amplitudes will be the expected values determined by the gradient parameters and pulse sequence calculations. This feature is typically used in experiments involving Cold Probes. This feature is only effective for gradient configurations, gradtypes of 'l', 'p', and 't'.

Related: pfgon Pulsed field gradient amplifiers on/off control (P)

gradtype Gradients for X, Y, and Z axes (P)

### gradientshapingActivate shaping on the gradient pulses (P)

Applicability: Systems with Varian, Inc. Cold Probes

Description: Activate shaping on the gradient pulses in the pulse sequence without changing

the pulse sequence source program. This feature works only the Z gradient

pulses, specified using the zgradpulse (...) PSG statement.

gradientshaping is a global parameter.

Values: gradientshaping='y' enables this feature and produces a WURST

shaping of gradient amplitudes.

gradientshaping='n' or destroy the parameter disables this feature and

produces rectangular gradients amplitudes.

# gradstepsz Gradient step size (P)

Description: The maximum gradient DAC value. gradstepsz determines the type of

gradient DAC board used in the system: 12-bit or 16-bit. It is used internally to

convert gauss/cm gradient levels to the proper hardware DAC level.

Values: Systems with 12-bit DACs (older SISCO spectrometers without gradient

waveform capabilities): -2047 to +2047 units, in integer steps.

Systems with 16-bit DACs (SISCO spectrometers with gradient waveform

capabilities): -32767 to +32767 units, in integer steps.

See also: VnmrJ Installation and Administration; VnmrJ Imaging NMR

# gradtype Gradients for X, Y, and Z axes (P)

Applicability: Systems with pulsed field gradient (PFG) or imaging capability.

Description: Configuration parameter for systems with optional gradients for axes. The value

is set using the label X Axis, Y Axis, Z Axis in the Spectrometer Configuration window (opened from config). The values available for each axis are None, WFG + GCU, Performa I, Performa II/III, Performa II/III + WFG, Performa XYZ, Performa XYZ + WFG, SIS (12 bit), Homospoil, and Shim DAC. WFG stands for the waveform generator; GCU stands for the gradient compensation

unit; and Performa I, II, III, and XYZ are types of PFG modules.

Values: String of three characters (e.g., 'nnp'). The first character is the gradient for

the X axis, second for the Y axis, and third for the Z axis. Each axis has value

'n' (None choice in Spectrometer Configuration window), 'w'

(WFG+GCU), 'l' (Performa I), 'p' (Performa II/III), 'q' (Performa II/III + WFG), 't' (Performa XYZ), 'u' (Performa XYZ + WFG), 's' (SIS (12 bit),

or 'h' (Homospoil). Homospoil is functional only for the Z axis.

See also: VnmrJ Installation and Administration; NMR Spectroscopy User Guide

Related: config Display current configuration and possibly change it (M)

pfgon PFG amplifiers on/off control (P)

## graphis Return the current graphics display status (C)

Syntax: (1) graphis:\$display command

(2) graphis (command): \$yes no

Description: Determines what command currently controls the graphics window.

Arguments: \$display command is a return value set to the name of the currently

controlling command.

command is the name of a command to be checked.

\$yes\_no is a return value set to 1 if the command name given by the command argument is controlling the graphics window, or set to 0 if it is not

controlling the window.

Examples: graphis:\$display

if (\$display='ds') then

... endif

graphis('ds'):\$ds\_on
if (\$ds on) then

... endif

See also: User Programming

Related: textis Return the current text display status (C)

#### grayctr Gray level window adjustment (P)

Description: Controls the grayscale display available in dcon. In the dconi program, the

center mouse button controls the grayscale bar, which changes the mean gray level and hence the value of grayctr. The grayctr parameter (along with the parameter graysl) records the current settings of the gray bar as the interaction changes; the value can also be set directly. The right mouse button controls the data level of the maximum data intensity. To create grayctr,

enter create('grayctr','real')
setgroup('grayctr','display')
setlimit('grayctr',64,0,1).

To create the set of imaging parameters grayctr, dcrmv and graysl, and in the current experiment, enter addpar ('image').

Values: 0 to 64 (typically 32)

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

dcon Display noninteractive color intensity map (C)

dconi Interactive 2D contour display (C)
graysl Gray level slope (contrast) adjustment (P)

## graysl Gray level slope (contrast) adjustment (P)

Description: Controls the grayscale display available in dcon. In the dconi program, the

center mouse button controls the grayscale slope as applied to the data changes and hence the value of <code>graysl</code>. Negative values of <code>graysl</code> will invert black and white; however, negative values can be set only from the keyboard. <code>graysl</code> (along with the parameter <code>grayctr</code>) records the current settings of the gray bar as the interaction changes; the value can also be set directly. The right mouse button controls the data level of the maximum data intensity. To

create grays1, enter the following command:

create('graysl','real') setgroup('graysl','display')
setlimit('graysl',10,-10,0.1)

To create the set of imaging parameters grays1, dcrmv, and grayctr in the current experiment, enter addpar ('image').

Values: -10 to +10 (-100 to +100, typically 1)

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

dcon Display noninteractive color intensity map (C)

dconi Interactive 2D contour display (C)
grayctr Gray level window adjustment (P)

## grecovery Eddy current testing (M)

Applicability: Systems with pulsed field gradient.

Description: Conditions an experiment for eddy current testing so that it is compatible with

standard installation procedures.

See also: Pulsed Field Gradient Modules Installation, NMR Spectroscopy User Guide

# grid Draw a grid on a 2D display (M)

```
Syntax: (1) grid<(<spacing><,><color>)>
```

(2) grid<(start\_f2,incr\_f2,start\_f1,incr\_f1<,color>)>

Description: Draws grid lines over a 2D display. Grid lines are drawn on the graphics screen

in the XOR mode—entering a second grid command with identical arguments

erases (not redraws) the grid displayed by the first command.

Arguments: spacing specifies the approximate spacing of the grid lines, in cm. The

default is intervals of approximately 1 cm, rounded so that the intervals fall at a

multiple of 1, 2, or 5 (in Hz), or 1p, 2p, or 5p (in ppm).

color specifies the color of the grid lines and is one of the following

keywords: 'red', 'green', 'blue', 'cyan', 'magenta', 'yellow',

'black', or 'white'. The default is 'blue'.

start\_f2, incr\_f2, start\_f1, incr\_f1 define a grid by supplying the starting and increment frequencies for f2 and f1. Add the p suffix to a value to

enter it in ppm (see third example below).

Examples: grid

grid(1.5, 'red')

grid(1p,0.5p,3p,0.5p)

See also: NMR Spectroscopy User Guide

Related: plgrid Plot a grid on a 2D plot (M)

## groupcopy Copy parameters of group from one tree to another (C)

Syntax: groupcopy(from tree, to tree, group)

Description: Copies a set of parameters of a group from one parameter tree to another.

Arguments: from tree, to tree are two different parameter trees, each given by the

one of the keywords 'global', 'current', or 'processed'. Refer to

the **create** command for more information on trees.

group is the set of parameters to be copied and is one of the keywords 'all',

'sample', 'acquisition', 'processing', and 'display'.

Examples: groupcopy('processed','current','acquisition')

See also: User Programming

Related: create new parameter in a parameter tree (C)

destroy Destroy a parameter (C)

destroygroupDestroy parameters of a group in a tree (C)displayDisplay parameters and their attributes (C)setgroupSet group of a parameter in a tree (C)

# gspoil Spoiler gradient level (P)

Description: Predefined parameter to set a spoiler gradient level.

## gsspat Slice-select gradient shape (P)

Description: Predefined string parameter to specify a slice-select gradient shape.

# gtnnoesy Set up a PFG TNNOESY parameter set (M)

Applicability: Systems with the pulsed field gradient (PFG) module.

Description: Converts a 1D standard two-pulse sequence parameter set into a parameter set

ready to run a PFG NOESY experiment (either absolute value or phase

sensitive) or a gtnnoesy experiment.

G

gtnroesy Set up a PFG absolute-value ROESY parameter set (M)

Applicability: Systems with the pulsed field gradient (PFG) module.

Description: Converts a 1D standard two-pulse sequence parameter set into a parameter set

ready to run a PFG absolute-value ROESY experiment or a gtnroesy

experiment.

gtotlimit Gradient total limit (P)

Applicability: Systems with three-axis gradients

Description: Sets the gradient limit, in gauss/cm, of the x, y, and z axes, summed together.

This parameter is taken from an entry of the same name in a gradient table and should only exist if a gradient amplifier limits the combined output of all three

gradient axis.

Related gcoil Read data from gradient calibration tables (P)

gtrim Trim gradient level (P)

Description: Predefined parameter to set a trim gradient level.

gxmax, gymax, gzmaxMaximum gradient strength for each axis (P)

Applicability: Systems with three-axis gradients.

Description: Defines the maximum gradient strength, in gauss/cm, for each gradient axis.

These values are read in from the selected system gradient table whenever the parameter set is retrieved or the gradient coil defined by gcoil has changed. When the values are read in, gmax is set to the lowest value of the three.

The parameters gxmax, gymax, and gzmax are used instead of gmax when the gradients strengths are not equal for each axis. Unequal gradient strengths per axis are generally true for systems with three-axis PFG coils, which have a strong z gradient, and can be true for microimaging systems. Horizontal-bore imaging systems usually have gradients set to the same maximum value, and

gmax can be used.

See also: NMR Spectroscopy User Guide; User Programming, VnmrJ Imaging NMR

Related: gcoil Read data from gradient calibration tables (P)

gmax Maximum gradient strength (P)

gzlvl Pulsed field gradient strength (P)

Applicability: Systems with gradient shimming installed.

Description: Specifies the pulsed field gradient DAC value.

Values: Range from +2047 to -2048 for 12-bit gradient module, and from +32767 to

-32768 for a 16-bit gradient module.

Related: gzsize Number of z-axis shims used by gradient shimming (P)

gzwin Spectral window percentage used for gradient shimming (P)

gzsize Number of z-axis shims used by gradient shimming (P)

Applicability: Systems with gradient shimming installed.

Description: Specifies the number of z-axis shims used by gradient shimming. For example,

gzsize set to 4 means that gradient shimming uses shims z1 to z4. By default,

coarse shims are used if present, as determined by the shimset value

Values: Integer from 1 to 8.

Related: gmapshim Start gradient autoshimming (M)

gmapsys Run gradient autoshimming, set parameters, map shims (M)
gmapz Get parameters and files for gmapz pulse sequence (M)

gzlvl Pulsed field gradient strength (P)

gzwin Spectral width percentage used by gradient shimming (P)

shimset Type of shimset (P)

gmap\_z1z4 Gradient shimming flag to first shim z1-z4 (P)

# gzwin Spectral width percentage used for gradient shimming (P)

Applicability: Systems with gradient shimming installed.

Description: Specifies the percentage of the spectral width sw used by gradient shimming for

shimmap calculations. The value is set automatically with the buttons Find gzlvl/gzwin and Find gzwin in the gradient shimming system menu

opened by gmapsys.

Values: A real number between 0 and 100. The typical value is 50.

Related: gmapshim Start gradient autoshimming (M)

gmapsys Run gradient autoshimming, set parameters, map shims (M)
gmapz Get parameters and files for gmapz pulse sequence (M)

gzlvl Pulsed field gradient strength (P)

gzsize

Number of z-axis shims used by gradient shimming (P)

sw

Spectral width in directly detected dimension (P)

tof

Frequency offset for observe transmitter (P)

# Н

h1 Automated proton acquisition (M)
h1freq Proton frequency of spectrometer (P)
h1p Process 1D proton spectra (M)

h2cal Calculate strength of the decoupler field (C)

halt Abort acquisition with no error (C)

hc Automated proton and carbon acquisition (M)
hcapt Automated proton, carbon, and APT acquisition (M)
hcchtocsy Set up parameters for HCCHTOCSY pulse sequence (M)
hccorr Automated proton, carbon, and HETCOR acquisition (M)
hcdept Automated proton, carbon, and DEPT acquisition (M)

hcosy Automated proton and COSY acquisition (M)

hdmf Modulation frequency for the band selective homonuclear

decoupling (P)

hcmult

Execute protocol actions of apptype hcmult (M)

hdof

Frequency offset for homodecoupling (P)

hdpwr

Power level for homodecoupling (P)

Homodecoupling fine power (optional) (P)

hdres

Sets the tip angle resolution (P)

nates sets the up ungle resolution (1)

hdseq Sets the decoupler waveform filename (P)

hdwshim Hardware shimming (P)

hdwshimlist List of shims for hardware shimming (P)

het2dj Set up parameters for HET2DJ pulse sequence (M)

HETCOR Change parameters for HETCOR experiment (M)

hetcor Set up parameters for HETCOR pulse sequence (M)

hetcorcp1 Set up parameters for solids HETCOR pulse sequence (M)

hetcorps Set up parameters for HETCORPS pulse sequence (M)

hetcorps Execute protocol actions of apptype hetero2d (M)

hidecommand Execute macro instead of command with same name (C)

hipwrampenable High Power Amplifier Enable (P)

HmbcConvert the parameter to a HMBC experiment (M)HmqcConvert the parameter to a HMQC experiment (M)HMQC15Set up parameters for 15N HMQC experiment (M)

HMQC\_d2 Set up parameters for <sup>15</sup>N HMQC experiment using dec. 2 (M)
HMQC\_d213 Set up parameters for <sup>13</sup>C HMQC experiment using dec. 2 (M)

hmqcr Set up parameters for HMQCR pulse sequence (M)

Hmqctoxy Convert the parameter to a HMQCTOXY experiment (M)

HMQCTOXY15 Set up parameters for <sup>15</sup>N HMQCTOXY experiment (M)

HMQCTOXY\_d2 Set up parameters for <sup>15</sup>N HMQCTOXY using decoupler 2 (M)

HMQCTOXY\_d213 Set up parameters for <sup>13</sup>C HMQCTOXY using decoupler 2 (M)

hmqctoxy3d Set up parameters for HMQC-TOCSY 3D pulse sequence (M)

ho Horizontal offset (P)

hom2dj Set up parameters for HOM2DJ pulse sequence (M)
homo Homodecoupling control for the observe channel (P)

# Н

HOMODECChange parameters for HOMODEC experiment (M)homo2dExecute protocol actions of apptype homo2d (M)homorof1Delay before turning on homo decoupling rf (P)

homorof2 Delay after blanking the amp and setting T/R to receive (P)

homorof3 Delay between setting T/R switch to receive and gating the recvr on (P)

hoult Set parameters alfa and rof2 according to Hoult (M)
hpa Plot parameters on special preprinted chart paper (C)

hregions Select integral regions in proton spectrum (M)

hs Homospoil pulses (P)

Hsqc Convert the parameter to a HSQC experiment (M)
HSQC15 Set up parameters for <sup>15</sup>N HSQC experiment (M)

HSQC\_d2 Set up parameters for <sup>15</sup>N HSQC experiment using dec. 2 (M)
HSQC\_d213 Set up parameters for <sup>13</sup>C HSQC experiment using dec. 2 (M)

hsqcHT Set up the hsqcHT experiment (M)

HSQCTOXY Convert parameters to a HSQCTOXY experiment (M)

HSQCTOXY15 Set up parameters for <sup>15</sup>N HSQCTOXY experiment (M)

HSQCTOXY\_d2 Set up parameters for <sup>15</sup>N HSQCTOXY using decoupler 2 (M)

HSQCTOXY\_d213 Set up parameters for <sup>13</sup>C HSQCTOXY using decoupler 2 (M)

hsqctoxySE Set up parameters for HSQC-TOCSY 3D pulse sequence (M)

hsrotor Display rotor speed for solids operation (P)

hst Homospoil time (P)

htbitrev Hadamard bit reversal flag (P)

httbwl Hadamard pulse excitation bandwidth in ni (P)
httall RF calibration flag for Hadamard waveforms in ni (P)

htfrq1 Hadamard frequency list in ni (P)

htofs1 Hadamard offset in ni (P)

Power level for RF calibration of Hadamard waveforms in ni (P)

htss1 Stepsize for Hadamard waveforms in ni (P)

hzmm Scaling factor for plots (P)

hztomm Convert locations from Hz or ppm to plotter units (C)

## h1 Automated proton acquisition (M)

Syntax: h1<(solvent)>

Description: Prepares parameters for automatically acquiring a standard <sup>1</sup>H spectrum. The

parameter wexp is set to 'procplot' for standard processing. If h1 is used as the command for automation via the enter command, then au is supplied automatically and should not be entered on the MACRO line of the enter program. However, it is possible to customize h1 on the MACRO line by following it with additional commands and parameters. (e.g., entering h1 nt=1

uses the standard h1 setup but with only one transient).

Arguments: solvent is the name of the solvent. In automation mode, the solvent is

supplied by the enter program. The default is 'CDC13'.

Examples: h1

h1('DMSO')

See also: NMR Spectroscopy User Guide

Related: Submit experiment to acquisition and process data (M)

> Enter sample information for automation run (C) enter

h1p Process 1D proton spectra (M) Automatically process FIDs (M) procplot When experiment completes (P) wexp

h1freq Proton frequency of spectrometer (P)

Configuration parameter for the resonance frequency of <sup>1</sup>H as determined by Description:

the field strength of the magnet. The value is set using the label Proton

Frequency in the Spectrometer Configuration window.

085, 100, 200, 300, 400, 500, 600, 700, 750, 800, 900 (in MHz); 3T, 4T. Values:

See also: VnmrJ Installation and Administration

Related: Display current configuration and possibly change it (M) config

h1p Process 1D proton spectra (M)

> Processes non-arrayed 1D proton spectra using standard macros. h1p is called Description:

> > by proc1d, but can also be used directly. Fully automatic processing (up to a point where a spectrum could be plotted) is provided: Fourier transformation (using preset weighting functions), automatic phasing (aphx macro), select integral regions (hregions macro), adjust integral size (integrate macro), vertical scale adjustment (vsadjc macro), avoiding excessive noise (noislm macro), threshold adjustment (if required, thadj macro), and referencing to the TMS signal if present (setref macro, then tmsref

macro).

See also: NMR Spectroscopy User Guide

Related: aphx Perform optimized automatic phasing (M)

> h1 Automated proton acquisition (M)

hregions Select integral regions for proton spectra (M) Automatically integrate 1D spectrum (M) integrate

noislm Avoids excessive noise (M)

proc1d Processing macro for simple (non-arrayed) spectra (M) setref

Set frequency referencing for proton spectra (M)

thadj Adjust threshold (M)

tmsref Reference spectrum to TMS line (M) vsadjh Adjust vertical scale for proton spectra (M)

Calculate strength of the decoupler field (C) h2cal

h2cal<(j1r,j2r<,j0>)><:gammah2,pw90,frequency>

Description: Calculates the strength of the decoupler field. It uses the results from two

> experiments: one with the decoupler off-resonance at a lower frequency and the other with the decoupler off-resonance at a higher frequency than the frequency

of the peak being decoupled.

Arguments: jlr is the frequency of the decoupler during these two experiments;. The

default is that h2cal prompts for a value. If the parameter dof is arrayed and has two values, h2cal assumes these two values represent the decoupler frequencies; if dof is arrayed and has more than two values, h2cal prompts

for the two decoupler frequencies.

j2r is the reduced coupling constants from the two experiments. The default is that h2cal prompts for a value

j 0 is the full coupling constant that results when no decoupling is done. The default is a value of 142 Hz, the constant for the standard sample dioxane, or 15 Hz for the methyl iodide sample.

gammah2 is a return value set to the strength of the decoupler field.

pw90 is a return value set to the pulse width of a 90° pulse from the decoupler. It is related to the value of parameter  $\frac{dmf}{dmf}$  through the equation  $\frac{dmf}{dmf} = 1/\frac{pw90}{pw90}$ .

frequency is a return value set to the coalescence point (i.e., frequency at which single-frequency decoupling would collapse the dioxane to a singlet).

See also: NMR Spectroscopy User Guide

Related: dmf Decoupler modulation frequency for first decoupler (P)

dof Frequency offset for first decoupler (P)

## halt Abort acquisition with no error (C)

Syntax: halt

Description: Aborts an experiment that has been submitted to acquisition. If the experiment

is active, it is aborted immediately, all data is discarded, and the experiment is interpreted as complete. Any data collected from an earlier block size transfer is retained. If any wexp processing is defined, that processing then occurs, followed by any queued experiments. The login name, and the FID directory path in file are used as keys to find the proper experiment to abort.

Under some circumstances, there is a delay between the time **go** is entered and the acquisition is started. During this time, instructions based on the selected pulse sequence are being generated. This is signified by the letters "PSG" appearing in the upper left corner of the status window. A halt command issued under these circumstances reports that no acquisition is active but it instead stops the instruction generation process and displays "PSG aborted".

See also: NMR Spectroscopy User Guide

Related: aa Abort acquisition with error (C)

File name of parameter set (P)
Submit experiment to acquisition (C)

wexp Specify action when experiment completes (C)

wexp When experiment completes (P)

## hc Automated proton and carbon acquisition (M)

Syntax: hc<(solvent)>

Description: Combines the operation of the h1 and c13 macros. In non-automation mode,

both spectra are acquired in the experiment in which the hc macro was entered. After the completion of the acquisition, rttmp can be used for further

processing of the two spectra.

Arguments: solvent is the solvent name In automation mode, the enter program

supplies the value. In non-automation mode, the default is 'cdcl3'.

Examples: hc

hc('dmso')

See also: NMR Spectroscopy User Guide

Related: c13 Automatic carbon acquisition (M)

enter Enter sample information for automation run (M,U)

h1 Automated proton acquisition (M)

rttmp Retrieve experiment data from experiment subfile (M)

hcapt Automated proton, carbon, and APT acquisition (M)

Syntax: hcapt<(solvent)>

Description: Combines the operation of the h1 and c13 macros and the APT experiment. In

non-automation mode, all spectra are acquired in the experiment in which the hcapt macro was entered. After acquisition completes, rttmp can be used for

further processing of the three spectra.

Arguments: solvent is the solvent name. In automation mode, the enter program

supplies the value. In non-automation mode, the default is 'cdcl3'.

Examples: hcapt

hcapt('dmso')

See also: NMR Spectroscopy User Guide

Related: Apt Set up parameters for APT experiment (M)

Automatic carbon acquisition (M)

enter Enter sample information for automation run (M,U)

h1 Automated proton acquisition (M)

rttmp Retrieve experiment data from experiment subfile (M)

hcchtocsy Set up parameters for HCCHTOCSY pulse sequence (M)

Description: Used for sidechain assignments in fully <sup>13</sup>C-enriched molecules.

See also: NMR Spectroscopy User Guide

hccorr Automated proton, carbon, and HETCOR acquisition (M)

Syntax: hccorr<(solvent)>

Description: Combines the operation of the h1 and c13 macros and the HETCOR

experiment. In non-automation mode, all spectra are acquired in the experiment in which hccorr is entered. After acquisition completes, rttmp can be used

for further processing of the three spectra.

Arguments: solvent is the solvent name. In automation mode, the enter program

supplies the value. In non-automation mode, the default is 'cdcl3'.

Examples: hccorr

hccorr('dmso')

See also: NMR Spectroscopy User Guide

Related: c13 Automated carbon acquisition (M)

enter Enter sample information for automation run (M,U)

h1 Automated proton acquisition (M)

hetcor Set up parameters for HETCOR experiment (M)
rttmp Retrieve experiment data from experiment subfile (M)

hcdept Automated proton, carbon, and DEPT acquisition (M)

Syntax: hcdept<(solvent)>

Description: Combines the operation of the h1 and c13 macros and the DEPT experiment.

In non-automation mode, all spectra are acquired in the experiment in which hcdept was entered. After the completion of the acquisition, rttmp can be

used for further processing of the three spectra.

Arguments: solvent is the solvent name. In automation mode, the <a href="mailto:enter">enter</a> program

supplies the value. In non-automation mode, the default is 'cdcl3'.

Examples: hcdept

hcdept('dmso')

See also: NMR Spectroscopy User Guide

Related: C13 Automatic carbon acquisition (M)

Dept Set up parameters for DEPT experiment (M)
enter Enter sample information for automation run (M,U)

h1 Automated proton acquisition (M)

rttmp Retrieve experiment data from experiment subfile (M)

hcosy Automated proton and COSY acquisition (M)

Syntax: hcosy<(solvent)>

Description: Combines the operation of the h1 macro and the COSY experiment. In non-

automation mode, both spectra are acquired in the experiment in which hcosy

is entered. After acquisition completes, rttmp can be used for further

processing of the two spectra.

Arguments: solvent is the solvent name. In automation mode, the enter program

supplies the value. In non-automation mode, the default is 'cdcl3'.

Examples: hcosy

hcosy('dmso')

See also: NMR Spectroscopy User Guide

Related: enter Enter sample information for automation run (C)

h1 Automated proton acquisition (M)

rttmp Retrieve experiment data from experiment subfile (M)

hdmf Modulation frequency for homonuclear decoupling (P)

Applicability: VNMRS liquids, 400 MR

Syntax: hdmf=<value>

Description: Sets the modulation frequency for the band selective homonuclear decoupling.

The parameter specifies 1/pw90 at the power value, hdpwr, used for homonuclear decoupling. The parameter is not used with single frequency

homonuclear decoupling.

Related: dutyc The rf duty cycle fraction for homonuclear decoupling (P)

hdof Frequency offset for homodecoupling (P)

hdpwr Sets the rf attenuator to control the power for homonuclear

decoupling (P)

hdpwrf Sets the rf linear modulator fine power for homonuclear

decoupling (P)

hdres Sets the tip angle resolution (P)

homo Sets the decoupler waveform filename (P)
homo Homodecoupling control for observe channel (P)
homorof1 Delay before turning on homo decoupling rf (P)

homorof2 Delay after blanking the amplifier and setting T/R switch to receive (P) homorof3 Delay between setting T/R switch to receive gating on the receiver (P)

tn Nucleus for observe transmitter (P)

hcmult Execute protocol actions of apptype hcmult (M)

Description: This macro is used to execute the protocol actions of the hcmult apptype.

Examples: hcmult ('setup') - execute hcmult experimental setup

hcmult('process') - execute hcmult processing

hcmult ('plot') - execute hcmult plotting

See also: NMR Spectroscopy User Guide, VnmrJ Walkup

Related: apptype Application type (P)

execpars Set up the exec parameters (M)

## hdof Frequency offset for homodecoupling (P)

Applicability: VNMRS systems

Syntax: hodf=<value>

Description: Sets the irradiation frequency offset for homonuclear decoupling and similar to

how tof, and dof determine the frequency. The parameter is not used if

hdseq is set to a filename.

Values: -100000 to 100000 Hz in steps of 0.1 Hz.

Related: dutyc The rf duty cycle fraction for homonuclear decoupling (P)

hdmf modulation frequency for the band selective homonuclear

decoupling (P)

hdpwr Sets the rf attenuator to control the power for homonuclear

decoupling (P)

hdpwrf Homodecoupling fine power (optional) (P)

hdres Sets the tip angle resolution (P)

homo Sets the decoupler waveform filename (P)
homo Homodecoupling control for observe channel (P)
homorof1 Delay before turning on homo decoupling rf (P)

homorof2 Delay after blanking the amplifier and setting T/R switch to receive (P) homorof3 Delay between setting T/R switch to receive gating on the receiver (P)

Nucleus for observe transmitter (P)

#### hdpwr Power level for homodecoupling (P)

Applicability: VNMRS systems, 400 MR

Syntax: hdpwr=<value>

Description: Sets the rf attenuator to control the power for homonuclear decoupling. The

dutyc parameter must be accounted for when setting hdpwr.

Values: -16 to 50 dB

CAUTION: Homodecoupling power greater than 2 watts in a switchable probe will

damage the probe. Always carefully calibrate homodecoupling to avoid exceeding 2 watts. The maximum value for hdpwr is set to 49, corresponding to about 2 watts of power. The actual power delivered depends on the CW duty cycle. Before using close to the maximum value of power or duty cycle, ensure safe operation by measuring the

output power.

Related: dutyc The rf duty cycle fraction for homonuclear decoupling (P)

hdmf modulation frequency for the band selective homonuclear

decoupling (P)

hdof Frequency offset for homodecoupling (P)
hdpwrf Homodecoupling fine power (optional) (P)
hdres Sets the tip angle resolution (P)

hdseq Sets the decoupler waveform filename (P)

homo Homodecoupling control for observe channel (P)
homorof1 Delay before turning on homo decoupling rf (P)

homorof2 Delay after blanking the amplifier and setting T/R switch to receive (P) Delay between setting T/R switch to receive gating on the receiver (P)

tn Nucleus for observe transmitter (P)

hdpwrf Homodecoupling fine power (optional) (P)

Applicability: VNMRS liquids systems

Syntax: hdpwrf=<value>

Description: Sets the rf linear modulator fine power for homonuclear decoupling. The default

is 4095 if the variable does not exist. Attenuation is added to the attenuation set

by hdpwr.

Values: 0-4095

Related: dutyc The rf duty cycle fraction for homonuclear decoupling (P)

hdmf Modulation frequency for the band selective homonuclear

decoupling (P)

hdof Frequency offset for homodecoupling (P)

hdpwr Sets the rf attenuator to control the power for homonuclear

decoupling (P)

hdres Sets the tip angle resolution (P)

homo Sets the decoupler waveform filename (P)
homo Homodecoupling control for observe channel (P)
homorof1 Delay before turning on homo decoupling rf (P)

homorof2 Delay after blanking the amplifier and setting T/R switch to receive (P) homorof3 Delay between setting T/R switch to receive gating on the receiver (P)

tn Nucleus for observe transmitter (P)

hdres Sets the tip angle resolution (P)

Applicability: VNMRS liquids systems

Syntax: hdres=<value>

Description: Sets the tip angle resolution to be used for the band selective waveform mode

of homonuclear decoupling. The parameter is not used with single frequency

homonuclear decoupling.

Values: 1 to 90 in units of degrees with 1 degree resolution

Related: dutyc The rf duty cycle fraction for homonuclear decoupling (P)

hdmf Modulation frequency for the band selective homonuclear

decoupling (P)

hdof Frequency offset for homodecoupling (P)

hdpwr Sets the rf attenuator to control the power for homonuclear

decoupling (P)

hdpwrf Sets the rf linear modulator fine power for homonuclear

decoupling (P)

homo Sets the decoupler waveform filename (P)
homo Homodecoupling control for observe channel (P)
homorof1 Delay before turning on homo decoupling rf (P)

homorof2 Delay after blanking the amplifier and setting T/R switch to receive (P) homorof3 Delay between setting T/R switch to receive gating on the receiver (P)

Nucleus for observe transmitter (P)

hdseq Waveform filename for band selective decoupling (P)

Applicability: VNMRS liquids systems

Syntax: hdseq='filename' — the file must have a.DEC. extension.

Description: Sets the decoupler waveform filename (.DEC extension) for the band selective

waveform mode. The irradiation frequency is determined by the transmitter offset last applied to the observe channel in the pulse sequence (typically tof) and any additional frequency offset from any phase modulation programmed

implicitly into the waveform .DEC file.

Examples: hdseq='' or does not exist — single frequency decoupling is used.

Related: dutyc

hdmf modulation frequency for the band selective homonuclear

decoupling (P)

hdof Frequency offset for homodecoupling (P)

hdpwr Sets the rf attenuator to control the power for homonuclear

decoupling (P)

hdpwrf Sets the rf linear modulator fine power for homonuclear

decoupling (P)

hdres Sets the tip angle resolution (P)

homo Homodecoupling control for observe channel (P) homorof1 Delay before turning on homo decoupling rf (P)

homorof2 Delay after blanking the amplifier and setting T/R switch to receive (P) homorof3 Delay between setting T/R switch to receive gating on the receiver (P)

Nucleus for observe transmitter (P)

# hdwshim Hardware shimming (P)

Applicability: Systems with additional Z1 shimming hardware.

Description: Allows go, su, au, etc., to turn on and off shimming hardware. Hardware

shimming is automatically suspended during software autoshimming.

Hardware shimming is only active during acquisition (go, ga, au). hdwshim

is a global parameter, so it affects all experiments.

Values: 'y' turns hardware shimming on.

'p' turns hardware shimming on during presaturation pulse (power level

change followed by pulse).

'n' turns shimming off.

See also: NMR Spectroscopy User Guide

Related: au Submit experiment to acquisition and process data (C)

go Submit experiment to acquisition (C)

Submit a setup experiment to acquisition (M)

ga Submit experiment to acquisition and FT the result (M)

# hdwshimlist List of shims for hardware shimming (P)

Description: A global parameter that sets the shims to use during hardware shimming. If it

does not exist, hardware shimming uses z1 by default. To create the parameter,

use create('hdwshimlist','string','global').

Values: Any string composed of z1, z1c, z2, z2c, x1, y1. Commas and blank space

are ignored. Shimming is done in the order z1, z2, x1, y1, regardless of the order

in the string.

# Н

Examples: hdwshimlist='z1'

hdwshimlist='z1z2x1y1'

See also: NMR Spectroscopy User Guide

Related: create Create new parameter in a parameter tree (C)

hdwshim Hardware shimming (P)

het2dj Set up parameters for HET2DJ pulse sequence (M)

Description: Sets up a HET2DJ (heteronuclear 2D-J) experiment.

See also: NMR Spectroscopy User Guide

Related: foldj Fold J-resolved 2D spectrum about fl=0 axis (C)

HETCOR Change parameters for HETCOR experiment (M)

Description: Converts the current parameter set to a HETCOR experiment. This is a phase-

sensitive, multiplicity-selected experiment.

hetcor Set up parameters for HETCOR pulse sequence (M)

Syntax: hetcor<(exp number)>

Description: Sets up a HETCOR (heteronuclear chemical shift correlation) experiment.

Arguments: exp number is the number of the experiment, from 1 to 9, in which a proton

spectrum of the sample already exists.

See also: NMR Spectroscopy User Guide

Related: plhxcor Plot X,H-correlation 2D spectrum (M)

ppcal Proton decoupler pulse calibration (M)

hetcorcp1 Set up parameters for solids HETCOR pulse sequence (M)

Applicability: Systems with the solids module.

Description: Sets up a parameter set, obtained with XPOLAR1, for HETCORCP1, the solid-

state heteronuclear correlation experiment.

See also: User Guide: Solid-State NMR

Related: xpolar1 Set up parameters for XPOLAR1 pulse sequence (M)

hetcorps Set up parameters for HETCORPS pulse sequence (M)

Description: Sets up parameters for a heteronuclear chemical shift correlation experiment

(absolute value and phase sensitive).

See also: NMR Spectroscopy User Guide

hetero2d Execute protocol actions of apptype hetero2d (M)

Applicability: Liquids

Description: Perform the actions for Homonuclear 2D protocols to set up, process, and plot

experiments.

Examples: hetero2d('setup') execute hetero2d experimental setup

hetero2d('process') execute hetero2d processing hetero2d('plot') execute hetero2d plotting

See also: NMR Spectroscopy User Guide, VnmrJ Walkup

Related: apptype Application type (P)

execpars Set up the exec parameters (M)

hidecommand Execute macro instead of command with same name (C)

Syntax: (1) hidecommand(command\_name) <: \$new\_name >

(2) hidecommand('?')

Description: Renames (or hides) a built-in VnmrJ command so that a macro with the same

name as the built-in command is executed instead of the built-in command.

Arguments: command name is the name of the command to be renamed. To reset the built-

in command back to its original name, enter hidecommand with the hidden

name as the argument.

\$new\_name returns the new name of the built-in command. By using this new

name, access is still available to the built-in command.

'?' is a keyword to display a list of all of the renamed built-in commands and

their original names.

Examples: hidecommand('sys'):\$newname

hidecommand('Sys')
hidecommand('?')

See also: System Administration; User Programming

Related: which Display which macro or command is used (M)

# hipwrampenable High Power Amplifier Enable (P)

Applicability: VNMRS solids and systems with high power amplifiers.

Description: This parameter controls the High/Low Power Relay. If the parameter does not

exist low power is used. If the parameter exists and the field corresponding to the physical channel is 'n' then low power is used. If the parameter exists and the field corresponding to the physical channel is 'y' then high power is used.

The parameter is created in the current tree as a flag with

create('hipwrampenable','flag').

Values: 'y' Enable high power

'n' Enable low power and disable high power

Examples: hipwrampenable='yny'

Physical channel 1 and 3 are high power enabled. Physical channel 2 is low

power.

Hmbc Convert the parameter to a HMBC experiment (M)

Description: Convert the parameter to a HMBC experiment.

See also: NMR Spectroscopy User Guide

Hmqc Convert the parameter to a HMQC experiment (M)

Description: Convert the parameter to a HMQC experiment.

HMOC15 Set up parameters for <sup>15</sup>N HMQC experiment (M)

Description: Converts the current parameter set to a HMQC experiment for <sup>15</sup>N.

Н

HMQC d2 Set up parameters for <sup>15</sup>N HMQC experiment using dec. 2 (M)

Description: Converts the current parameter set to a HMQC experiment for <sup>15</sup>N with

decoupler 2 as <sup>15</sup>N.

HMQC\_d213 Set up parameters for <sup>13</sup>C HMQC experiment using dec. 2 (M)

Description: Converts the current parameter set to a HMQC experiment for <sup>13</sup>C with

decoupler 2 as <sup>13</sup>C.

hmqcr Set up parameters for HMQCR pulse sequence (M)

Applicability: Not needed in current systems. Normally was used in systems with a <sup>1</sup>H only

decoupler.

Description: Sets up a HMQC (heteronuclear multiple-quantum coherence) experiment with

"reverse" configuration.

See also: NMR Spectroscopy User Guide

Hmqctoxy Convert the parameter to a HMQCTOXY experiment (M)

Description: Convert the parameter to a HMQCTOXY experiment.

HMQCTOXY15 Set up parameters for <sup>15</sup>N HMQCTOXY experiment (M)

Description: Converts the current parameter set to a HMQCTOXY experiment for <sup>15</sup>N.

HMQCTOXY d2 Set up parameters for <sup>15</sup>N HMQCTOXY using decoupler 2 (M)

Description: Converts the current parameter set to a HMQCTOXY experiment for <sup>15</sup>N with

decoupler 2 as <sup>15</sup>N.

HMQCTOXY d213 Set up parameters for <sup>13</sup>C HMQCTOXY using decoupler 2 (M)

Description: Converts the current parameter set to a HMQCTOXY experiment for <sup>13</sup>C with

decoupler 2 as <sup>13</sup>C.

hmqctoxy3d Set up parameters for HMQC-TOCSY 3D pulse sequence (M)

Description: Sets up parameters for a HMQC-TOCSY 3D experiment with a presaturation

option.

ho Horizontal offset (P)

Description: Horizontal offset of the each spectrum in a "stacked display" with respect to the

previous spectrum,. For 1D data sets, the parameter vo sets the vertical offset. For 2D data sets, the parameter wc2 sets the vertical distance (in mm) between

the first and last traces.

Values: Number, in mm, for offset size. For a "left-to-right" presentation, ho is typically

negative; for "bottom-to-top" presentation, vo or wc2 is positive.

hom2dj Set up parameters for HOM2DJ pulse sequence (M)

Description: Sets up a HOM2DJ (homonuclear J-resolved 2D) experiment.

See also: NMR Spectroscopy User Guide

homo Homodecoupling control for the observe channel (P)

Applicability: VNMRS liquids systems

Description: Homonuclear decoupling irradiates a single frequency if hdseq = '' (or if

hdseq does not exist) or a frequency band if hdseq = 'filename'. Pulse sequences do not require explicit homonuclear gating commands (homo function is similar to dm). A single RF channel, the observe channel, is used. The homo='y' setting cannot be used with pulse sequences containing explicit

acquire commands.

Syntax: homo=<'y' or 'n'>

Values: 'y' homonuclear decoupling rf and receiver gating is turned on during the

acquisition time. Provides single frequency or band selective (hdseq =

'filename') decoupling.

'n' homonuclear decoupling rf and receiver gating is turned off.

Related: hdof Frequency offset for homodecoupling (P)

hdpwr Power level for homodecoupling (P)
hdpwrf Homodecoupling fine power (P)

dutyc Duty cycle for homodecoupling (optional) (P)

Nucleus for observe transmitter (P)

homorof1 Delay before turning on homo decoupling rf (P)

homorof2 Delay after blanking the amplifier and setting T/R switch to receive (P) homorof3 Delay between setting T/R switch to receive gating on the receiver (P)

HOMODEC Change parameters for HOMODEC experiment (M)

Description: Converts the current parameter set to a HOMODEC experiment. A 1D proton

spectrum is displayed to do peak selection.

homo2d Execute protocol actions of apptype homo2d (M)

Applicability: Liquids

Description: Perform the actions for Heteronuclear 2D protocols to set up, process, and plot

experiments.

Examples: homo2d('setup') execute homo2d experimental setup

homo2d('process') execute homo2d processing homo2d('plot') execute homo2d plotting

See also: NMR Spectroscopy User Guide, VnmrJ Walkup

Related: apptype Application type (P)

execpars Set up the exec parameters (M)

homorof1 Delay before turning on homo decoupling rf (P)

Applicability: VNMRS liquids systems

Description: Optional parameter for delay before turning on homonuclear decoupling after

gating the receiver off. The amplifier is un-blanked and T/R switch set to transmit mode during homorof1 delay (in  $\mu$ sec. units). A default delay of 2  $\mu$ sec.

is used if the parameter does not exist.

Values: 2 to 5 µsec. are typical.

Related: dutyc The rf duty cycle fraction for homonuclear decoupling (P)

hdmf Modulation frequency for the band selective homonuclear decoupling

(P)

hdof Frequency offset for homodecoupling (P)

hdpwr Sets the rf attenuator to control the power for homonuclear decoupling

(P)

hdpwrf Sets the rf linear modulator fine power for homonuclear decoupling (P)

hdseq Sets the decoupler waveform filename (P)

hdres Sets the tip angle resolution (P)

homo Homodecoupling control for observe channel (P)

homorof2 Delay after blanking the amplifier and setting T/R switch to receive (P) homorof3 Delay between setting T/R switch to receive gating on the receiver (P)

tn Nucleus for observe transmitter (P)

# homorof2 Delay after blanking the amp and setting T/R switch to recv (P)

Applicability: VNMRS liquids systems

Description: Optional parameter for delay after the transmitter is gated off, the amplifier is

blanked, and before the T/R switch is set to receive. A default delay of 2 µsec.

is used if the parameter does not exist.

Values: 2 to 5 μsec. are typical.

Related: dutyc The rf duty cycle fraction for homonuclear decoupling (P)

hdmf Modulation frequency for the band selective homonuclear decoupling (P)

hdof Frequency offset for homodecoupling (P)

hdpwr Sets the rf attenuator to control the power for homonuclear decoupling (P) hdpwrf Sets the rf linear modulator fine power for homonuclear decoupling (P)

hdseq Sets the decoupler waveform filename (P)

hdres Sets the tip angle resolution (P)

homo Homodecoupling control for observe channel (P) homorof1 Delay before turning on homo decoupling rf (P)

homorof3 Delay between setting T/R switch to receive gating on the receiver (P)

tn Nucleus for observe transmitter (P)

# homorof3 Delay between setting T/R to receive and gating the recvr on (P)

Applicability: VNMRS liquids systems

Description: Optional parameter for delay after the T/R switch is set to receive and before the

receiver gate is gated on. A default delay of 2 µsec. is used if the parameter does

not exist.

Values: 2 to 5  $\mu$ sec. are typical

Related: dutyc The rf duty cycle fraction for homonuclear decoupling (P)

hdmf Modulation frequency for the band selective homonuclear decoupling (P)

hdof Frequency offset for homodecoupling (P)

hdpwr Sets the rf attenuator to control the power for homonuclear decoupling (P)
hdpwrf Sets the rf linear modulator fine power for homonuclear decoupling (P)

hdseq Sets the decoupler waveform filename (P)

hdres Sets the tip angle resolution (P)

homo Homodecoupling control for observe channel (P) homorof1 Delay before turning on homo decoupling rf (P)

homorof2 Delay after blanking the amplifier and setting T/R switch to receive (P)

Nucleus for observe transmitter (P)

hoult Set parameters alfa and rof2 according to Hoult (M)

Description: Sets the values of alfa and rof2 according to a prescription advanced by D.

I. Hoult (*J. Magn. Reson.* **51**, 110 (1983)). These parameters set the times that follow the final pulse, which can be important where the flatness of the baseline

is of concern.

See also: NMR Spectroscopy User Guide

Related: alfa Set alfa delay before acquisition (P)

calfa Recalculate alfa so that first-order phase is zero (M)

rof2 Receiver gating time following pulse (P)

hpa Plot parameters on special preprinted chart paper (C)

Description: Plots a predetermined list of parameters by "filling in the blanks" at the bottom

of the preprinted chart paper available for Hewlett-Packard 7475- and 7550-

series plotters.

See also: NMR Spectroscopy User Guide

Related: apa Plot parameters automatically (M)

X-zero position of HP plotter or Postscript device (P)
Y-zero position of HP plotter or Postscript device (P)

Hprescan Proton prescan (P))

Applicability: VnmrJ Walkup

Description: This parameter is used to keep track of the type and status of the Proton prescan.

It is used for Proton, Presat, Wet1d, and Minsw protocols.

See also: VnmrJ Walkup

Related: xmHprescan Set up and process Proton prescans (M)

hregions Select integral regions in proton spectrum (M)

Description: Selects integral regions, a critical step in automatic processing of proton spectra.

It is critical not only because of aesthetic reasons (some people like many small integrals, others prefer a few large regions), but also because other commands, such as bc, depend on the correct integration: bc can either fail or it can make broad, unintegrated lines disappear from the spectrum. hregions was specifically designed for proton spectra and should not be used for other types

of spectra. The result of hregions also depends on the lineshape and the  $\,$ 

signal-to-noise ratio of a spectrum

See also: NMR Spectroscopy User Guide

Related: bc 1D and 2D baseline correction (C)

integrate Automatically integrate 1D spectrum (M)

hs Homospoil pulses (P)

Description: Turns on homospoil pulses at various times in different pulse sequences.

Homospoil is a process by which the homogeneity is temporarily made very bad ("spoiled") to cause any transverse magnetizations present at that time to decay

rapidly to zero. hst controls the length of any homospoil pulse.

Values: In a standard two-pulse sequence, homospoil pulses can be inserted during

periods A and B (delays d1 and d2): hs='yn' gives a homospoil pulse at the beginning of d1, hs='ny' gives a pulse during d2, and hs='yy' gives

homospoil pulses during both d1 and d2. The desired value is generally

hs='nn'.

See also: NMR Spectroscopy User Guide
Related: d1 First delay (P)

d2 Incremented delay in 1st indirectly detected dimension (P)

hst Homospoil time (P)

Hsqc Convert the parameter to a HSQC experiment (M)

Description: Convert the parameter to a HSQC experiment.

HSQC15 Set up parameters for <sup>15</sup>N HSQC experiment (M)

Description: Converts the current parameter set to a HSQC experiment for <sup>15</sup>N.

HSQC d2 Set up parameters for <sup>15</sup>N HSQC experiment using dec. 2 (M)

Description: Converts the current parameter set to a HSQC experiment for <sup>15</sup>N with

decoupler 2 as <sup>15</sup>N.

HSQC d213 Set up parameters for <sup>13</sup>C HSQC experiment using dec. 2 (M)

Description: Converts the current parameter set to a HSQC experiment for <sup>13</sup>C with

decoupler 2 as <sup>13</sup>C.

hsqcHT Set up the hsqcHT experiment (M)

Description: Sets up parameters for a Hadamard-encoded hsqc experiment.

See also: NMR Spectroscopy User Guide

htfrq1 Hadamard offset in ni (P)

fn1 Fourier number in 1st indirectly detected dimension (P)

ni Number of increments in 1st indirectly detected dimension (P)

ft2d Fourier transform 2D data (C)

sethtfrq1 Set a Hadamard frequency list from a line list (M)
Hsqc Set up parameters for HSQC experiment (M)

Hsqctoxy Convert parameters to a HSQCTOXY experiment (M)

Description: Convert the parameter to a HSQCTOXY experiment.

HSQCTOXY15 Set up parameters for <sup>15</sup>N HSQCTOXY experiment (M)

Description: Converts the current parameter set to a HSQCTOXY experiment for <sup>15</sup>N.

HSQCTOXY d2 Set up parameters for <sup>15</sup>N HSQCTOXY using decoupler 2 (M)

Description: Converts the current parameter set to a HSQCTOXY experiment for <sup>15</sup>N with

decoupler 2 as <sup>15</sup>N.

HSQCTOXY d213 Set up parameters for <sup>13</sup>C HSQCTOXY using decoupler 2 (M)

Description: Converts the current parameter set to a HSQCTOXY experiment for <sup>13</sup>C with

decoupler 2 as <sup>13</sup>C.

hsgctoxySE Set up parameters for HSQC-TOCSY 3D pulse sequence (M)

Description: Sets up parameters for a HSQC -TOCSY 3D experiment.

hsrotor Display rotor speed for solids operation (P)

Applicability: Systems equipped with the rotor synchronization module.

Description: Controls display of rotor speed. Depending on whether the rotor

synchronization module is present (set by the Rotor Synchronization label in the Spectrometer Configuration window, parameter rotorsync is set to 1 or 0. The xpolar1 macro in turn uses this to create hsrotor, which is set to 'y' if rotor synchronization is present. If the parameter srate exists, it is updated to the spin speed of the rotor at the end of the experiment. The interlock function specified by parameter in also changes. If hsrotor='y' and in='y', the

experiment is terminated if rotor speed deviates more than 100 Hz.

hst Homospoil time (P)

Description: Controls pulse length if homospoil is activated by the hs parameter.

Values: 0 to 20 ms (limited by hardware).

Values: 'n' makes srate unmodified by acquisition and turns off the rotor speed

display in Acqstat.

'y' makes the hardware information from the rotor synchronization board update srate and displays the rotor speed in the Acqstat status display.

See also: User Guide: Solid-State NMR

Related: Acqstat Bring up the acquisition status display (U)

config Display current configuration and possibly change it (M)

in Interlock (P)

rotorsync Rotor synchronization (P)
srate Spinning speed (P)

xpolar1 Set up parameters for XPOLAR1 pulse sequence (M)

htbitrev Hadamard bit reversal flag (P)

Description: A flag to enable or disable bit reversal of the Hadamard matrix. The flag should

be the same for both acquisition and processing for the Hadamard transform to

be successful.

Values: 'y' enable Hadamard bit reversal

'n' disable Hadamard bit reversal

Default value is 'n'.

See also: NMR Spectroscopy User Guide

Related: htfrq1 Hadamard frequency list in ni (P)

htbw1 Hadamard pulse excitation bandwidth in ni (P)

Description: The excitation bandwidth used to generate the frequencies contained in the

shaped pulses used by the Hadamard matrix. If a single value is specified, the same bandwidth is used for all frequencies. If the parameter is arrayed, the bandwidth array element is used by the corresponding array element in htfrq1.

Values: Default value is 20.0 if the parameter does not exist.

See also: NMR Spectroscopy User Guide

Related: <a href="https://http

Number of increments in 1st indirectly detected dimension (P)

htcal1 RF calibration flag for Hadamard waveforms in ni (P)

Description: A flag to allow power optimization of Hadamard waveforms in the 1st indirect

dimension.

Values: 0 power optimization using htpwr1 is disallowed

>0 power optimization using htpwr1 is allowed

Default value is 0.

See also: NMR Spectroscopy User Guide

Related: <a href="https://http

htpwrl Power level for rf calibration of Hadamard waveforms in ni (P)

Number of increments in 1st indirectly detected dimension (P)

htfrq1 Hadamard frequency list in ni (P)

Description: A list of frequencies used in Hadamard spectroscopy, used for creating the

Hadamard pulse shapes, and for placing the transformed traces at the correct

frequencies in the indirect dimension.

Values: Typical values are an arrayed set of frequencies between -sw1/2 and sw1/2.

See also: NMR Spectroscopy User Guide

Related: htofs1 Hadamard offset in ni (P)

fn1 Fourier number in 1st indirectly detected dimension (P)

ni Number of increments in 1st indirectly detected dimension (P)

sethtfrq1 Set Hadamard frequency list from a line list (M)

proc1 Type of processing on ni interferogram (P)

Spectral width in 1st indirectly detected dimension (P)

htofs1 Hadamard offset in ni (P)

Description: The number of array elements to skip in ni when doing the Hadamard transform.

The first element of the Hadamard matrix typically has all positive values (++++), and is usually not useful in constructing the Hadamard data.

Values: Default value is 0. Typical values are 1 or 2.

See also: NMR Spectroscopy User Guide

Related: htfrq1 Hadamard frequency list in ni (P)

fn1 Fourier number in 1st indirectly detected dimension (P)

ni Number of increments in 1st indirectly detected dimension (P)

ft2d Fourier transform 2D data (C)

Type of processing on ni interferogram (P)

htpwr1 Power level for RF calibration of Hadamard waveforms in ni (P)

Description: Power level for optimizing Hadamard waveforms in the 1st indirect dimension.

Values: -16 to 63 dB in steps of 1 dB. See also: *NMR Spectroscopy User Guide* 

Related: <a href="https://http

htcall RF calibration flag for Hadamard waveforms in ni (P)

ni Number of increments in 1st indirectly detected dimension (P)

htss1 Stepsize for Hadamard waveforms in ni (P)

Description: Sets the stepsize during Hadamard waveform creation. Typically, this parameter

is not needed, and a default stepsize is used.

Values: Does not exist - default stepsize is used.

0 default stepsize is used.>0 stepsize in microseconds.

See also: NMR Spectroscopy User Guide

Related: <a href="https://http

Number of increments in 1st indirectly detected dimension (P)

#### hzmm Scaling factor for plots (P)

Description: Contains the quotient of wp divided by wc, a scaling factor useful for plotting.

hzmm applies to 1D only.

See also: NMR Spectroscopy User Guide

Related: wc Width of chart (P)

wp Width of plot (P)

# hz tomm Convert locations from Hz or ppm to plotter units (C)

Syntax: (1) hztomm(x position)<:xmm>

(2) hztomm(x\_position, y\_position) <:xmm, ymm>

(3) hztomm(<'box', ><'plotter'|'graphics', >x\_left, x\_right, y\_bottom, y\_top) <: x1mm, x2mm, y1mm, y2mm>

Description: Converts locations from Hz, or ppm, to plotter units.

Arguments: x\_position in syntax 1 is a location along the 1D axis, in Hz or ppm, to be

converted to plotter units using the current values of parameters sp and wp. Plotter units are mm on most plots and are scaled for graphics display. For ppm entries, use the p suffix following numerical values (see first example below).

x\_position, y\_position in syntax 2 is a coordinate, in Hz or ppm, on a 2D plot to be converted to plotter units, using the parameters sp and wp to convert the horizontal position and the parameters sp1 and wp1 to convert the vertical position.

x\_left,x\_right,y\_bottom,y\_top in syntax 3 are box edges, in Hz or ppm, on a 2D plot to be converted to plotter units, using the parameters sp and wp to convert the left and right edges, and parameters sp1 and wp1 to convert the top and bottom edges.

'box' is a keyword to draw a box and to make the first two return arguments, if supplied, give the location of the upper left corner of the box, in plotter units.

'plotter' is a keyword to select the plotter. The default is 'graphics'.

'graphics' is a keyword to select the graphics screen. This is the default.

x1mm, x2mm, y1mm, y2mm are return arguments giving values in plotter units. If return arguments are not supplied, the results are displayed instead.

Examples: hztomm(20p)

hztomm(xpos, ypos):xmm,ymm

hztomm('box','plotter',20,50,10,30)

See also: NMR Spectroscopy User Guide

Related: box Draw a box on a plotter or graphics display (C)

Start of plot in directly detected dimension (P)

Start of plot in 1st indirectly detected dimension (P)

Н

wp Width of plot in directly detected dimension (P)
wp1 Width of plot in 1st indirectly detected dimension (P)

i Insert sample (M)

ihwinfo Hardware status of console (U)

il Interleave arrayed and 2D experiments (P)
ilfid Interleave FIDs during data processing (C)

imagefile Display an image file (M)

imageprint Plot non interactive gray scale image (M)

imconi Display 2D data in interactive grayscale mode (M)

in Lock and spin interlock (P)

inadqt Set up parameters for INADEQUATE pulse sequence (M)

index2 Projection or 3D plane index selected (P)

input Receive input from keyboard (C)
ins Integral normalization scale (P)

ins2 2D volume value (P)

insref Fourier number scaled value of an integral (P)
ins2ref Fourier number scaled volume of a peak (P)

insert Insert sample (M)

inset Display an inset spectrum (C)

integ
integrate
Find largest integral in a specified region (C)
Automatically integrate 1D spectrum (M)

intmod Integral display mode (P)

intvast Produces a text file of integral regions (M)

io Integral offset (P)
is Integral scale (P)

**isadj** Automatic integral scale adjustment (M)

isadj 2 Automatic integral scale adjustment by powers of two (M)

isrealUtility macro to determine a parameter type (M)isstringUtility macro to determine a parameter type (M)

iterate Parameters to be iterated (P)

# i Insert sample (M)

Description: Turns off the eject air, waits for sample to slowly drop, and then turns off the

slow drop air. The macro insert functions the same as i.

See also: NMR Spectroscopy User Guide

Related: e Eject sample (M)

eject Eject sample (M)
insert Insert sample (M)

#### ihwinfo Hardware status of console (U)

Syntax: (From UNIX) ihwinfo('startup'|'abort')

Description: Displays status of digital hardware in the console. The output is intended for

service personnel and probably not meaningful to users.

Arguments: 'startup' is a keyword to display the status at the conclusion of the last

console startup (powerup, reboot, etc.).

'abort' is a keyword to display the status the last time an acquisition was aborted or the console rebooted from the host computer (abortallacgs). In this context, exiting from either the FID display or lock display of acqi counts

as an abort. Only the status from the last abort can be displayed.

Examples: ihwinfo('startup')

ihwinfo('abort')

See also: NMR Spectroscopy User Guide

Reset acquisition computer in a drastic situation (C) Related: abortallacgs

> showconsole Show console configuration parameters (U)

#### Interleave arrayed and 2D experiments (P) il

Applicability: Interleaving is not currently supported for the VNMRS or MR400 system.

#### ilfid Interleave FIDs during data processing (C)

Description: Converts a multiple FID element into a single FID. It is possible to effectively

extend the Nyquist frequency (i.e., increase the effective spectral width sw) by acquiring a number of FIDs with different tau2 values and then reprocessing the data. ilfid does the necessary processing of time-domain data to achieve this extension, assuming that a pulse sequence (not supplied) has been written to

generate the required data.

When invoked in an experiment of nf FIDs, each of np points, ilfid sorts the data into a single FID of np\*nf points that can then be transformed. The interleaving takes the first complex point of each of the nf FIDs and places them in sequential order in the new FID. It then takes the second complex point from each of the nf FIDs and appends them sequentially to the new FID. This operation is repeated for all complex points. Although ilfid adjusts np and nf, it does not alter other parameters such as sw.

CAUTION: Because ilfidalters the data irrevocably, it is strongly recommended

that you save the FID before using ilfid.

Examples: Illustrated below is the interleaving of an FID with nf=3 and np=4. Each point

> is represented by two digits. The first digit is the nf number and the second digit is the sequential point for that nf value. Data before the ilfid command:

11, 12, 13, 14; 21, 22, 23, 24; 31, 32, 33, 34

Data after the ilfid command:

11, 21, 31, 12, 22, 32, 13, 23, 33, 14, 24, 34

See also: NMR Spectroscopy User Guide

Related: Number of FIDs (P)

> np Number of data points (P)

Spectral width in directly detected dimension (P) SW

#### imagefile Display an image file (M)

Applicability: Imaging

> imagefile('output option','imagefile'<,x,y,w,h,'mol'>) Syntax:

Description: Display or plot an imagefile at default location and size or, optionally, at

location and size specified by: x (x-position), y (y-position), w (width), h

ı

(height), and mol if it is an image file of a molecular structure. Display all, plot all, or clear all images for the current experiment.

Arguments: output option choices are:

clear, clear all images for the current experiment

display, display imagefile

displayall, displays all images for the current experiment

plot, plot imagefile

plotall, plot all images for the current experiment

imagefile, name of image file to display or plot

x, x position y, y position w, width h, height

mol molecular structure image file

Examples: imagefile('clear') clear all images for the current experiment.

imagefile ('displayall') display all images for the current

experiment.

#### imageprint Plot non interactive gray scale image (M)

Description: Sends to the plotter a dcon color intensity map with linear instead of

logarithmic increments and with grayscale instead of colors.

See also: NMR Spectroscopy User Guide

Related: dcon Display noninteractive color intensity map (C)

#### imconi Display 2D data in interactive grayscale mode (M)

Description: Calls the dconi program with the arguments required for grayscale image

display: dconi ('dcon', 'gray', 'linear').

#### in Lock and spin interlock (P)

Description: Controls error handling based on lock level and spin speed, and specifies action

based on lock level failure or spinner failure. The action can be to generate an error and halt acquisition, or to generate a warning and continue acquisition.

Values: Can be set to one or two characters:

- If set to two characters, the first character specifies the action for lock failure and the second character specifies the action for spinner failure.
- If set to only one character, that character specifies the same action for either lock or spinner failure.

'n' stops any system checking so that acquisition continues regardless of the lock level or spin speed.

'w' makes the system check the lock level and the spin speed. A warning message is added to the log file if the lock level falls below a preset hardware level (about 20 on the lock meter) or if spin is set to a particular value and the spin speed goes out of regulation; however, acquisition is not stopped.

'y' makes the system check the lock level and spin speed. Acquisition is halted if the lock level falls below a preset hardware level (about 20 on the lock meter) or if spin is set to a particular value and the spin speed goes out of regulation.

See also: NMR Spectroscopy User Guide

Related: spin Sample spin rate (P)

inadgt Set up parameters for INADEQUATE pulse sequence (M)

Description: Sets up parameters for 2D INADEQUATE (Incredible Natural Abundance

Double-Quantum Transfer Experiment).

See also: NMR Spectroscopy User Guide

Related: foldcc Fold INADEQUATE data about 2-quantum axis (C)

index2 Projection or 3D plane index selected (P)

Description: Stores whether a projection or 3D plane index is selected. It shows the current

status only and cannot be used to select a plane or projection. This parameter is

also displayed in the Status window below "Index."

Values: 0 indicates a projection is selected.

1 to the half the Fourier number of the normal axis indicates a 3D plane is

selected; the number is the index of the 3D plane.

See also: NMR Spectroscopy User Guide

Related: dplane Display a 3D plane (M)

dproj Display a 3D plane projection (M)
nextpl Display the next 3D plane (M)
prevpl Display the previous 3D plane (M)

select Select a spectrum or 2D plane without displaying it (C)

inept Set up parameters for INEPT pulse sequence (M)

Description: Sets up parameters for the INEPT (Insensitive Nuclei Enhanced by Polarization

Transfer) experiment.

See also: NMR Spectroscopy User Guide

Related: ppcal Proton decoupler pulse calibration (M)

initialize iterateSet iterate string to contain relevant parameters (M)

Description: Takes the current spin system (contained in spinsys) and derives from it

relevant parameters. This can be used to control which parameters are iterated during a spin simulation iteration (e.g., for an ABC spin system, iterate is

set to 'A, JAB, JAC, B, JBC, C').

See also: NMR Spectroscopy User Guide

Related: iterate Parameters to be iterated (P)

input Receive input from keyboard (C)

Syntax: input<(<pre>oprompt><,delimiter>)>:var1,var2,...

Description: Receives fields of characters from the keyboard and stores them into one or

more variables.

Arguments: prompt is a string displayed on the command line.

delimiter is a character separating input fields. The default is a comma.

var1, var2, . . . are return values. input stores the values into as many of

these arguments as given and ignores the rest of the input line.

Examples: input:\$b

input('Enter pulse width:'):pw
input('x and y coordinates'):cr,cr1

input('Enter lastname:firstname',':'):\$last,\$first

See also: User Programming

Related: string Create a string variable (C)

#### ins Integral normalization scale (P)

Description: Sets the integral value, independent of is and vs. Reported integral values are

scaled by fn; that is, the reported integral of a given region is independent of fn. The insref parameter is also used to determine a reference integral value.

The setint macro sets integral value.

See also: NMR Spectroscopy User Guide

Related: dlni Display list of normalized integrals (M)

fn Fourier number in directly detected dimension (P)

is Integral scale (P)

insref Fourier number scaled value of an integral (P)
mark Determine intensity of spectrum at a point (C)

setint Set value of an integral (M)

Vertical scale (P)

#### ins2 2D volume value (P)

Description: Adjusts the 2D volume value, independent of is and vs. The volume is scaled

by Fourier numbers for the two dimensions.

See also: NMR Spectroscopy User Guide

Related: is Integral scale (P

Fourier number scaled volume of a peak (P)

Automatic and interactive 2D peak peaking (C)

VS Vertical scale (P)

# insref Fourier number scaled value of an integral (P)

Description: Set to the Fourier number scaled value of a selected integral. The reported

integral values will be (<code>integral value</code>)\*ins/insref/fn. If insref is "not used", the sum of all integrals will be <code>ins</code>. The "not used" mode is the equivalent of the normalized integral mode. If <code>insref</code> is zero or not defined,

the reported integrals will be (integral value)\*ins/fn.

See also: NMR Spectroscopy User Guide

Related: fn Fourier number in directly detected dimension (P)

ins Integral normalization scale (P)

liamp Amplitudes of integral reset points (P)

setint Set value of an integral (M)

# ins2ref Fourier number scaled volume of a peak (P)

Description: Set to the Fourier number scaled volume of the selected peak. The reported

volume is *volume*\*ins2/ins2ref/fn/fn1. If ins2ref is "not used", sum of all volumes is ins2. The "not used" mode is equivalent to a normalized volume mode. If ins2ref is zero or not defined, the reported volume is

volume\*ins2/fn/fn1.

See also: NMR Spectroscopy User Guide

Related: fn Fourier number in directly detected dimension (P)

fn1 Fourier number in first indirectly detected dimension (P)

ins2 2D volume value (P)

112d Automation and interactive 2D peak picking (C)

#### insert Insert sample (M)

Description: Turns off the eject air, waits for the sample to slowly drop, and then turns off the

slow drop air. The macro i is identical in function to insert.

See also: NMR Spectroscopy User Guide

Related: e Eject sample (M)

eject Eject sample (M)
i Insert sample (M)

# inset Display an inset spectrum (C)

Description: Displays the part of the spectrum between the two cursors as an inset. Before

entering inset, run the ds command and display two cursors. The vertical position is shifted up about one-quarter of the height of the whole display canvas. The old spectrum remains on the screen, but the parameters shown at the bottom are relevant to the new display. If present, the integral trace is duplicated. The scale is also duplicated if it is present. After running inset, you can shift the displayed spectrum, expand it, or even contract it with the left

and right mouse buttons.

See also: NMR Spectroscopy User Guide

Related: ds Display a spectrum FID (C)

# integ Find largest integral in a specified region (C)

Syntax: integ<(highfield,lowfield)><:size,value>

Description: Finds the largest absolute-value integral in the specified region, or the total

integral if no reset points are present between the specified limits.

Arguments: highfield and lowfield are the limits of the region. The default values

are the parameters sp and sp+wp, respectively.

size is a return value with the size of the largest integral. The size depends on

the value of the parameter is and can be positive or negative.

value is a return argument with the value of the largest integral. This value

depends on ins, insref, and fn, and is independent of is.

Examples: inteq:r1,r2

integ(500,1000):\$height

integ(100+sp,300+sp):\$ht,\$val

See also: User Programming

Related: fn Fourier number in directly detected dimension (P)

ins Integral normalization scale (P)

insref Fourier number scaled value of an integral (P)

is Integral scale (P)

zero-order phase in directly detected dimension (P)
Start of plot in directly detected dimension (P)
Wp Width of plot in directly detected dimension (P)

# integrate Automatically integrate 1D spectrum (M)

Description: A universal macro for selecting integral regions and adjusting the integrals in

size and offset. Only if regions are not already selected, and if <code>intmod</code> is set to 'partial', will integrate call region to select integral regions. For proton spectra, the selection is done through the hregions macro; for <sup>19</sup>F and <sup>31</sup>P spectra (for wide spectral windows, multiplet spectra), region is called with optimized arguments, and for other nuclei (mostly decoupled, single-line spectra) other optimized parameters are used with region, such that lines consisting of a few data points only are recognized.

See also: NMR Spectroscopy User Guide

Related: <a href="hregions">hregions</a> Select integral regions in proton spectrum (M)

region Automatically select integral regions (C)

# intmod Integral display mode (P)

Description: Controls display and plotting of the spectral integral.

Values: 'off' indicates that no integrals are displayed or plotted.

'full' indicates that all integral regions are displayed or plotted.

 $\verb"partial" indicates that every other integral region is plotted (typically used$ 

to display integrals of only peaks and not of the baseline region).

See also: NMR Spectroscopy User Guide

Related: plc Plot carbon spectrum (M)

plh Plot proton spectrum (M)
plp Plot phosphorus spectrum (M)

# intvast Produces a text file of integral regions (M)

Applicability: Systems with VAST accessory.

Syntax: intvast (last)

Description: intvast produces a text file, integ.out in the current experiment, containing

the integrals of the partial regions of each spectra from wells 0 to last.

Arguments: last is the number last sample well. The default is 96.

See also: NMR Spectroscopy User Guide

Related: pintvast Plot the integrals (M)

#### io Integral offset (P)

Description: Offset of the integral with respect to the spectrum.

Values: 0 to 200, in mm.

See also: NMR Spectroscopy User Guide

### is Integral scale (P)

Description: Multiplier that adjusts height of the displayed integral trace. Note that the ins

parameter controls integral value, and that is has no effect on integral value.

Values: 1 to 1e9

See also: NMR Spectroscopy User Guide

Related: ins Integral normalization scale (P)

ins2 2D volume value (P)

insref Fourier number scaled value of an integral (P) integ Find largest integral in a specified region (C)

#### isadj Automatic integral scale adjustment (M)

Syntax: isadj<(height<,neg height>)>

Description: Adjusts the height of the integrals in a display to make the tallest integral fit the

paper. Optionally, the height of the maximum integral can be specified by an argument. Negative integrals, if present, are given a limit of 10 mm if parameter io is less than 10; otherwise, they are set so they end 5 mm above the spectrum. Negative integrals can also be given a height. Whichever part of the integrals (positive or negative) runs into the given limit will be used to scale is.

Arguments: height is the size, in mm, of the maximum integral on display. The default is

the height that makes the tallest integral fit the paper.

neg\_height is the desired height, in mm, of the largest negative integral. If io is less than 10, the default is 10; otherwise, the default height is 5 mm above

the spectrum.

Examples: isadj

isadj(100)
isadj(100,100)

See also: NMR Spectroscopy User Guide

Related: io Integral offset (P)

is Integral scale (P)

isadj2 Automatic integral scale adjustment by powers of two (M)

# isadj2 Automatic integral scale adjustment by powers of two (M)

Syntax: isadj2<(height<,neg height>)>:scaling factor

Description: Functionally the same as isadj except that isadj 2 adjusts the integral height

by powers of two and returns the scaling factor to the calling macro.

Arguments: height is the size, in mm, of the maximum integral on display.

neg height is the desired height, in mm, of the maximum negative integral

on display.

scaling factor is a return value giving the ratio of the new integral size

to the old value (new is/old is).

Examples: isadj2

isadj2(100) isadj2(100,100) isadj2(50):r1

See also: NMR Spectroscopy User Guide
Related: is Integral scale (P)

Automatic integral scale adjustment (M)

#### isreal Utility macro to determine a parameter type (M)

Syntax: isreal(paramname<, tree>)

Description: Returns 1 if and only if paramname is a real type. It returns 0 if paramname is

a string type. If there is an error, the error is reported and the macro also returns

0. The value of tree is 'current', 'global', 'processed' or 'systemglobal' and the default is 'current'.

There is some unfortunate ambiguity and vagueness in regard to vnmr parameters and their types. The meaning of real and string vary slightly depending upon context. There are seven types altogether. The macro gettype returns a unique integer value when operating on the parameter. Of the seven types, two can be broadly categorized as string, and five can be broadly categorized as real. Since one of the string category types is 'string' and one of the real category types is 'real', this is where the ambiguity arises. The return values for gettype are:

category	type	gettype returns
string	'string'	2
	'flag'	4
real	'real'	1
	'delay'	3
	'frequency'	5
	'pulse'	6
	'integer'	7

The isreal function returns 0 for the string category and 1 for the real category. This function is consistent with the typeof () operator. The typeof () operator is primarily intended to ascertain the type of the input argument to a macro, so using it for other purposes is not recommended. Also, it does not take a tree argument. Note that typeof () returns 0 for reals and 1 for strings, the opposite of this macro, but it should be clear from the name what is intended. A sister macro isstring returns the same value as typeof().

Related: isstring Utility macro to determine a parameter type (M) typeof

Utility macro to determine a parameter type (M) isstring

Syntax: isstring(paramname<, tree>)

Returns 1 if and only if paramname is a string type. It returns 0 if paramname is a real type. If there is an error, the error is reported and the macro also returns 0. The value of tree is 'current', 'global', 'processed' or 'systemglobal' and the default is 'current'.

Return identifier for argument type (O)

There is some unfortunate ambiguity and vagueness in regard to vnmr parameters and their types. The meaning of real and string vary slightly depending upon context. There are seven types altogether. The macro gettype returns a unique integer value when operating on the parameter. Of the seven types, two can be broadly categorized as string, and five can be broadly categorized as real. Since one of the string category types is 'string' and one of the real category types is 'real', this is where the ambiguity arises. The return values for gettype are:

category	type	gettype returns
string	'string'	2
	'flag'	4
real	'real'	1
	'delay'	3

Description:

category type gettype returns

'frequency' 5
'pulse' 6
'integer' 7

The function isstring returns 0 for the real category and 1 for the string category. This function is consistent with the typeof() operator. The typeof() operator is primarily intended to ascertain the type of the input argument to a macro, so using it for other purposes is not recommended. Also, it does not take a tree argument. Note that typeof() returns 0 for reals and 1 for strings, the opposite of this macro, but it should be clear from the name what is intended. A sister macro isstring returns the same value as typeof().

Related: isreal Utility macro to determine a parameter type (M)

typeof Return identifier for argument type (O)

# iterate Parameters to be iterated (P)

Description: Contains parameters to be iterated during iterative spin simulations. If the Set

Params button is used in setting up spin simulation parameters, iterate is initialized to a string containing all parameters appropriate to the current spin

system.

Values: List of parameters, separated by commas (e.g., iterate='A,B,JAB').

See also: NMR Spectroscopy User Guide

Related: initialize\_iterate Set iterate string to contain relevant parameters (M)

# J

jcurwin Work space numbers of all viewports (P) Start Plot Designer Program (M) jdesign Join existing experiment (C) jexp jexp1-jexp9999 Join existing experiment and display new parameters (M) jplot Plot from Plot Designer program (C) jplotscale Scale plot parameters (M) *j*plotunscale Restore current experiment parameters (M) jprint Prints the selected images to a printer or file (M) Set up parameters for JUMPRET pulse sequence (M) jumpret jviewport Work space numbers of the current viewports (P) Work space labels for all viewport buttons (P) jviewportlabel jviewports Viewport layout (P)

jcurwin Work space numbers of all viewports (P)

Description: An arrayed global parameter, set to the work space numbers used by all

viewports.

jwin

See also: NMR Spectroscopy User Guide, VnmrJ Walkup

Related: curwin Current window (P)

jviewport Work space numbers of the current viewports (P)
jviewportlabel Work space labels for all viewport buttons (P)

Activate and record activity in current window (M)

jdesign Start Plot Designer Program (M)

Syntax: jdesign

Description: Opens the Plot Designer program, which provides mechanisms for positioning

spectra, parameters, axes, and other plot output on a page. Text annotation and

drawing features are available.

See also: NMR Spectroscopy User Guide

Related: jplot Plot from Plot Designer program (C)

jexp Join existing experiment (C)

Syntax: (1) jexp(exp\_number)

(2) jexp:\$current\_exp\_number,\$current\_exp\_name

Description: Joins an existing experiment (syntax 1) or returns the current experiment

number and experiment name (syntax 2). After entering this command, until another "join experiment" command or macro is entered, all actions (including changes of parameters, acquisition of data, and display of data) apply to the

parameters and data of the experiment joined.

The jexp command does not refresh the display or display new experiment parameters. Use one of the macros jexp1, jexp2, etc. to join an experiment

and have the screen refreshed and new parameters displayed.

Arguments: exp number is a number from 1 to 9999 for existing experiment to be joined.

\$current exp number is a return value with the current experiment

number.

\$current exp name is a return value with the current experiment name.

Examples: jexp(3)

jexp:\$expp
jexp:r1,n1

See also: NMR Spectroscopy User Guide; VnmrJ Walkup

Related:  $\begin{array}{c} \text{cexp} \end{array}$  Create an experiment (M)

delexp Delete an experiment (M)

jexp1-jexp9 Join existing experiment and display new parameters (M)

unlock Remove inactive lock and join experiment (C)

# jexp1-jexp9999Join existing experiment and display new parameters (M)

Syntax: jexp1, jexp2, jexp3, ...,jexp9999

Description: Joins an existing experiment, refreshes the screen, and displays the main menu

and the new experiment parameters. After entering this macro, until another "join experiment" command or macro is entered, all actions (including changes of parameters, acquisition of data, and display of data) apply to the parameters

and data of the experiment joined.

To join an experiment without refreshing the screen and displaying new

parameters, use the jexp command.

Examples: jexp8

jexp354

See also: NMR Spectroscopy User Guide

Related: cexp Create an experiment (M)

delexpDelete an experiment (M)jexpJoin existing experiment (C)

unlock Remove inactive lock and join experiment (C)

# jplot Plot from Plot Designer program (C)

Syntax: jplot<(<'-setup'><,template)>

Description: Starts plotting from the Plot Designer program to the current plotter.

Arguments: '-setup' is a keyword to start jdesign, the Plot Designer program, to

allow interactive design and plotting.

template is the name of a file that will be used to make a plot of the current

experiment. The default is a saved file chosen by the user.

Examples: jplot

jplot('t1')

See also: NMR Spectroscopy User Guide

Related: jdesign Start Plot Designer program (M)

jplotscale Scale plot parameters (M)

jplotunscale Restore current experiment parameters (M)

# jplotscale Scale plot parameters (M)

Applicability: Plot Designer program

Description: Scales parameters of plotting area and an imported plot. When a region is drawn

in Plot Designer, jplotscale automatically changes the plotting area parameters wcmax and wc2max. The parameters io, is, vs, wc, and wc2 of a plot imported into a region are adjusted according to wcmax and wc2max.

See also: NMR Spectroscopy User Guide

Related: jplot Plot from Plot Designer program (C)

jplotunscale Restore current experiment parameters (M)

# jplotunscale Restore current experiment parameters (M)

Applicability: Plot Designer program

Description: Restores the current experiment parameters (io, is, vs, wc, and wc2) to a plot

within a region that was created in Plot Designer. For example, entering jplotunscale jexp2 jplotscale restores the parameters of

experiment 2 to a plot and then jplotscale applies the adjusted parameters

to the plot.

See also: NMR Spectroscopy User Guide

Related: jplot Plot from Plot Designer program (C)

#### jprint Prints the selected images to a printer or file (M)

Description: The jprint macro takes the value of the parameters printregion,

printsend, printfile, printlayout, printformat, printsize.

# jumpret Set up parameters for JUMPRET pulse sequence (M)

Description: Sets up parameters for a jump-and-return water suppression sequence.

See also: NMR Spectroscopy User Guide

# jviewport Work space numbers of the current viewports (P)

Description: A global parameter, set to the work space number that the current viewport is

joined to. The parameter is set when the viewport starts. Each viewport may be

joined to a different work space.

See also: NMR Spectroscopy User Guide, VnmrJ Walkup

Related: Current window (P)

jcurwin Work space numbers of all viewports (P)

jviewports
Viewport layout (P)

jviewportlabel Work space labels for all viewport buttons (P)

# jviewportlabel Work space labels for all viewport buttons (P)

Description: An arrayed global parameter, set to the labels on the toolbar buttons used to

switch viewports. It is used by the viewport editor under **Edit -> Viewports**.

See also: NMR Spectroscopy User Guide, VnmrJ Walkup

Related: jviewport Work space numbers of the current viewports (P)

vpaction Set initial state for multiple viewports (M)

# jviewports Viewport layout (P)

Description: An arrayed global parameter, used to keep track of the viewport layout. It is

used by the viewport editor under Edit -> Viewports to change the viewport

layout.

Related: jcurwin Work space numbers of all viewports (P)

jviewportWork space numbers of the current viewports (P)jviewportlabelWork space labels for all viewport buttons (P)vpactionSet initial state for multiple viewports (M)

vpset3def Set the viewport state to three default viewports (M)

vpsetup Set new viewports (M)

# jwin Activate and record activity in current window (M)

Syntax: jwin(pane number)

Description: Activates and records the activity in a specific window pane, created by

setgrid, in the VnmrJ graphics window. jwin is executed when you double-

click the left mouse button in a multiple-paned graphics window.

Arguments: pane number is the number of the pane to join.

Examples: jwin(2)

See also: NMR Spectroscopy User Guide

Related: curwin Current window (P)

fontselectOpen FontSelect window (C)mapwinList of experiment numbers (P)setgridActivate selected window (M)setwinActivate selected window (C)



**killft3d** Terminate any ft3d process started in an experiment (M,U)

killplotStop plot jobs and remove from plot queue (M)killprintStop print jobs and remove from print queue (M)kindKinetics analysis, decreasing intensity (M)

kinds Kinetics analysis, decreasing intensity, short form (M)

kini Kinetics analysis, increasing intensity (M)

kinis Kinetics analysis, increasing intensity, short form (M)

#### killft3d Terminate any ft3d process started in an experiment (M,U)

Syntax: killft3d(exp\_number)

Description: Terminates any ft3d program that has been started in the specified VnmrJ

experiment. killft3d can be executed from any experiment. For each ft3d process terminated, the relevant 3D data subdirectory is also deleted. Remote ft3d processes, denoted by the call name ftr3d in the process table (displayed by the UNIX command ps -azx), are not directly terminated by killft3d but die of their own accord due to the deletion of the 3D data

subdirectory.

The killft3d command can also be run as a shellscript from UNIX. Its

function is analogous to the associated VnmrJ macro.

Arguments: exp number is a number from 1 to 9 that identifies the experiment that started

the ft3d program.

Examples: killft3d(4)

See also: NMR Spectroscopy User Guide

Related: ft3d Perform a 3D Fourier transform (M,U)

# killplot Stop plot jobs and remove from plot queue (M)

Description:

Kills all current plot jobs in the plot queue for the active plotter in VnmrJ, then removes the jobs from the plot queue. Unless the user executing killplot is root, only that user's plot jobs are deleted from the plot queue. To kill a plot that is in progress (i.e., a plot in which you have not entered page), use the page ('clear') command.

The plotter may have to be reinitialized after killplot is executed. To reinitialize the plotter, turn it off and then back on after a few seconds. Hewlett-Packard (HP) pen plotters appear to be more susceptible to this problem than the other HP output devices supported by VnmrJ.

If one port is configured to be both a printer and a plotter, killplot can cause both plot and print jobs to that port to be deleted. For example, if printer='LaserJet\_300', plotter='LaserJet\_300R', and a
plot command pl pscale page is followed by a print command ptext(vnmruser+'/psglib/noesy.c'), entering killplot deletes

both jobs.

See also: NMR Spectroscopy User Guide

Related: killprint Stop print jobs and remove from print queue (M)

page Move plotter forward one or more pages (C)

pl Plot spectra (C)

pscale Plot scale below spectrum or FID (C)

ptext
Print out a text file (M)

showplotq Display plot jobs in plot queue (M)

#### killprint Stop print jobs and remove from print queue (M)

Description: Kills all current print jobs in the print queue for the active printer in VnmrJ, then

removes the jobs from the print queue. Unless the user executing killprint is root, only that user's print job is deleted from the print queue. It is slightly possible that the printer may have to be reinitialized after the execution of this macro. To reinitalize the printer, turn it off, wait a few seconds, and then turn it

back on.

If one port is configured to be both a printer and a plotter, killprint can cause both print *and* plot jobs to that port to be deleted. For example, if printer='LaserJet\_300', plotter='LaserJet\_300R', and a plot command plans called page is followed by a print command.

plot command pl pscale page is followed by a print command
ptext (vnmruser+'/psglib/noesy.c'), entering killprint

deletes both jobs.

See also: NMR Spectroscopy User Guide

Related: killplot Stop plot jobs and remove from plot queue (M)

ptext Print out a text file (M)

showprintq Display print jobs in print queue (M)

#### kind Kinetics analysis, decreasing intensity (M)

Description: If the signal decreases exponentially toward a limit, the output is matched by I

= A1 \* EXP(-T/TAU) + A3. This macro supplies the necessary keywords to the analyze command, which uses the output of fp (i.e., the file fp.out) as

input. The results can be displayed with expl.

See also: NMR Spectroscopy User Guide

Related: analyze Generalized curve fitting (C)

expl Display exponential/polynomial curves (C)

fp Find peak heights (C)

kinds Kinetic analysis, decreasing intensity, short form (M)

kini Kinetics analysis, increasing intensity (M)

kinis Kinetic analysis, increasing intensity, short form (M)

# kinds Kinetics analysis, decreasing intensity, short form (M)

Description: Produces a summary of the results from kind.

See also: NMR Spectroscopy User Guide

Related: kind Kinetics analysis, decreasing intensity (M)

#### kini Kinetics analysis, increasing intensity (M)

Description: If the signal increases exponentially toward a limit, the output is matched by

I = -A1 \* EXP(-T/TAU) + A3 - A1. This macro supplies the necessary

keywords to the  ${\tt analyze}$  command, which uses the output of  ${\tt fp}$  (i.e., the file

fp.out) as input. The results can be displayed with expl.

See also: NMR Spectroscopy User Guide

Related: kind Kinetics analysis, decreasing intensity (M)

kinis Kinetic analysis, increasing intensity, short form (M)

kinis Kinetics analysis, increasing intensity, short form (M)

Description: Produces a summary of the results from kini.

See also: NMR Spectroscopy User Guide

Related: kind Kinetics analysis, decreasing intensity (M)

kini Kinetics analysis, increasing intensity (M)

# L

lastlk Last lock solvent used (P)

lastmenu Menu to display when Return button is selected (P)

latch Frequency synthesizer latching (P)

Line broadening in directly detected dimension (P)
 Line broadening in 1st indirectly detected dimension (P)
 Line broadening in 2nd indirectly detected dimension (P)

lc1d Pulse sequence for LC-NMR (M)

1cpar2d Create 2D LC-NMR acquisition parameters (M)

lcpeak Peak number (P)
lcplot Plot LC-NMR data (M)

lcpsgset Set up parameters for various LC-NMR pulse sequences (M)

lcset2dGeneral setup for 2D LC-NMR experiments (M)leftSet display limits to left half of screen (C)legrelayIndependent control of magnet leg relay (P)

length Determine length of a string (C)

List files in directory (C)

liampAmplitudes of integral reset points (P)lifrqFrequencies of integral reset points (P)

liqbear Liquids Bearing Air Level (P)

listenoffDisable receipt of messages from send2Vnmr (M)listenonEnable receipt of messages from send2Vnmr (M)

1kof Track changes in lock frequency (P)

112d Automatic and interactive 2D peak picking (C)
112dbackup Copy current ll2d peak file to another file (M)
112dmode Control display of peaks picked by ll2d (P)

11amp List of line amplitudes (P)11frq List of line frequencies (P)

In Find natural logarithm of a number (C)load Load status of displayed shims (P)

loadcolors Load colors for graphics window and plotters (M)

Location of sample in tray (P)

locaction Locator action (M)

lockSubmit an Autolock experiment to acquisition (C)lockacqtcLock loop time constant during acquisition (P)

lockfreqLock frequency (P)lockgainLock gain (P)lockphaseLock phase (P)lockpowerLock power (P)locktcLock time constant (P)

logate Transmitter local oscillator gate (P)

lookupLook up words and lines from a text file (C)locprotoexecExecute a protocol from the locator (M)

First-order phase in directly detected dimension (P)

L

1p1 First-order phase in 1st indirectly detected dimension (P)
1p2 First-order phase in 2nd indirectly detected dimension (P)

lpalgLP algorithm in np dimension (P)lpalg1LP algorithm in ni dimension (P)lpalg2LP algorithm in ni2 dimension (P)lpextLP data extension in np dimension (P)lpext1LP data extension in ni dimension (P)lpext2LP data extension in ni2 dimension (P)

lpfilt LP coefficients to calculate in np dimension (P) LP coefficients to calculate in ni dimension (P) lpfilt1 1pfilt2 LP coefficients to calculate in ni2 dimension (P) LP number of data points in np dimension (P) 1pnupts LP number of data points in ni dimension (P) 1pnupts1 1pnupts2 LP number of data points in ni2 dimension (P) lpopt LP algorithm data extension in np dimension (P) lpopt1 LP algorithm data extension in ni dimension (P) 1popt2 LP algorithm data extension in ni2 dimension (P)

lpprintLP print output for np dimension (P)lpprint1LP print output for ni dimension (P)lpprint2LP print output for ni2 dimension (P)lptraceLP output spectrum in np dimension (P)lptrace1LP output spectrum in ni dimension (P)lptrace2LP output spectrum in ni2 dimension (P)

List files in directory (C)

lsfidNumber of complex points to left-shift the np FID (P)lsfid1Number of complex points to left-shift ni interferogram (P)lsfid2Number of complex points to left-shift ni2 interferogram (P)

lsfrqFrequency shift of the fn spectrum (P)lsfrq1Frequency shift of the fn1 spectrum (P)lsfrq2Frequency shift of the fn2 spectrum (P)lv1Zero-order baseline correction (P)

lvltlt Control sensitivity of lvl and tlt adjustments (P)

# lastlk Last lock solvent used (P)

Description: Contains the name of the last lock solvent. Intended for use with the optional

sample changer, this parameter is a user global variable (stored in the user's global file) and is not accessible to multiple users simultaneously. On a multiuser automation run, you should preferably access the last lock solvent

from the file /vnmr/acqqueue/lastlk.

Values: String containing the name of the solvent.

See also: NMR Spectroscopy User Guide
Related: solvent Lock solvent (P)

#### lastmenu Menu to display when Return button is selected (P)

Description: Contains the name of the menu to display when the Return button is clicked on

certain menus. For example, if the Phase F2 button in the 2D Processing menu

(controlled by the file process\_2D) is clicked, lastmenu is set to 'process 2D', the ft and aph commands are executed, the ds window is

opened, and the Interactive 1D Spectrum Display menu (ds\_1 file) is

displayed. Appearing in this menu is a Return button. Because lastmenu is still set to 'process\_2D', clicking on the Return button redisplays the 2D Processing menu. lastmenu is stored in the \$vnmrsys/global file.

Values: String containing the name of a menu (e.g., 'process 2D').

See also: User Programming

Related: menu Change status of menu system (C)

newmenu Select a menu without immediate activation (C)

# latch Frequency synthesizer latching (P)

Description: Configuration parameter for whether the PTS frequency synthesizer has

latching capabilities (all digits of the frequency value are sent to the synthesizer at once). The value for each channel is by the Latching label in the Spectrometer

Configuration window.

Values: 'n' indicates the synthesizers do not have latching capabilities (Not Present

choice from the Spectrometer Configuration window).

'y' indicates the synthesizers have latching capabilities (Present choice from

the Spectrometer Configuration window).

See also: VnmrJ Installation and Administration

Related: config Display current configuration and possibly change it (M)

# 1b Line broadening in directly detected dimension (P)

Description: Sets line broadening and exponential weighting along the directly detected

dimension. This dimension is often referred to as the f2 dimension in 2D data

sets, the f<sub>3</sub> dimension in 3D data sets, etc.

Values: A positive value gives the desired line broadening, in Hz, which is then used to

calculate a decaying exponential function of the form  $\exp(-t*\pi*lb)$ .

A negative value gives a resolution enhancement function (increasing

exponential) of the form  $\exp(-t*\pi*lb)$ .

'n' turns off line broadening and exponential weighting.

See also: NMR Spectroscopy User Guide

Related: exp Find exponential value of a number (C)

Line broadening in 1st indirectly detected dimension (P)
Line broadening in 2nd indirectly detected dimension (P)

#### 1b1 Line broadening in 1st indirectly detected dimension (P)

Description: Sets line broadening and exponential weighting along the first indirectly

detected dimension. This dimension is often referred to as the  $f_1$  dimension in multidimensional data sets. 1b1 works analogously to the parameter 1b. The "conventional" parameters (1b, gf, etc.) operate on the detected FIDs, while

this "2D" parameter is used during processing of the interferograms.

ĺ

Values: A positive value gives the desired line broadening, in Hz, which is then used to

calculate a decaying exponential function of the form  $\exp(-t*\pi*lb1)$ . A

typical value is between 0.0001 to 1000 Hz.

A negative value gives a resolution enhancement function (increasing

exponential) of the form exp(-t\*p\*lb1).
'n' turns off line broadening and exponential weighting.

See also: NMR Spectroscopy User Guide

Related: exp Find exponential value of a number (C)

Line broadening in directly detected dimension (P)
Line broadening in 2nd indirectly detected dimension (P)

# 1b2 Line broadening in 2nd indirectly detected dimension (P)

Description: Sets line broadening and exponential weighting along the second indirectly

detected dimension. This dimension is often referred to as the  $f_2$  dimension in multidimensional data sets. 1b2 works analogously to the parameter 1b. 1b2

can be set with wti on the 2D interferogram data.

Values: A positive value gives the desired line broadening, in Hz, which is then used to

calculate a decaying exponential function of the form  $\exp(-t*\pi*lb2)$ .

A negative value gives a resolution enhancement function (increasing

exponential) of the form  $\exp(-t*\pi*lb2)$ .

'n' turns off line broadening and exponential weighting.

See also: NMR Spectroscopy User Guide

Related: exp Find exponential value of a number (C)

Line broadening in directly detected dimension (P)

wti Interactive weighting (C)

#### 1c1d Pulse sequence for LC-NMR (M)

Applicability: Systems with LC-NMR accessory.

Description: Creates parameters to set up a pulse sequence that can be used to start an LC-

NMR run, including triggering the injection of a sample, and can be used also to obtain multiple solvent-suppressed spectra using multi frequency Shifted Laminar Pulses (SLP) and gradients. The sequence is coded without a d2 variable, thus allowing ni to be used to obtain a series of spectra without

resulting in any delay in the sequence being incremented.

The sequence requires a phase table, lcld, to be found in the tablib directory. Phases of the selective pulses, the observe pulse, and the receiver and

separately controlled by phase variables.

Note that the lcld sequence uses power scaling of shaped pulses, which is supported starting in VnmrJ 5.2. Because of this feature, this sequence will not

run in earlier versions of VnmrJ.

# 1cpar2d Create 2D LC-NMR acquisition parameters (M)

Applicability: Systems with LC-NMR accessory.

Description: Creates the acquisition parameters ni, sw1, and phase, which can be used to

acquire a 2D LC-NMR data set. lcpar2d is functionally the same as

addpar('2d').

Related: addpar Add selected parameters to current experiment (M)

lcset2d General setup for 2D LC-NMR experiments (M)

lcpeak Peak number (P)

Applicability: Systems with LC-NMR accessory.

Description: Contains the number of the peak being sensed or the loop being flushed.

lcplot Plot LC-NMR data (M)

Applicability: Systems with LC-NMR accessory.

Syntax: lcplot

Description: Plots LC-NMR data. This macro is executed with the Plot LC-NMR button on

the Spare pane when LC-NMR is active.

1cpsgset Set up parameters for various LC-NMR pulse sequences (M)

Applicability: Systems with LC-NMR accessory.

Syntax: lcpsgset(file,parameter1,parameter2,...,parameterN)

Description: Sets up parameters for various LC-NMR pulse sequences using information in

a parlib file. Rather than returning the entire parameter file, lcpsgset returns the parameters listed. lcpsgset, in general, is never entered from the

keyboard but is used as part of experiment setup macros.

Arguments: file is the file from the user or system parlib that provides information on

setting up parameters listed. The parameters seqfil and pslabel are set to

the supplied file name.

parameter1, parameter1, ..., parameterN are 1 to 11 parameters to

be returned from the parlib file.

Examples: lcpsgset('lccosy','ds','ap','ss','d1','axis','phase')

lcset2d General setup for 2D LC-NMR experiments (M)

Applicability: Systems with LC-NMR accessory.

Syntax: lcset2d(experiment<,F2 dig res<,F1 dig res>>)

Description: Runs the macro lcpar2d to create new parameters needed for 2D LC-NMR

experiments, then selects starting values for a number of parameters. The lcset2d macro is "internal" and not normally entered directly by the user.

Arguments: experiment is the name of a 2D LC-NMR experiment.

F2\_dig\_res is the  $f_2$  digital resolution desired, in Hz/pt. F1 dig res is the  $f_1$  digital resolution desired, in Hz/pt.

Examples: lcset2d('lcnoesy')

left Set display limits to left half of screen (C)

Description: Sets the horizontal control parameters sc and wc to produce a display (and

subsequent plot) in the left half of a screen (and page). For 2D data, space is left

for the scales.

Related: center Set display limits for center of screen (C)

full Set display limits for a full screen (C)

fullt Set display limits for full screen with room for traces (C)

right Set display limits for right half of screen (C)

### legrelay Independent control of magnet leg relay (P)

Description: Gives override capability over the magnetic leg high and low (broad) band rf

signal routing. This parameter does not normally exist but can be created by the

user with the command create('legrelay', 'string').

The legrelay override is operational only on standard systems shipped starting in November 1990 and on certain special systems shipped before that date. A system includes the override capability if it uses N-type connectors instead by BNC connectors on the magnet leg.

Values: 'n' indicates normal logic is used to set the leg relay.

ii maleutes normal logic is used to set the leg letay.

'h' indicates the leg relay is set to the high band.
'l' indicates the leg relay is set to the low (broad) band.

Any other value results in an error message and an abort of pulse sequence

generation.

See also: User Programming

Related: create Create new parameter in a parameter tree (C)

#### length Determine length of a string (C)

Syntax: length(string):\$string length

Description: Returns the length in characters of a specified string.

Arguments: string is zero or more characters enclosed in single quotes.

string length is the number of characters (a real number) in string.

Examples: length('abc'):r1

length(solvent):\$len

See also: *User Programming* 

Related: substring from a string (C)

# 1f List files in directory (C)

Syntax: lf<(directory)>

Description: Lists the files in a directory, with output on the text output window. Directories

are suffixed by "/", executable files by "\*", and links by "@".

Arguments: directory is the name of a directory. The default is the current working

directory. 1f is equivalent to the UNIX command 1s -F and uses the same

options (e.g., -1 for a long listing such as lf('-1 \*.fid')).

Examples: 1f

lf('data'))
lf('-l \*.fid')

See also: NMR Spectroscopy User Guide

Related: dir List files in directory (C)

1s List files in directory (C)

#### liamp Amplitudes of integral reset points (P)

Description: Stores the integral amplitudes at the integral reset points for a list of integrals.

To display the values of liamp, enter display ('liamp'). Values of liamp can also be accessed in MAGICAL macros using, for example, liamp[Sil Values are stored as absolute numbers (summations of data point).

liamp [\$i]. Values are stored as absolute numbers (summations of data point values) and, as such, are a function of the parameter fn. The values displayed

П

by the dli, pir, and dpir programs are related to liamp values by the relationship:

Displayed or plotted integral = liamp[i] \*is/(fn/128) \*ins)

See also: NMR Spectroscopy User Guide

Related: display Display parameters and their attributes (C)

dli Display list of integrals (C)

dpir Display integral amplitudes below spectrum (C)

fn Fourier number in directly detected dimension (P)

Frequencies of integral reset points (P)

Plot integral amplitudes below spectrum (C)

# lifrq Frequencies of integral reset points (P)

Description: Stores the frequencies of integral reset points for a list of integrals. The

frequencies are stored in Hz and are not adjusted by the reference parameters

rfl and rfp.

See also: NMR Spectroscopy User Guide

Related: liamp Amplitudes of integral reset points (P)

rfl Ref. peak position in directly detected dimension (P)
rfp Ref. peak frequency in directly detected dimension (P)

# liqbear Liquids Bearing Air Level (P)

Description: This global parameter is the DAC value used when the liquids spinner bearing

air is turned on. If the parameter does not exist the value defaults to 0xc000.

To create the parameter:

create('liqbear','integer','global')
setlimit('liqbear',65535,0,1,'global')

Values: 0 - 65535

# listenoff Disable receipt of messages from send2Vnmr (M)

Description: Deletes the file \$vnmruser/.talk, thereby disallowing send2Vnmr to

send commands to VnmrJ

See also: User Programming

Related: listenon Enable receipt of messages from send2Vnmr (M)

send2vnmr Send a command to VnmrJ (U)

#### listenon Enable receipt of messages from send2Vnmr (M)

Description: Writes files with the VnmrJ port number that /vnmr/bin/send2Vnmr

needs to talk to VnmrJ. The command then to send commands to VnmrJ is

/vnmr/bin/send2Vnmr \$vnmruser/.talk command.

See also: User Programming

 $Related: \begin{array}{cc} \textbf{listenoff} & Disable\ receipt\ of\ messages\ from\ \texttt{send2Vnmr}\ (M) \end{array}$ 

send2vnmr Send a command to VnmrJ (U)

# 1kof Track changes in lock frequency (P)

Description: Tracks changes in the lock frequency resulting from changes in the solvent, and

minor changes caused by the magnet drifting. The frequency units for 1kof are in Hz, analogous to sfrq and tof, or dfrq and dof. 1kof affects two

components of the system: autolock on the console and acqi on the host computer. If lkof exists, it offsets the current value of the lockfreq parameter.

See also: NMR Spectroscopy User Guide
Related: lockfreq Lock frequency (P)

#### 112d Automatic and interactive 2D peak picking (C)

Syntax: (1) 112d< (options) ><:\$num>

(2) 112d('info'<, #>): \$peak\_number, \$f1, \$f2, \$amplitude, \$volume, \$label, \$comment, \$FWHH1, \$FWHH2, \$f1\_min, \$f1 max, \$f2 min, \$f2 max

Description:

Automatically finds and integrates peaks that are above the threshold th in a 2D spectrum or a 2D plane of a 3D spectrum, and writes the peak location, volume, full-width at half-height (FWHH), volume, and the boundaries of the integrated region to a file in the 112d subdirectory of the current experiment directory. For 2D spectra, the file name is peaks.bin, and for 2D planes of 3D spectra, the file name is peaks\_f#f#\_#.bin, where f#f# gives the plane direction (e.g., f1f3) and the final # gives the number of the plane. For easy import and export of peak data, 112d also allows insertion and deletion of peaks interactively as well as reading and writing of text peak files.

Two-dimensional volumes are scaled in a manner analogous to 1D integrals, using the parameters <code>ins2</code> and <code>ins2ref</code>. The <code>ins2ref</code> parameter is the Fourier number scaled value of a selected volume. The reported value of a peak volume is (<code>unscaled volume</code>) × <code>ins2/ins2ref/fn/fn1</code>. The unscaled volume of a peak can be obtained from the command <code>ll2d('info', peak#).ins2ref</code> can be set to the unscaled value divided by <code>fn</code> and <code>fn1</code>. The report volume for that peak is then the value of <code>ins2</code>.

Arguments:

options (syntax 1) are any of the following (dconi is not necessarily active):

- 'adjust' is a keyword to adjust the bounds of all peaks in the displayed area so that no boundaries overlap, and then to recalculate peak volumes.
- 'draw' is a keyword to draw the peaks, boxes, numbers, and labels on the spectrum based on the value of the parameter 112dmode.
- 'info', 'total' displays the total number of peaks in the current peak table. If a single return value is requested, printing is suppressed and the total number of peaks is returned.
- 'peaks' is a keyword to find all peaks in the displayed area above a threshold th. If dconi is active and in the box mode, ll2d finds peaks only in the area defined by the cursors. The 'peaks' option is the default if no arguments are entered.
- 'pos' or 'neg' keywords can be used in addition to 'peak',
  'volume', or 'clear' to operate only on positive or negative peaks.
- 'read'<, file > reads in a binary peak file, where file is the name of the peak file. If a full path is not specified, the file is searched for first in the current working directory and then in the ll2d subdirectory of the current experiment directory.
- 'readtext'<, file> reads in a text peak file, where file is the name of the peak file. If a full path is not specified, the file is searched for first in the current working directory and then in the ll2d subdirectory of the current experiment directory.

- 'reset' is a keyword to delete all peaks in the peak table.
- 'volume' is a keyword to find the bounds of each peak in the displayed area and integrate this area.
- 'writetext' < , file > writes a peak file to a text file, where file is the name of the text file written. If a full path is not specified, the file is written in the current working directory.

options (syntax 1) can also be any of the following (dconi must be active):

- 'clear' is a keyword to delete all peaks in the displayed region if in the dconi cursor mode, or to delete all peaks within the cursors if in the dconi box mode.
- 'combine' is a keyword to combine all peaks within the area defined by the cursors into a single peak (in dconi box mode only). The center of the new peak is at the average of all combined peaks' centers, and the bounds of this peak contains the maximum extents of the combined peaks' bounds. If all combined peaks have the same label, this label is assigned to the new peak. CAUTION: All individual peaks to be combined are deleted prior to the creation of the new combination peak, and there is no automatic way to restore the original peaks. Therefore, it is recommended that you make a backup copy of the peak file prior to using this option.
- 'comment' is a keyword to prompt for an 80-character comment. The comment is assigned to the nearest peak in the dconi cursor mode or to all peaks within the cursors in the dconi box mode.
- 'comment', text executes the 'comment' option using the string entered for text instead of prompting for a comment.
- 'label' is a keyword to prompt for a 15-character label. The label is assigned to the nearest peak in dconi cursor mode or assigned to all peaks within the cursors in dconi box mode. To erase an existing label, enter a label consisting of one or more spaces.
- 'label', text executes the 'label' option using the string entered for text instead of prompting for a label.
- 'mark' is a keyword to insert a peak at the current cursor position if in the dconi cursor mode. If in the dconi box mode, 'mark' is a keyword to integrate the area within the cursors and assign that area to all peaks within the cursors that do not have their bounds already defined. If there are no peaks within the area defined by the cursors, using 'mark' finds the highest point within this area, marks that as a peak, integrates the area within the cursors, and assigns that area to the peak. The displayed values of the volume integrals are scaled by ins2 and ins2ref and the Fourier number of the 2D experiment.
- 'unmark' is a keyword to delete the nearest peak if in dconi cursor mode. If in the dconi box mode, 'unmark' deletes all peak bounds that are completely within the area defined by the cursors. Peaks are not deleted in the box mode.

options (syntax 1) also can be any of the following (dconi does not have to be active because 112d is executed on a peak number):

• 'combine', #1, #2, ... executes the 'combine' option on the list of peak numbers that follow the 'combine' keyword. If a single return value is requested, the peak number of the new combination peak is returned.

- 'comment', text, # executes the 'comment' option on peak # using the string entered for text instead of prompting for a comment.
- 'label', text, # executes the 'label' option on peak # using the string entered for text instead of prompting for a label.
- 'unmark', # deletes peak number #.

\$num (syntax 1) is a return value set to the total number of peaks that have been
picked unless the arguments 'combine', #1, #2, ... are used, in which
case \$num is the number of the newly created combination peak.

Syntax 2 arguments are the following:

- 'info'<, #> displays information in the text window about peak number #. If no peak number is included, dconi must be active and the default is the peak nearest to the cursor. If return values are requested, the display is suppressed.
- \$peak\_number is a return value set to the number of the peak, either the second argument # or, if no value is given for #, the peak nearest to the cursor in dconi.
- \$f1 and \$f2 are return values set to the peak frequencies in f<sub>1</sub> and f<sub>2</sub> of peak \$peak number.
- \$amp is a return value set to the amplitude of peak \$peak number.
- \$vol is a return value set to the unscaled volume of \$peak\_number. peak. This value can be used to set the ins2ref parameter.
- \$label is a return value set to the label of peak \$peak number.
- \$comment is a return value set to the comment about \$peak\_number.
- \$FWHH1 and \$FWHH2 are return values set to full-width at half-height of \$peak number.
- \$f1\_min, \$f1\_max, \$f2\_min, \$f2\_max are return values set to the bounds of \$peak number.

112d('info','total'):\$npeaks
112d('combine',3,4,5,6):\$cpn
112d('info',3):\$num,\$f1,\$f2,\$amp,\$vol,\$label

See also: NMR Spectroscopy User Guide

Related: dconi Interactive 2D contour display (C)

ins2 2D volume value (P)

ins2ref Fourier number scaled volume of a peak (P)

112dbackup Copy current 112d peak file to another file (M)

112dmode Control display of peaks picked by 112d (P)

par112d Create parameters for 2D peak picking (M)

pll2d Plot results of 2D peak picking (C)

th Threshold (P)

th2d Threshold for integrating peaks in 2D spectra (P)

xdiag Threshold for excluding diagonal peaks when peak picking (P)

### 112dbackup Copy current II2d peak file to another file (M)

Syntax: 112dbackup<(file)>

Description: Backs up the current 112d peak file by copying it to a file with a different file

name. The default 112d peak file is peaks.bin for 2D data.

Arguments: file is the name to be given to the backup file. If a full path is not specified,

the file is written to the current working directory. If no argument is provided, the system prompts for a file name. If no file name is specified at the prompt,

the default 112d peak file name with .bck appended is used.

See also: NMR Spectroscopy User Guide

Related: 112d Automatic and interactive 2D peak picking (C)

#### 112dmode Control display of peaks picked by II2d (P)

Description: Sets the display attributes of peaks picked by the 112d command

Values: A string variable composed of 4 characters, with each character taking the value

'y' (display the peak attribute) or 'n' (do not display the attribute). The first character determines if a "+" is drawn on the screen in dconi displays to mark peaks, the second character controls the drawing of the peak number, the third character controls drawing of the peak bounds box, and the last character

controls drawing of the peak label.

See also: NMR Spectroscopy User Guide

Related: 112d Automatic and interactive 2D peak picking (C)

#### 1lamp List of line amplitudes (P)

Description: Stores a list of line amplitudes above the threshold set by th.

See also: NMR Spectroscopy User Guide

Related: dll Display listed line frequencies and intensities (C

llfrq List of line frequencies (P)

th Threshold (P)

# 11frq List of line frequencies (P)

Description: Stores a list of line frequencies above the threshold set by th. Frequencies are

stored in Hz and are *not* adjusted by reference parameters rfl and rfp.

See also: NMR Spectroscopy User Guide

Related: llamp List of line amplitudes (P)

rfl Ref. peak position in directly detected dimension (P)
rfp Ref. peak frequency in directly detected dimension (P)

th Threshold (P)

#### 1n Find natural logarithm of a number (C)

Syntax: ln(value)<:n>

Description: Finds the natural logarithm (base e) of a number. To convert the value to base

10, use  $log_{10}x = 0.43429*ln(x)$ .

Arguments: value is a number.

n is the return value giving the logarithm of value. The default is to display

the logarithmic value in the status window.

L

Examples: ln(.5)

ln(val):ln val

See also: User Programming

Related: atan Find arc tangent of a number (C)

Find cosine value of an angle (C)

Find exponential value of a number (C)

Find sine value of an angle (C)

Find sine value of an angle (C)
tan
Find tangent value of an angle (C)

# load Load status of displayed shims (P)

Description: Sets whether shim values are used. load is automatically set to 'y' by the

rts and is automatically set to 'n' by su, go, au, and shim. Shim DAC values are automatically loaded after the console is rebooted (the last values

returned before the console was rebooted).

Values: 'y' begins any noninteractive shimming process or data acquisition after

loading the shim DACs with the shim values from the current experiment. It

also prevents acqi from delivering shim values to that experiment.

'n' begins any noninteractive shimming process or data acquisition with the current values stored in the shim DACs. Shim values in the current experiment

are ignored.

See also: NMR Spectroscopy User Guide

Related: acqi Interactive acquisition display process (C)

au Submit experiment to acquisition and process data (C)

Submit experiment to acquisition (C)

rts Retrieve shim coil settings (C)

shim Submit an autoshim experiment to acquisition (C) su Submit a setup experiment to acquisition (M)

### loadcolors Load colors for graphics window and plotters (M)

Syntax: loadcolors<(color file)>

Description: Loads the color table for VnmrJ graphics window and plotters. loadcolors

is generated by the color program and includes a series of setcolor commands. On bootup, the bootup macro calls loadcolors to set the

graphics and plotter colors.

The loadcolors macro checks the value of maxpen to decide if the plotter supports colors. If maxpen is greater than 1, a color printer is configured.

Arguments: color file is the name of the file to load. loadcolors first searches for

this file in the directory \$vnmruser/templates/ directory. If not found there, loadcolors then searches the user\_templates/vnmr directory. The default is a color table with the same name as the value of the plotter parameter that loadcolors searches for in the same two directories.

Examples: loadcolors

loadcolors('mycolortable')

See also: *VnmrJ Imaging NMR* 

Related: bootup Macro executed automatically when VnmrJ activated (M)

color Select plotting colors from a graphic interface (M)

maxpen Maximum number of pens to use (P)

setcolor Set colors for graphics window and for plotters (C)

#### loc Location of sample in tray (P)

Description: Indicates whether a sample changer is present and enabled, present but disabled,

or not present. If the changer is present and enabled, the value of loc sets the location in the tray of the sample in use or to be used. The loc parameter is stored in the global tree. When an acquisition is started, certain global parameters, including loc, are saved with the experiment parameters. The saveglobal parameter specifies which global parameters are saved.

The auto\_au macro controls most of the automation features, including

setting the value of loc.

Values: A number between 1 and traymax indicates the sample location.

0 indicates the changer is not present or disabled.

See also: NMR Spectroscopy User Guide; VnmrJ Walkup

Related: auto au Controlling macro for automation (M)

saveglobal Save selected parameters from global tree (P)

traymax Sample changer tray size (P)

# locaction Locator action (M)

Description: Perform an action on an object in the locator database. The action depends on

the type of object selected, the action performed, and the target selected for the

action.

Related: dndfid Retrieve and process fid data from the locator (M)

dndjoinJoin a work space from the locator (M)dndparRetrieve a parameter set from the locator (M)dndshimsRetrieve a shimset set from the locator (M)locprotoexecExecute a protocol from the locator (M)xmmakenodeMake a new study queue node (M)

#### lock Submit an Autolock experiment to acquisition (C)

Description: Performs an automatic locking operation using the acquisition computer,

optimizing lock power, phase, and gain. If necessary, lock obtains lock through a software-controlled search. lock is the only method to automatically adjust lock phase (usually needed only after probe change or lock channel tuning). lock also sets the rf frequencies, decoupler status, and temperature.

See also: NMR Spectroscopy User Guide

Related: au Submit experiment to acquisition and process data (C)

change Submit a change sample experiment to acquisition (M) ga Submit experiment to acquisition and FT the result (C)

go Submit experiment to acquisition (C)

sample Submit change sample, autoshim experiment to acquisition (M)

shimSubmit an Autoshim experiment to acquisition (C)spinSubmit a spin setup experiment to acquisition (C)suSubmit a setup experiment to acquisition (M)

# lockacqtc Lock loop time constant during acquisition (P)

Description: Controls time constant of lock loop during acquisition (i.e., time constant by

which the lock feedback corrects disturbances of the magnetic field).

Values: 1, 2, 3, or 4 (where 1 sets 1.2 seconds, 2 sets 4.7 seconds, 3 sets 12 seconds, and

4 sets 48 seconds).

If lockacqtc does not exist, it is set to 48 seconds. All systems are designed to work well with the default settings, and there should rarely be a reason to alter the lock time constant. However, to experiment with other values, create lockacqtc and set a new value:

```
create('lockacqtc','integer','global')
setlimit('lockacqtc',4,1,1,'global') lockacqtc=n
```

where n is the new value.

See also: NMR Spectroscopy User Guide

Related: create Create new parameter in a parameter tree (C)

locktc Lock time constant (P)

setlimit Set limits of a parameter in a tree (C)

# lockfreq Lock frequency (P)

Description: Sets system lock frequency. The value is entered using the Lock Frequency

label in Spectrometer Configuration window. The value of lockfreq must be set correctly in order to observe NMR signals.

lockfreq can find the lock signal or resonance. Traditionally, Varian spectrometers have used the parameter z0 for this purpose; however, using lockfreq can require less shimming when switching solvents and less adjustment to the lock phase. To use lockfreq, set z0='n'.

Values: 1 to 160 (in MHz), 'n'

Use the true  ${}^2\!H$  frequency. Typical values of lockfreq are shown in the chart below.

<sup>1</sup> H Frequency		
200	30.710	30.6976
300	46.044	46.0625
400	61.395	61.471
500	76.729	
600	92.095	
750	115.250	

Refer to the manual *VnmrJ Installation and Administration* for details on finding the correct lock frequency.

The commands; go, lock, shim, and su reset the lock frequency in the console to the current value of lockfreq. Lock frequency in the console can be set with the sethw command.

lockfreq is offset by the value of lkof, if that parameter exists, but sethw directly uses its numeric argument, without any offset by lkof.

See also: VnmrJ Installation and Administration; NMR Spectroscopy User Guide

Related:	config	Display current configuration and possibly change it (M)
	~~	Submit appariment to acquisition (M)

go Submit experiment to acquisition (M)
lkof Track changes in lock frequency (P)

lock Submit an Autolock experiment to acquisition (C) sethw Set values for hardware in acquisition system (C)

setlockfreq Set lock frequency (C)

shim Submit an Autoshim experiment to acquisition (C)
su Submit a setup experiment to acquisition (M)

Z0 field position (P)

lockgain Lock gain (P)

Description: Contains the current lock gain value as set by computer control. The value is

stored in vnmrsys/global and can be examined by typing lockgain?.

Values: 0 to 48 dB, in 1-dB steps.

See also: NMR Spectroscopy User Guide

lockphase Lock phase (P)

Description: Contains the current lock phase. The value is stored in vnmrsys/global and

can be examined by typing lockphase?.

Values: 0 to 360, in degrees, in 1.4-degree steps.

See also: NMR Spectroscopy User Guide

lockpower Lock power (P)

Description: Contains the current lock power value as set by computer control. The value is

stored in vnmrsys/global and can be examined by typing lockpower?.

Values: 0 to 68 dB, in 1-dB steps, 68 is full power.

See also: NMR Spectroscopy User Guide

locktc Lock time constant (P)

Description: Controls lock loop time constant when system is not performing acquisition

(idle, lock display, shim display, FID display, autoshim, autolock, etc.).

Values: 1, 2, 3, or 4 (where 1 corresponds to 1.2 seconds, 2 to 4.7 seconds, 3 to 12

seconds, and 4 to 48 seconds). If locktc does not exist, the system uses a value of 1, the fastest value. To experiment with other value, create locktc and set a value (e.g., create ('locktc', 'integer', 'global')

setlimit('locktc',4,1,'global') locktc=2).

See also: NMR Spectroscopy User Guide

Related: create Create new parameter in a parameter tree (C)

lockacqtc Lock acquisition time constant (P)
setlimit Set limits of a parameter in a tree (C)

logate Transmitter local oscillator gate (P)

Description: Specifies whether the transmitter local oscillator (L.O.) is gated with the

transmitter rf output or with the transmitter I.F. (intermediate frequency).

The logate parameter does not exist in most parameter sets; the system internally sets it to 'l'. To use the value 's', create logate and change the

value by entering: create('logate','string')
setenumeral('logate',2,'l','s') logate='s'.

Values: '1' makes the transmitter L.O. gate with the rf output, producing better signal-

to-noise, usually most important in liquids NMR.

's' makes the transmitter L.O. gate with the I.F. signal, producing sharper

pulses, especially important in solid-state NMR.

See also: User Guide: Solid-State NMR

Related: create Create new parameter in a parameter tree (C)

setenumeral Set values of a string variable in a tree (C)

П

# lookup Look up words and lines from a text file (C)

Applicability: VnmrJ

Syntax: lookup('codeword', argument<, 'codeword',

argument<,...>>):\$n1<\$n2<,...>>

Description: Search a text file or files for a word or any string of characters delimited by

white space characters (space character, a tab, a new line, a carriage return, or a comma) or codeword and return to the user subsequent words or lines.

The white space characters may be specified. Punctuation marks, unless they are defined as white space as the comma is by default, also form words or are part of a word. A line is any string of characters from the current word to the next carriage return. A line will include all "white space" characters except the carriage return. Depending on the codeword, word searches and word counts can be case insensitive or case sensitive.

The codewords mfile and filekey implement multiple text file lookup and lookup reads the contents of the specified files.

The mfile and file keywords are used together to keep track of various locations within a single file to restart the search from that location.

The first time a file is selected, or the search is restarted at the beginning of the file, use the name of the file instead of the filekey. Subsequent calls to lookup on this file use the value returned by the filekey codeword as the argument following the mfile codeword. The mfile codeword resets the white space to the default values.

Arguments: Default white space characters: space character, tab, new line, carriage return, or comma.

file codeword specifies that the next supplied argument is the name of the active text file. This codeword must be the first argument and the file name must be the second argument passed to lookup. The search through a text file is a top to bottom search. The file codeword resets the search to start from the top of the text file. Subsequent searches through a previously accessed text file will continue from where the previous search stopped provided the file codeword is not used. The file codeword resets the white space characters to their default values.

mfile codeword specifies that the next supplied argument is the filekey to select one of multiple text files to access. This codeword must be first argument and the filekey must be the second argument passed to lookup if mfile is used.

seek this codeword causes the lookup program to search the text file for words which match those supplied as arguments following the seek codeword. An implicit seek is initially assumed for each call to lookup. The lookup program maintains a pointer to the word following the last successful seek. The first argument following an explicit seek codeword is interpreted as a word to search for and not a codeword. The second or later argument following an explicit seek is interpreted as a codeword if it matches one of the nine cases. Therefore, for example, one can search for the word file without having it interpreted as a codeword by having it immediately follow the seek codeword in the argument list. This seek is case insensitive.

seekcs this codeword is the case sensitive equivalent to the seek codeword and follows the same rules as seek. Alternate case sensitive and case insensitive searches are allowed.

skip increments the word pointer to the next word in the text file. This codeword may optionally be followed by a number which will specify how many words to skip.

read returns to the user the word currently being pointed to and increments the pointer to the next word in the text file. This codeword may optionally be followed by a number which will specify how many words to return to the user.

readline returns to the user the word currently being pointed to and all following words until the end of the current line. The pointer is moves to the first word of the next line in the text file. This codeword may optionally be followed by a number which will specify how many lines to return to the user.

count returns to the user the number of times words in the text file match the subsequent argument. The count starts at the current word pointer and proceeds to the end of the text file. The word count is not case sensitive.

countcs this codeword is the case sensitive equivalent to the count codeword. In all other respects, it is the same as count.

delimiter this codeword specifies that the next supplied argument is a list of characters which are used to identify the white space used to identify words.

Characters are specified by the following:

```
\n — new line
\t — tab
\r — carriage return
\\ — backslash
\' — single quote.
```

The two arguments delimiter,' \t\n\r', reselect the default white space. The file codeword will also reselect the default white space. The distinction is that the file codeword restarts the search from the beginning of the file while the delimiter codeword continues from the current search position. An implicit seek is applied following the 'delimiter' codeword and argument.

filekey returns the current location within the file being accessed. Combined with the mfile codeword, a subsequent call to lookup starts the search at the location within the file specified by the value of filekey. The filekey serves both as a pointer to the file and as the character offset within that file.

Examples:

```
lookup('file', systemdir + '/manual/lookup')
Select this file for the search.
```

```
lookup('user','skip',2,'read',2,'readline'):$n1,$n2,
$n3,$ret
```

Seek is assumed with the call to lookup. Finding the word user the next instruction, 'skip', 2, causes the pointer to jump two words. The codeword read causes the word to be put into \$n1. The argument 2 specifies two words to be read into \$n2. The word pointer now points to the next. The codeword readline causes the remaining characters up to the next carriage return to be placed in \$n3. The pointer now points to the first word in the next line. The variable \$ret is set to the number of arguments successfully returned from the text file and is used to determine if the end of the text file has been reached.

```
lookup('skip',8,'read','skip',3,'read',2,'seek','com
ma'):$n3,$n4,$n5
```

'Skip', 8 causes the pointer to jump eight words. The 'read' sets \$n3 equal to word where the pointer is now located. 'Skip', 3 jumps the next three words. 'Read', 2 reads two consecutive words and sets \$n4 to the first word and \$n5 equal the second word. The seek argument searches for the word 'comma'. If the word 'comma' is at the end of a sentence it will not be found because the period is treated (by default) as part of the word. Define the period as a white space and occurrences comma at the end of sentences are also found. The word pointer now points to the next word.

```
lookup('delimiter',' ,\'.\n\t"','seek','file',
'skip',6,'read'):n6
```

The delimiter with the argument ',\'.\n\t"' sets white space to space, comma, single quote, period, new line, tab, and double quote. Setting single quotes to white space causes the explicit seek to select the next argument file as a search word not a codeword. The search for the word must matches both MUST and must because seek is not case sensitive. 'Skip', 6 jumps six words. Read sets \$n6 equal to word found between the next set of single quotes because single quotes are defined as white space.

lookup('seekcs','Test','read'):\$n7

seekcs is the case sensitive form of seek and searches for the word that is an exact match to the case of Test (the argument following the codeword seekcs). Finding the word 'Test', read sets \$n7 to search. Any

occurrence of the word test is skipped.

See also: *User Programming* 

Related: dialog Display a dialog box from a macro (C)

> VnmrJ system directory (P) systemdir

#### locprotoexec Execute a protocol from the locator (M)

Description: When a protocol is dragged from the locator and dropped onto the graphics

canvas, this macro adds the protocol to the end of the study queue, and executes

the macro associated with the protocol.

Related: dndfid Retrieve and process fid data from the locator (M)

> dndjoin Join a work space from the locator (M) Retrieve a parameter set from the locator (M) dndpar dndshims Retrieve a shimset set from the locator (M)

locaction Locator action (M)

xmmakenode Make a new study queue node (M)

#### First-order phase in directly detected dimension (P) 1p

Specifies the first-order phase-correction angles along the directly detected Description:

dimension according to the formula

absorption spectrum( $\omega$ ) =

real channel( $\omega$ ) \* sin  $\theta$  + imaginary channel( $\omega$ ) \* cos  $\theta$ 

where the phase angle  $\theta$  is a function of frequency, i.e.

 $\theta = \mathbf{rp} + (\omega - \omega_0) * 1\mathbf{p}$ 

 $\omega_0$  is defined to be the right end of the spectrum (i.e., 1p has zero effect at the right edge of the spectrum and a linearly increasing effect going to the left). In multidimensional data sets, 1p controls the phase of the directly detected dimension: f<sub>2</sub> dimension in 2D data sets, f<sub>3</sub> dimension in 3D data sets, etc.

Values: -3600 to +3600, in degrees. Typical values are between 0 and -180.

See also: NMR Spectroscopy User Guide

Related: aph Automatic phase adjustment of spectra (C)

> lp1 First-order phase in 1st indirectly detected dimension (P) 1p2 First-order phase in 2nd indirectly detected dimension (P) Zero-order phase in directly detected dimension (P) rp

Set parameters for zero linear phase (M) setlp0

#### lp1 First-order phase in 1st indirectly detected dimension (P)

Controls the first-order phase constant along the first indirectly detected Description:

dimension during the process of phase-sensitive 2D transformation. The first

indirectly detected dimension is often referred to as the  $\mathbf{f}_1$  dimension of a multidimensional data set.

See also: NMR Spectroscopy User Guide

Related: 1p First-order phase in directly detected dimension (P)

First-order phase in 2nd indirectly detected dimension (P)

Zero-order phase in 1st indirectly detected dimension (P)

# 1p2 First-order phase in 2nd indirectly detected dimension (P)

Description: Controls the first-order phase constant along the second indirectly detected

dimension during a ds, dconi, or equivalent display operation on the 2D data or a 1D trace therein. The second indirectly detected dimension is often referred

to as the f<sub>2</sub> dimension of a 3D (or higher dimensionality) data set.

See also: NMR Spectroscopy User Guide

Related: dconi Interactive 2D contour display (C)

ds Display a spectrum (C)

1p First-order phase in directly detected dimension (P)
rp2 Zero-order phase in 2nd indirectly detected dimension (P)

# lpalg LP algorithm in np dimension (P)

Description: Specifies the linear prediction (LP) algorithm to use in the np dimension. The

resulting LP coefficients are used to appropriately extend the complex time-domain data prior to a normal Fourier transform. The LP algorithms work both on complex  $t_2$  FIDs and on hypercomplex or complex  $t_1$  interferograms. Enter addpar ('lp') to create lpalg and other np dimension LP parameters in

the current experiment

Values: 'lpfft' does a least-squares calculation of lpfilt complex LP coefficients using lpnupts complex time-domain data points. Eigenvalue decomposition of the least-squares matrix is done using Householder tridiagonalization

followed by the QL method with implicit shifts.

'lparfft' does a non-least-squares calculation of lpfilt complex LP coefficients using (lpfilt+1) complex, autoregressive (AR) matrix elements. These AR matrix elements are calculated from the raw, complex timedomain data using lpnupts points.

Note that the 'lpfft' algorithm is preferred by far. While 'lparfft' can model broad lines and can extend data sets when mostly noise exists, it cannot model narrow lines.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

lpalg1 LP algorithm in ni dimension (P)
lpalg2 LP algorithm in ni 2 dimension (P)
lpext LP data extension in np dimension (P)

lpfilt LP coefficients to calculate in np dimension (P)
lpnupts LP number of data points in np dimension (P)
lpopt LP algorithm data extension in np dimension (P)

lpprint LP print output in np dimension (P)
lptrace LP output spectrum in np dimension (P)

np Number of data points (P)
proc Type of processing on np FID (P)

strtlpStarting point for LP calculation in np dimension (P)strtextStarting point for LP data extension in np dimension (P)

ı

#### lpalg1 LP algorithm in ni dimension (P)

Description: Specifies the LP (linear prediction) algorithm to use in the ni dimension.

lpalg1 functions analogously to lpalg. Enter addpar ('lp', 1) to create lpalg1 and other ni dimension LP parameters in the current experiment.

Values: 'lpfft' or 'lparfft'
See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

lpalg LP algorithm in np dimension (P)

Number of increments in 1st indirectly detected dimension (P)

# 1palg2 LP algorithm in ni2 dimension (P)

Description: Specifies the LP (linear prediction) algorithm to use in the ni2 dimension.

lpalg2 functions analogously to lpalg. Enter addpar ('lp', 2) to create lpalg2 and other ni2 dimension LP parameters in the current experiment.

Values: 'lpfft' or 'lparfft'

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M

lpalg LP algorithm in np dimension (P)

Number of increments in 2nd indirectly detected dimension (P)

# lpext LP data extension in np dimension (P)

Description: Specifies number of complex time-domain data points for LP (linear prediction)

in the np dimension by which the original data is to be extended (or altered) in

either the forward or backward direction. 1pext is constrained by

(strtext-lpext) >= ≥0 for lpopt='b' and by (strtext+lpext-1) <=fn/2 for lpopt='f'. In the np direction, if (strtext-lpext) =0 and lpopt='b' (backwards linear prediction with calculation of the first point), fpmult defaults to the theoretical value of 0.5 instead of 1.0. Enter addpar('lp') to create lpext and other np dimension LP parameters in

the current experiment.

Related: addpar Add selected parameters to the current experiment (M)

lpalgLP algorithm in np dimension (P)lpext1LP data extension in ni dimension (P)lpext2LP data extension in ni2 dimension (P)

LP algorithm data extension in np dimension (P)

np Number of data points (P)

strtext Starting point for LP data extension in np dimension (P)

# lpext1 LP data extension in ni dimension (P)

Description: Specifies number of complex time-domain data points for LP (linear prediction)

in the ni dimension by which the original data is to be extended (or altered) in either the forward or backward direction. lpext1 functions analogously to lpext. Enter addpar ('lp', 1) to create lpext1 and other ni dimension

LP parameters in the current experiment.

Related: addpar Add selected parameters to the current experiment (M)

lpext LP data extension in np dimension (P)

Number of increments in 1st indirectly detected dimension (P)

### 1pext2 LP data extension in ni2 dimension (P)

Description: Specifies number of complex time-domain data points for LP (linear prediction)

in the ni2 dimension by which the original data is to be extended (or altered) in either the forward or backward direction. lpext2 functions analogously to

lpext. Enter addpar('lp',2) to create lpext2 and other ni2

dimension LP parameters in the current experiment.

Related: addpar Add selected parameters to the current experiment (M)

lpext LP data extension in np dimension (P)

Number of increments in 2nd indirectly detected dimension (P)

# lpfilt LP coefficients to calculate in np dimension (P)

Description: Specifies number of complex LP (linear prediction) coefficients in the np

dimension to be calculated from a specified region of the time-domain data. lpfilt should be greater than nsignals, where nsignals is the number

of sinusoidal signals contained in that FID (or interferogram). Enter

 ${\tt addpar}\,(\,{\tt 'lp'}\,)\,\,{\tt to}\,{\tt create}\,\,{\tt lpfilt}\,\,{\tt and}\,\,{\tt other}\,{\tt np}\,\,{\tt dimension}\,LP\,\,{\tt parameters}\,\,{\tt in}$ 

the current experiment.

Related: addpar Add selected parameters to the current experiment (M)

lpalg LP algorithm in np dimension (P)

lpfilt1 LP coefficients to calculate in ni dimension (P)
lpfilt2 LP coefficients to calculate in ni2 dimension (P)

np Number of data points (P)

#### lpfilt1 LP coefficients to calculate in ni dimension (P)

Description: Specifies number of complex LP (linear prediction) coefficients in the ni

dimension to be calculated from a specified region of the time-domain data. lpfilt1 functions analogously to lpfilt. Enter addpar ('lp', 1) to create lpfilt1 and other ni dimension LP parameters in the current

experiment.

Related: addpar Add selected parameters to the current experiment (M)

lpfilt LP coefficients to calculate in np dimension (P)

Number of increments in 1st indirectly detected dimension (P)

### 1pfilt2 LP coefficients to calculate in ni2 dimension (P)

Description: Specifies number of complex LP (linear prediction) coefficients in the ni2

dimension to be calculated from a specified region of the time-domain data. lpfilt2 functions analogously to lpfilt. Enter addpar('lp',2) to create lpfilt1 and other ni2 dimension LP parameters in the current

experiment.

Related: addpar Add selected parameters to the current experiment (M)

lpfilt LP coefficients to calculate in np dimension (P)

Number of increments in 1st indirectly detected dimension (P)

# 1pnupts LP number of data points in np dimension (P)

Description: Specifies number of complex time-domain data points in the np dimension to

be used in constructing the autoregressive (lpalg='lparfft') or least-

squares (lpalg='lpnefft') matrix from which the complex LP (linear prediction) coefficients are calculated. Note that lpnupts greater than or equal to 2\*lpfilt is required for both algorithms. Enter addpar ('lp') to create lpnupts and other np dimension LP parameters in the current experiment.

Related: addpar Add selected parameters to the current experiment (M)

> lpalq LP algorithm in np dimension (P)

lpfilt LP coefficients to calculate in np dimension (P) lpnupts1 LP number of data points in ni dimension (P) lpnupts2 LP number of data points in ni2 dimension (P)

Number of data points (P) np

#### lpnupts1 LP number of data points in ni dimension (P)

Description: Specifies number of complex time-domain data points in the ni dimension to

be used in constructing the autoregressive (lpalg1='lparfft') or leastsquares (lpalg1='lpnefft') matrix from which the complex LP (linear prediction) coefficients are calculated. lpnupts1 functions analogously to lpnupts. Enter addpar ('lp', 1) to create lpnupts1 and other ni

dimension LP parameters in the current experiment.

Related: addpar Add selected parameters to the current experiment (M)

> lpalq1 LP algorithm in ni dimension (P)

LP number of data points in np dimension (P) lpnupts

Number of increments in 1st indirectly detected dimension (P) ni

#### lpnupts2 LP number of data points in ni2 dimension (P)

Description: Specifies number of complex time-domain data points in the ni2 dimension to

be used in constructing the autoregressive (lpalq2='lparfft') or leastsquares (lpalq2='lpnefft') matrix from which the complex LP (linear prediction) coefficients are calculated. lpnupts2 functions analogously to lpnupts. Enter addpar('lp', 2) to create lpnupts2 and other ni2

dimension LP parameters in the current experiment.

Related: addpar Add selected parameters to the current experiment (M)

> lpalg2 LP algorithm in ni2 dimension (P)

1pnupts LP number of data points in np dimension (P)

ni2 Number of increments in 2nd indirectly detected dimension (P)

#### LP algorithm data extension in np dimension (P) lpopt

Specifies how the specific LP (linear prediction) algorithm is to extend (or alter) Description:

> forward or backward the time-domain data in the np dimension. Enter addpar ('lp') to create lpopt and other np dimension LP parameters in

the current experiment.

Multiple LP operations, extended forward or backward, can be performed on each FID or interferogram. This is accomplished by arraying the LP processing parameters (e.g., lpopt='b', 'f', 'b'). The number of LP operations is determined by the LP processing parameter with the largest array size. LP parameters having a smaller array size are padded out with their last value. The most common use for this capability is to back-calculate the first 1 to 2 points in an FID or interferogram and subsequently to extend the length of the time-

domain data by LP.

A printout can be obtained for each LP operation on an individually definable FID or interferogram. For example, if lpprint=30,30 and lptrace=1,2, the text file lpanalyz.out.1 contains the LP printout for the first LP operation on FID 1 and lpanalyz.out.2 contains the LP printout for the second LP operation on FID 2.

Values:

'b' indicates the LP coefficients are to be used in the back-calculation of a specified number of time-domain data points.

'f' indicates the LP coefficients are to be used in the forward extension of the time-domain data by a specified number of points. The characteristic polynomial in z space, derived from the complex LP coefficients, is set up and rooted. Any root found to lie outside the unit circle is reflected back into the unit circle. New complex LP coefficients are then calculated from these adjusted complex roots.

Related: addpar Add selected parameters to the current experiment (M)

lpalg LP algorithm in np dimension (P)

lpopt1 LP algorithm data extension for ni dimension (P)
lpopt2 LP algorithm data extension for ni2 dimension (P)

lpprint LP print output for np dimension (P)
lptrace LP output spectrum for np dimension (P)

np Number of data points (P)

# 1popt1 LP algorithm data extension in ni dimension (P)

Description: Specifies how the specific LP (linear prediction) algorithm is to extend (or alter)

forward or backward the time-domain data in the ni dimension. lpopt1 functions analogously to lpopt. Enter addpar ('lp', 1) to create lpopt1

and other ni dimension LP parameters in the current experiment.

 $Related: \quad \text{addpar} \qquad \quad Add \ selected \ parameters \ to \ the \ current \ experiment \ (M)$ 

LP algorithm data extension for np dimension (P)

Number of increments in 1st indirectly detected dimension (P)

#### 1popt2 LP algorithm data extension in ni2 dimension (P)

Description: Specifies how the specific LP (linear prediction) algorithm is to extend (or alter)

forward or backward the time-domain data in the ni2 dimension. lpopt2 functions analogously to lpopt. Enter addpar ('lp', 2) to create lpopt2

and other ni2 dimension LP parameters in the current experiment.

Related: addpar Add selected parameters to the current experiment (M)

lpopt LP algorithm data extension for np dimension (P)

Number of increments in 2nd indirectly detected dimension (P)

# 1pprint LP print output for np dimension (P)

Description: Controls LP (linear prediction) print output for the np dimension and creates an

output file in the current experiment directory (curexp) with the name lpanalyz.out.1. Enter addpar('lp') to create lpprint and other

np dimension LP parameters in the current experiment.

ues: Comprised of sum of decimal values of the following bit fields, in which each bit field controls an independent output option:

• Bit 0 (decimal value 1) writes out the LP matrix and Y vector from which the LP coefficients are calculated.

- Bit 1 (decimal value 2) writes out the LP coefficients that have been obtained using either of the two supported algorithms.
  - Bit 2 (decimal value 4) writes out the LP roots obtained from the characteristic polynomial derived from the LP coefficients; this only applies for lpalg='lpfft' and lpopt='f'.
  - Bit 3 (decimal value 8) writes out the original and recalculated values for each LP extended (or altered) complex time-domain data point.
  - Bit 4 (decimal value 16) writes out the internal LP parameter structure.

For example, lpprint=12 and lptrace=1 yields the following information in the file curexp/lpanalyz.out.1 for spectrum 1 along  $f_2$ : the values for all lpfilt complex LP coefficients and the original and recalculated values for each of the lpext LP extended (or altered) complex time-domain data points.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

curexpCurrent experiment directory (P)lpalgLP algorithm in np dimension (P)lpextLP data extension in np dimension (P)

lpfilt LP coefficients to calculate in np dimension (P)
lpopt LP algorithm data extension for np dimension (P)

lpprint1 LP print output for ni dimension (P)
lpprint2 LP print output for ni2 dimension (P)
lptrace LP output spectrum in np dimension (P)

np Number of data points (P)

#### 

Description: Controls LP (linear prediction) print output for the ni dimension and creates an

output file in the current experiment directory (curexp) with the name lpanalyz1.out.1.lpprint1 functions analogously to lpprint. Enter addpar('lp',1) to create lpprint1 and other ni dimension LP

parameters in the current experiment.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

lpprint LP print output for np dimension (P)

Number of increments in 1st indirectly detected dimension (P)

#### lpprint2 LP print output for ni2 dimension (P)

Description: Controls LP (linear prediction) print output for the ni2 dimension and creates

an output file in the current experiment directory (curexp) with the name lpanalyz2.out.1.lpprint2 functions analogously to lpprint. Enter addpar('lp',2) to create lpprint2 and other ni2 dimension LP

parameters in the current experiment.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

lpprint LP print output for np dimension (P)

Number of increments in 2nd indirectly detected dimension (P)

### 1ptrace LP output spectrum in np dimension (P)

Description: Specifies for which spectrum LP (linear prediction) output in the np dimension

is produced in accordance with the parameter lpprint. Enter

addpar ('lp') to create lptrace and other np dimension LP parameters

in the current experiment.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

np Number of data points (P)

# lptrace1 LP output spectrum in ni dimension (P)

Description: Specifies for which spectrum or trace LP (linear prediction) output in the ni

dimension is produced in accordance with the parameter lpprint1.

lptrace1 functions analogously to lptrace. Enter addpar ('lp', 1) to create t lpprint2 and other ni dimension LP parameters in the current

experiment.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

lpprint1 LP print output in ni dimension (P)
lptrace LP output spectrum in np dimension (P)

ni Number of increments in 1st indirectly detected dimension (P)

### lptrace2 LP output spectrum in ni2 dimension (P)

Description: Specifies for which spectrum or trace LP (linear prediction) output in the ni2

dimension is produced in accordance with the parameter lpprint2.

lptrace2 functions analogously to lptrace. Enter addpar ('lp', 2) to create lptrace2 and other ni2 dimension LP parameters in the current

experiment.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

lpprint2 LP print output in ni2 dimension (P)
lptrace LP output spectrum in np dimension (P)

Number of increments in 2nd indirectly detected dimension (P)

### 1s List files in directory (C)

Syntax: ls<(directory)>

Description: Lists the names of files in a directory on the text output window. 1s is identical

to dir and lf.

Arguments: directory is the name of a directory. The default is the current working

directory. 1s is equivalent to the UNIX command 1s and uses the same options

(e.g., -1 for a long listing such as ls('-1 \*.fid')).

Examples: ls

ls('data')
ls('-l \*.fid')

Related: dir List files in directory (C)

List files in directory (C)

# 1sfid Number of complex points to left-shift the np FID (P)

Description: Specifies number of complex points (not real points) that the np FID is to be

either left-shifted (lsfid>0) or right-shifted (lsfid<0). A right shift adds zeros to the front of the FID. lsfid (and related parameters phfid and lsfrq) operate on complex np FID data, referred to as the t<sub>2</sub> dimension in a 2D experiment or as the t<sub>3</sub> dimension in a 3D experiment. lsfid is in the processing group and is properly handled by a wti operation (display).

Values: -fn/2 to np/2 (or -fn/2 to fn/2 if fn < np), 'n'

Related: dfid Display a single FID (C)

ds Display a spectrum FID (C)

fn Fourier number in directly detected dimension (P)

ft Fourier transform 1D data (C)

Fourier transform along f<sub>2</sub> dimension (C)

ft2d Fourier transform 2D data (C)

lsfid1 Number of complex points to left-shift ni interferogram(P)

lsfid2 Number of complex points to left-shift ni2 interferogram (P)

lsfrq Frequency shift of the fn spectrum in Hz (P)

np Number of data points (P)

phfid Zero-order phasing constant for the np FID (P)
wft Weight and Fourier transform 1D data (C)
wftld Weight and Fourier transform f<sub>2</sub> of 2D data (C)
wftld Weight and Fourier transform 2D data (C)

wti Interactive weighting (C)

### 1sfid1 Number of complex points to left-shift ni interferogram (P)

Description: Specifies number of hypercomplex (for hypercomplex interferogram data) or

complex (for complex interferogram data) points that the ni interferogram is to be either left-shifted (lsfidl>0) or right-shifted (lsfidl<0). A right shift adds zeros to the front of the FID. lsfidl (and related parameters phfidl and lsfrql) operate on ni interferogram data, both hypercomplex and complex. ni interferogram data are referred to as the t<sub>1</sub> dimension in both a 2D and a 3D experiment. lsfidl is in the processing group and is properly handled by a wti operation (display); that is, a wti operation on an ni interferogram applies the parameters phfidl, lsfidl, and lsfrql, if selected, to the time-domain data prior to the Fourier transformation.

Values: -fn1/2 to ni (or -fn1/2 to fn1/2 if fn1<2\*ni), 'n'

Related: fn1 Fourier number in 1st indirectly detected dimension (P)

1sfid Number of complex points to left-shift np FID (P)

1sfid2 Number of complex points to left-shift ni2 interferogram (P)

lsfrq1 Frequency shift of the fn1 spectrum in Hz (P)

Number of increments in 1st indirectly detected dimension (P)

phfid1 Zero-order phasing constant for ni interferogram (P)

wti Interactive weighting (C)

#### 1sfid2 Number of complex points to left-shift ni2 interferogram (P)

Description: Specifies the number of hypercomplex (for hypercomplex interferogram data)

or complex (for complex interferogram data) points that the ni2 interferogram is to be either left-shifted (lsfid2>0) or right-shifted (lsfid2<0). A right shift adds zeros to the front of the FID. lsfid2 (and related parameters phfid2 and lsfrq2) operate on ni2 interferogram data, both hypercomplex and complex. ni2 interferogram data are referred to as the t<sub>2</sub> dimension in a 3D experiment. 1sfid2 is in the processing group and is properly handled by a

wti operation (display).

-fn2/2 to ni2 (or -fn2/2 to fn2/2 if fn2<2\*ni2), 'n' Values:

Related: fn2 Fourier number in 2nd indirectly detected dimension (P)

> lsfid Number of complex points to left-shift np FID (P)

Number of complex points to left-shift ni interferogram(P) 1sfid1

lsfrq2 Frequency shift of the fn2 spectrum in Hz (P)

Number of increments in 2nd indirectly detected dimension (P) ni2

phfid2 Zero-order phasing constant for ni2 interferogram (P)

wti Interactive weighting (C)

#### lsfrq Frequency shift of the fn spectrum (P)

Sets a frequency shift of spectral data, in Hz. 1sfrq is the time-domain

equivalent of 1p within VnmrJ. 1sfrq (and related parameters phfid and 1sfid) operate on complex np FID data, referred to as the t<sub>2</sub> dimension in a 2D experiment or as the t<sub>3</sub> dimension in a 3D experiment. 1sfrq is in the processing group and is properly handled by a wti operation (display).

Values: A positive value results in peaks being shifted downfield (to the left).

A negative value results in peaks being shifted upfield (to the right).

Related: dfid Display a single FID (C)

ds Display a spectrum FID (C)

Fourier number in directly detected dimension (P) fn

ft Fourier transform 1D data (C)

ft1d Fourier transform along f<sub>2</sub> dimension (C)

ft2d Fourier transform 2D data (C)

First-order phase in directly detected dimension (P) 1p lsfid Number of complex points to left-shift np FID (P) lsfrq1 Frequency shift of the fn1 spectrum in Hz (P) Frequency shift of the fn2 spectrum in Hz (P) lsfrq2 phfid Zero-order phasing constant for np FID (P) Weight and Fourier transform 1D data (C) wft wft1d Weight and Fourier transform f<sub>2</sub> of 2D data (C) Weight and Fourier transform 2D data (C)

wt i Interactive weighting (C)

#### Frequency shift of the fn1 spectrum (P) lsfrq1

wft2d

Description: Sets a frequency shift of spectral data, in Hz. lsfrq1 is the time-domain

equivalent of lp1 within VnmrJ. lsfrq1 (and related parameters phfid1 and lsfid1) operate on ni interferogram data, both hypercomplex and complex. ni interferogram data are referred to as the t<sub>1</sub> dimension in both a 2D and a 3D experiment. 1sfrq1 is in the processing group and is properly handled by a wti operation (display); that is, a wti operation on an ni

L

interferogram applies the parameters <a href="phfid1">phfid1</a>, <a href="lsfrq1">lsfid1</a>, and <a href="lsfrq1">lsfrq1</a>, if selected, to the time-domain data prior to the Fourier transformation.

Values: A positive value results in peaks being shifted downfield (to the left).

A negative value results in peaks being shifted upfield (to the right).

Related: fn1 Fourier number in 1st indirectly detected dimension (P)

lp1 First-order phase in 1st indirectly detected dimension (P)

lsfid1 Number of complex points to left-shift ni interferogram(P)

lsfrq Frequency shift of the fn spectrum in Hz (P)
lsfrq2 Frequency shift of the fn2 spectrum in Hz (P)

Number of increments in 1st indirectly detected dimension (P)

phfid1 Zero-order phasing constant for ni interferogram (P)

wti Interactive weighting (C)

#### 1sfrq2 Frequency shift of the fn2 spectrum (P)

ni

Description: Sets a frequency shift of spectral data in Hz. lsfrq2 is the time-domain

equivalent of lp2 within VnmrJ. lsfrq2 (and related parameters phfid2 and lsfid2) operate on ni2 interferogram data, both hypercomplex and complex. ni2 interferogram data is referred to as the t<sub>2</sub> dimension in a 3D experiment. lsfrq2 is in the processing group and is properly handled by a

wti operation (display).

Values: A positive value results in peaks being shifted downfield (to the left).

A negative value results in peaks being shifted upfield (to the right).

Related: fn2 Fourier number in 2nd indirectly detected dimension (P)

lp2 First-order phase in 2nd indirectly detected dimension (P)

lsfid1 Number of complex points to left-shift ni interferogram (P)

Number of complex points to left-shift ni2 interferogram (P)

lsfrq Frequency shift of the fn spectrum in Hz (P)

Number of increments in 2nd indirectly detected dimension (P)

phfid2 Zero-order phasing constant for ni2 interferogram (P)

wti Interactive weighting (C)

#### 1v1 Zero-order baseline correction (P)

Description: When spectral display is active, the command dc turns on a linear drift

correction (baseline correction). The result of this operation includes calculating a zero-order baseline correction parameter lvl. This is done by averaging of a small number of points at either end of the display and drawing

a straight line baseline between them.

Related: cdc Cancel drift correction (C)

lvltlt Control sensitivity of lvl and tlt adjustments (P)

tlt First-order baseline correction (P)

# lvltlt Control sensitivity of lvl and tlt adjustments (P)

Description: Controls the sensitivity of the interactive <a href="line">1vl</a> and <a href="tlt">tlt</a> adjustments. <a href="line">1vlt</a> tlt

is in the "current" parameter set and is basically a multiplier for the sensitivity.

If this parameter does not exist, it can be created by commands create('lvltlt') setgroup('lvltlt', 'display').

Values: The default value is 1.0. Larger values make the adjustments larger. Smaller

values make the adjustments smaller.

Related: create Create new parameter in a parameter tree (C)

ds Display a spectrum (C)

lvl Zero-order baseline correction (P)

# M

macro Macro name (P)

macrocat Display a user macro file in text window (C)

macrocp Copy a user macro file (C)
macrodir List user macro files (C)

macroedit Edit a macro with user-selectable editor (M)

macrold Load a macro into memory (C)
macrorm Remove a user macro (C)

macrosyscat Display a system macro file in text window (C)
macrosyscp Copy a system macro to become a user macro (C)

macrosysdir List system macros (C)
macrosysrm Remove a system macro (C)

macrovi Edit a user macro with the vi text editor (M)

make3dcoef Make a 3D coefficients file from 2D coefficients (M)

makedosyparamsCreate parameters for DOSY processing (M)makefidMake a FID element using numeric text input (C)makeeccglobalsCreate global parameters for ECC control (M)

makeslice Synthesize 2D projection of 3D DOSY experiment (C)
man Display online description of command or macro (M)

managedb Update user files (U)

manualpath Path to user's manual directory (P)

manvi Edit online description of a command or macro (M)

mapwin List of experiment numbers (P)

mark

Determine intensity of spectrum at a point (C)

masvt

Type of variable temperature system (P)

maxattench1-4 Maximum limit for attenuator setting for rf channel 1-4 (P)

maxpen Maximum number of pens to use (P)

md Move display parameters between experiments (C)

menu Change status of menu system (C)
menuvi Edit a menu with vi text editor (M)

method (P)

mf Move FIDs between experiments (C)

mfblk Copy FID block (C)

mfclose Close memory map FID (C)

mfdata Move FID data (C)

mfopen Memory map open FID file (C)

mftrace Move FID trace (C)

minsw Reduce spectral width to minimum required (M)

mkdir Create new directory (C)

mlabel Menu label (P)

move Move to an absolute location to start a line (C)

movedsswSet down sampling parameters for selected spectral region (M)moveosswSet over sampling parameters for selected spectral region (M)

movesw Move spectral window according to cursors (M)

# M

movetof Move transmitter offset (M)

Move parameters between experiments (C)

mqcosy
Set up parameters for MQCOSY pulse sequence (M)

mrev8
Set up parameters for MREV8 pulse sequence (M)

mrfb
Set the filter bandwidths for multiple receivers (P)

mref Set referencing based on a existing spectrum of the sample (M)

mrgain Set the gain for multiple receivers (P)
mstat Display memory usage statistics (C)

mstring Menu string (P)

mtune Tune probe using swept-tune graphical display (M)

mv Move and/or rename a file (C)
mxconst Maximum scaling constant (P)

# macro Macro name (P)

Description: A string parameter, available in each experiment, similar to the n1, n2, and n3

parameters. Certain macros, such as h1p, need to know which macro invoked

them. This parameter is used to pass that information.

See also: User Programming

Related: h1p Process simple proton spectra from h1 macro (M)

n1, n2, n3 Name storage for macros (P)

### macrocat Display a user macro file in text window (C)

Syntax: macrocat(file1<,file2><,...>)

Description: Displays one or more user macro files in the text window.

Arguments: file1, file2, ... are the names of macros in the user macro library.

Examples: macrocat('build')

macrocat('dan','george')

See also: User Programming

Related: macrodir List user macros (C)

macrosyscat Display a system macro file in text window (C)

# macrocp Copy a user macro file (C)

Syntax: macrocp(from file, to file)

Description: Makes a copy of the existing user macro file and places the copy in the user's

macro library. Using macrocp to make a backup copy is the recommended procedure to modify a macro but still be able to revert to the previous version if you are unsure about the modification. macrocp can also be useful for writing

a new macro that is very similar to an existing macro.

Arguments: from\_file is the name of an existing user macro file to be copied. The file

must be in the user's macro library.

to\_file is the file name to be given to the copy. This name must be different

from the name of the original macro.

Examples: macrocp('dan','dan.old')

See also: *User Programming* 

Related: macrocat Display a user macro file in text window (C)

macrodir List user macros (C)

macrosyscp Copy a system macro to become a user macro (C)

#### macrodir List user macro files (C)

Description: Lists the names of user macro files in the user's macro library.

See also: User Programming

Related: macrosysdir Lists system macros (C)

# macroedit Edit a macro with user-selectable editor (M)

Syntax: macroedit(file)

Description: Opens a MAGICAL macro file from a user's personal macro library for editing

(if you want to edit a system macro, copy it to a personal library and then use

macroedit).

The default editor is vi. To select another editor, first set UNIX environmental variable vnmreditor to the name of the editor; that is, in the .login file,

change the line

setenv vnmreditor old ed

to become

setenv vnmreditor new ed (e.g., setenv vnmreditor emacs).

Second, make sure a script with the prefix vnmr\_followed by the name of the editor is placed in the bin subdirectory of the VnmrJ system directory (e.g.,

vnmr emacs).

The script file makes adjustments for the type of graphic interface in use. Scripts provided in the software include <code>vnmr\_vi</code> and <code>vnmr\_textedit</code>. To create other scripts, refer to the <code>vnmr\_vi</code> script for non-window editor interfaces or

refer to vnmr textedit for window-based editor interfaces.

Arguments: file is the name of the macro file you wish to edit.

Examples: macroedit('pa')

See also: *User Programming* 

Related: paramedit Edit a parameter and its attributes with user-selected editor (C)

Edit a parameter and its attributes with *vi* editor (M)

edit Edit a file with user-selectable editor (C)
macrovi Edit a user macro with vi editor (M)
menuvi Edit a menu with the vi editor (M)

Edit text file of current experiment with vi editor (M)

# macrold Load a macro into memory (C)

Syntax: macrold(file)<:dummy>

Description: Loads a macro, user or system, into memory. If the macro already exists in

memory, it is overwritten by the new macro. Loading a macro into memory increases the execution speed of the macro. The trade-off is that the macro uses memory. The mstat command displays macros that have been loaded into memory. One or more individual macros, or all the macros loaded in memory,

can be removed from memory with the purge command.

If a macro already loaded into memory is edited using macrovi or

macroedit, the changed macro automatically is loaded by those macros. This

overwrites the previous macro. However, if a macro is edited or created some other way (with macrocp perhaps), the changed version is not automatically loaded. If the macro already exists in memory, the previous version executes

unless the user runs macrold.

Arguments: file is the name of the macro file to be loaded into memory. For loading

> macros, the same search path is used as when deciding which macro to execute. That is, the user's private maclib directory is searched first and finally the system maclib. If an absolute path is supplied as the file argument, that macro is loaded. This allows macros not in a maclib to be loaded and executed

from VnmrJ.

dummy is any throwaway variable. Requesting a return value suppresses the

message in the status window (line 3) that the macro is loaded.

Examples: macrold('pa')

macrold('\_sw'):\$noline3

See also: User Programming

Related: Copy a user macro file (C) macrocp

> macroedit Edit a macro with user-selectable editor (M) Edit a user macro with the vi text editor (M) macrovi

mstat Display memory usage statistics (C) Remove macros from memory (C) purge

#### Remove a user macro (C) macrorm

Syntax: macrorm(file)

Description: Removes a user macro from the user's macro directory. If the macro has already

been loaded in memory, it remains in memory until a new macro of the same

name is loaded or the program exits.

Arguments: file is the name of the user macro to be removed.

Examples: macrorm('pa') See also: User Programming

Related: delcom Delete a user macro (M)

> List user macros (C) macrodir macrosysrm Remove a system macro (C)

purge Remove all macros from memory (C)

#### Display a system macro file in text window (C) macrosyscat

Syntax: macrosyscat(file1<,file2><,...>)

Description: Displays one or more system macro files in the text window.

Arguments: file1, file2, ... are names of macros in the system macro library.

Examples: macrosyscat('build')

macrosyscat('dan','george')

See also: User Programming

Related: Display a user macro file in text window (C) macrocat

macrosysdir Lists system macros (C)

#### macrosyscp Copy a system macro to become a user macro (C)

Syntax: macrosyscp(from file, to file)

Description: Makes a copy of the existing system macro file and places the copy in the user's

macro library. This is the recommended way to modify a system macro for

personal use.

Arguments: from file is the name of an existing system macro file to be copied. The file

must be in the system macro library.

to\_file is the file name to be given to the copy. In this case, the name of the copied macro can be the same as the original macro. In many cases, it is the same, allowing the user to have a personal macro of the same name as the

system macro but which will override the system macro.

Examples: macrosyscp('pa','pa')

macrosyscp('pa','mypa')

See also: User Programming

Related: macrocp Copy a user macro file (C)

macrosyscat Display a system macro file in text window (C)

macrosysdir Lists system macros (C)

macrosysdir List system macros (C)

Description: Lists the names of system macros in the system macro library.

See also: User Programming

Related: macrodir List user macros (C)

macrosysrm Remove a system macro (C)

Syntax: macrosysrm(file)

Description: Removes a system macro file from the system macro directory. If the macro has

already been loaded in memory, it remains in memory until a new macro of the

same name is loaded or the program exits.

Arguments: file is the name of the system macro file to be removed.

Examples: macrosysrm('pa')

See also: User Programming

Related: macrorm Remove a user macro (C)

macrosysdir Lists system macros (C)

purge Remove all macros from memory (C)

macrovi Edit a user macro with the vi text editor (M)

Syntax: macrovi(file)

Description: Initiates creating a new user macro or modifying an existing user macro using

the UNIX vi text editor. On the Sun workstation, a pop-up window contains the edit. On the GraphOn, the edit is done on the entire terminal. To edit a system macro, first copy the macro to a personal library and then edit it using

macroedit or macrovi.

Arguments: file is the name of an existing user's macro to be edited or the name of a new

user's macro to be created.

Examples: macrovi('pa')

See also: User Programming

Related: macroedit Edit a macro with a user-selectable editor (C)

vi Edit text file with vi text editor (C)

#### Make a 3D coefficients file from 2D coefficients (M) make3dcoef

Syntax: make3dcoef<('t1t2'|'t2t1')>

Description: Makes a 3D coefficients file from 2D coefficients and writes the file in the path stored by curexp. 2D coefficients are supplied as strings in the parameters f2coef and f1coef. This macro is capable of handling 3D data collected with any number of data sets (e.g., TPPI, Hypercomplex, Rance SE, Kay SE, and phase-sensitive gradient in one or both dimensions). make3dcoef is called by the ft3d macro.

> The 2D coefficients are supplied as strings in flcoef and flcoef. These coefficients are the same as found by processing with wft2d (2dcoefs). Note that wft2da (for States-Hypercomplex method) is equivalent to wft2d(1,0,0,0,0,0,-1,0), and that wft2d (for absolute-value mode) is equivalent to wft2d(1,0,0,-1).

> Coefficients are separated by spaces and not commas. For example, if a 3D data set collected by the States-Hypercomplex method in both ni and ni2 dimensions, f1coef='100000-10' and f2coef='100000-10'. And if a 3D data set collected in absolute-value mode in both ni and ni 2 dimensions, flcoef='100-1' and f2coef='100-1'.

> The flcoef and flcoef parameters are created by the par3d macro. Execution of make3dcoef when floof and floof have no value or inconsistent values causes the macro to abort, which enables the user to enter these values and reexecute the macro. For example, the value of flcoef when the F1 dimension can be processed with wft2da is '100000-10'. The value of f2coef when the F2 dimension can be processed with wft2d(1,0,1,0,0,-1,0,1) is '10100-101'.

> The parameters floof and floof must be 2D coefficients that give proper ni and ni2 first planes with the same rp (assuming lp is 0 by using calfa) values. For example, processing the phase-sensitive gradient dimension should not be done with 1 0 0 1 0 1 1 0 and applying 45° phase shifts to rp, but with 1 0 1 0 0 1 0 -1, or its variant, that gives the same rp value as the other dimension. This also applies to Rance-type or Kay-type sensitivity-enhanced dimensions.

Note that sensitivity-enhanced sequences (gradient or otherwise) can be processed two different ways to give "orthogonal" data sets. The coefficients must be picked so that they have the same rp as the other dimension.

This macro can also handle coefficients that are not 1s or 0s. For example, if processing requires that a data set contributes to the interferogram after a 30° phase shift, cos(30) and sin(30) can be selected as the real and imaginary contributions, respectively, during the construction of the interferogram.

Arguments:

't1t2' means array='phase, phase2' in simple hypercomplex data sets. It means array='t1related', 't2related' with multiple sets in general.

't2t1' means array='phase2, phase' in simple hypercomplex data sets. It means array='t2related', 't1related' with multiple sets in general.

If no argument is used and if array='phase, phase2' or array= 'phase2, phase, the macro automatically decides on 't1t2' or 't2t1', respectively.

See also: NMR Spectroscopy User Guide

Related: array Parameter order and precedence (P)

> calfa Recalculate alfa so that first-order phase is zero (M)

curexp Current experiment directory (P)

f1coef Coefficient to construct F1 interferogram (P) f2coef Coefficient to construct F2 interferogram (P)

ft3d	Perform a 3D Fourier transform on a 3D FID data set (M)
lp	First-order phase in directly detected dimension (P)
ni	Number of increments in 1st indirectly detected dimension (P)
ni2	Number of increments in 2nd indirectly detected dimension (P)
ntype3d	Specify whether f <sub>1</sub> or f <sub>2</sub> display expected to be N-type (P)
rp	Zero-order phase in directly detected dimension (P)
wft2d	Weight and Fourier transform 2D data (C)
wft2da	Weight and Fourier transform phase-sensitive data (M)

# makedosyparamsCreate parameters for DOSY processing (M)

Syntax: makedosyparams(dosytimecubed,dosyfrq)

Description: This macro is automatically called by the Dbppste, DgcsteSL, Doneshot,

Dbppsteinept, Dgcstecosy, and Dgcstehmqc sequences to create the parameters dosyfrq, dosygamma, and dosytimecubed, which are necessary for the

dosy analysis. Do not manually run makedosyparams.

See also: NMR Spectroscopy User Guide

Related: dosy Process DOSY experiments (M)

dosyfrqLarmor frequency of phase encoded nucleus in DOSY (P)dosygammaGyromagnetic constant of phase encoded nucleus in DOSY (P)dosytimecubedGyromagnetic constant of phase encoded nucleus in DOSY (P)

# makefid Make a FID element using numeric text input (C)

Syntax: makefid(file<,element\_number<,format>)

Description: Creates FID files that can be used to introduce computed data into an

experiment. The number of points comes from the number of numeric values read from the input file. If the current experiment already contains a FID, you will not be able to change either the format or the number of points from that present in the FID file. Use rm(curexp+'/acqfil/fid') to remove the

FID.

The makefid command does not look at parameter values when establishing the format of the data or the number of points in an element. Thus, if the FID file is not present, it is possible for makefid to write a FID file with a header that does not match the value of dp or np. Because the active value is in the processed tree, you need to use the setvalue command if any changes are

required.

Arguments: file is the name of the input file. It contains numeric values, two per line. The

first value is assigned to the X (or real) channel; the second value on the line is

assigned to the Y (or imaginary) channel.

element\_number is the number of the element or FID and is any integer larger than 0. The default is the first element or FID. If the FID element already exists in the FID file, the program overwrites the old data.

format is a character string with the precision of the resulting FID file and can be specified by one of the following strings:

'dp=n' single-precision (16-bit) data
'dp=y' double-precision (32-bit) data
'16-bit' single-precision (16-bit) data
'32-bit' double-precision (32-bit) data

If an FID file exists, makefid uses the same format string for precision; otherwise, the default is double-precision (32-bit) data.

element number and format arguments can be entered in any order.

Examples: makfid('fid.in',2,'32-bit')

See also: NMR Spectroscopy User Guide; User Programming

Related: cp Copy a file (C)

curexp Current experiment directory

dp Double precision (P)

mv Move and/or rename a file (C)
np Number of data points (P)

rm Delete file (C)

setvalue Set value of any parameter in a tree (C)
writefid Write numeric text file using a FID element (C)

# makeeccglobals Create global parameters for ECC control (M)

Applicability: Systems with Varian, Inc. Cold Probes

Description: Creates the following nine global parameters required for ECC control by PSG:

tc1z, tc2z, tc3z, tc4z, amp1z, amp2z, amp3z, amp4z, and

chiliConf

Related: chiliConf

# makeslice Synthesize 2D projection of 3D DOSY experiment (C)

Syntax: makeslice(<option>,lowerlimit,upperlimit)

Arguments: option is either 'i' or 's'.

'i' includes the "tails" of diffusion peaks that lie outside the range between

lowerlimit and upperlimit. The default is 'i'.

's' only includes the integration peaks whose diffusion coefficient lies

between the specified limits.

lowerlimit is the lower diffusion limit (in units of  $10^{-10}$  m<sup>2</sup>/s) to be

displayed.

upperlimit is the upper diffusion limit (in units of  $10^{-10}$  m<sup>2</sup>/s) to be

displayed.

Description: Synthesizes an integral projection between specified diffusion limits of a

3D DOSY spectrum onto the frequency-frequency plane. makeslice

requires the first 2D increment of the 3D DOSY data to have been transformed.

See also: NMR Spectroscopy User Guide

Related: dosy Process DOSY experiments (M)

showoriginal Restore first 2D spectrum in 3D DOSY spectrum (M)

# man Display online description of command or macro (M)

Syntax: man('file')<:\$return>

Displays a description of commands and macros from files in the applications directory. The manual file is displayed in the text window when it is retrieved by the man macro. The man macro aborts if a name is not supplied as an

argument.

Arguments: file — name of a command or macro in one of the applications directories.

: \$res — supply a return argument to suppress messages if the manual page

does not exist.

Examples: man('mark')

man('notAcommand'):\$res

See also: NMR Spectroscopy User Guide; User Programming

Related: manvi Edit online description of a command or macro (M)

manualpath Path to user's manual directory (P)

managedb Update user files (U)

Syntax: managedb update

Description: Updates VnmrJ database for the Locator.

See also: NMR Spectroscopy User Guide

manualpath Path to user's manual directory (P)

Description: Contains the absolute path to a user's directory of VnmrJ manual entries. If

manualpath exists for a user, it must be defined in the user's global parameter file. Enter create ('manualpath', 'string', 'global') to create

the manualpath parameter.

See also: User Programming

Related: man Display online description of a command or macro (M)

manvi Edit online description of a command or macro (M)

Syntax: manvi('file')

Description: Enables editing or creating an online description of commands and macros

stored in any of the applications directories for to which the user has write

permission.

Arguments: file is the name of a command macro.

Examples: manvi('mark')
See also: User Programming

Related: man Display online description of command or macro (M)

mapwin List of experiment numbers (P)

Description: Arrayed global parameter that maintains a list of experiment numbers for the

window panes in the VnmrJ graphics window.

Related: curwin Current window (P)

fontselectOpen FontSelect window (C)jwinActivate current window (M)setgridActivate selected window (M)setwinActivate selected window (C)

mark Determine intensity of spectrum at a point (C)

Syntax: (1) mark<(f1\_position)><:intensity>

(2) mark<(left\_edge,region\_width)><:intensity,
 integral>

(3) mark<(f1\_position,f2\_position)><:intensity>

(4) mark<(f1\_start,f1\_end,f2\_start,f2\_end) >
 <:intensity,integral,c1,c2>

```
(5) mark<('trace',<options>)><:intensity,integral,
   c1,c2>
```

(6) mark('reset')

### Description:

Find the intensity of a spectrum at a point. Either 1D or 2D operations can be performed in the cursor or box mode for a total of four separate functions: 1D operations in cursor mode (syntax 1), 1D operations in box mode (syntax 2), 2D operations in cursor mode (syntax 3) and 2D operations in box mode (syntax 4).

In the *cursor mode*, the intensity at a particular point is found. In the *box mode*, the integral over a region is calculated. The displayed integral is scaled in the same way as output from dli is scaled; that is, by the ins and insref parameters. For 2D operations, this is the volume integral and the volume is scaled by ins2 and ins2ref. In addition, the mark command in the box mode finds the maximum intensity and the coordinate(s) of the maximum intensity.

The mark command requires that transformed data be present in the current experiment. If required, it recomputes the phase file from the complex data (i.e., it rephases the data if required); however, the mark command requires parameters from the command line if no data is displayed (i.e., if ds or dconi has not been executed).

Note that 2D operations require that 2D data be present. This not only means that ni must be larger than 1, but also that the data was transformed using ftld, ftld or an equivalent (and not ft or its equivalents).

The mark command, as well as the MARK button of ds, writes output to a file in the current experiment. For 1D operations, the file is named mark1d.out; for 2D operations, it is mark2d.out. If this file already exits, VnmrJ appends output from the current mark operation to the end of the file. (Older versions of VnmrJ used ds.out and dconi.out as files for output from the MARK button). Either file can be read by other programs at any time between operations.

The following criteria establish the exact function. The command checks them in the following order until it determines the exact function:

- 1. Number of numeric parameters.
- 2. Number of return values called out.
- 3. Which display command (ds or dconi) was last used.
- 4. Nature of the data in the experiment.

The first two criteria only serve to distinguish between box mode and cursor mode. The nature of the data in the experiment and the last display command entered determines whether a 1D or a 2D operation is selected.

Arguments: f1 position defines the position, in Hz, along the f<sub>1</sub> axis in the 1D and 2D cursor modes. The default is **cr** (1D) or **cr1** (2D).

> left edge defines the position of the left edge of the region, in Hz, to be integrated in 1D box mode. The default is cr.

region width defines the width, in Hz, of the region, which extends to the right of left edge, in 1D box mode. The default is delta.

f2 position defines the position, in Hz, along the f<sub>2</sub> axis in the 2D cursor mode. The default is delta1.

f1 start and f1 end define region along the  $f_1$  axis in the 2D box mode.

f2 start and f2 end define region along the  $f_2$  axis in the 2D box mode.

'trace' is a keyword to select a 1D operation if 2D data is present. It must be either the first or the last argument (e.g., mark ('trace', 400) determines the intensity at 400 Hz in the current trace).

'reset' is a keyword to erase the output files from the mark command. No other argument can be used with this keyword. Use rename to rename the current mark output files (e.g., rename (curexp+'/mark1d.out', curexp+'/mark.16.01.89')

intensity is a return value set to the intensity of the spectrum at the point for either 1D or 2D operations (the maximum if cursor mode was selected).

integral is a return value set to the integral of the spectrum at the point. integral is not returned in the cursor mode.

c1, c2 are return values set to the coordinates where the maximum intensity was found in 2D mode. c1 and c2 are not returned in the cursor mode.

#### Examples: 1D data sets:

```
mark(cr) cursor mode for 1D data
mark(cr,delta) box mode for 1D data
```

2D data sets (2D mode): In this mode, the order of the arguments to mark is independent of the trace parameter.

```
mark(cr1,cr) cursor mode for 2D data
mark(cr1,delta1,cr,delta) box mode for 2D data
```

2D data sets (1D mode): In this mode, the selection of the arguments to mark is dependent on the trace parameter. If trace='f2', then cr, delta, sp, or wp are appropriate. If trace='f1', then cr1, delta1, sp1, and wp1 are appropriate.

```
mark('trace',cr) cursor mode for selected 2D trace
mark('trace',cr1,delta1) box mode for selected 2D trace
```

Alternate: MARK button in the ds program.

See also: NMR Spectroscopy User Guide; User Programming

Related: cr Cursor position in directly detected dimension (P)

Cursor position in 1st indirectly detected dimension (P)

curexp Current experiment directory (P)
dconi Interactive 2D contour display (C)
delta Difference of two frequency cursors (P)

dli Display list of integrals (C)
ds Display a spectrum (C)

ft1d Fourier transform along f<sub>2</sub> dimension (C)

Fourier transform 2D data (C)
ins Integral normalization scale (P)

ins2 2D volume value (P)

insref Fourier number scaled value of an integral (P)
ins2ref Fourier number scaled volume of a peak (P)

Move and/or rename a file (C)

Number of increments in 1st indirectly detected dimension (P)

#### masvt Type of variable temperature system (P)

#### Description:

Identifies the type of VT system in use: the standard Oxford VT controller or the Oxford-Sorenson or solids VT controller system (used with the Varian VT CP/MAS probe). masvt is a global parameter that is active on all of each user's experiments on a per user account basis. The current value of the parameter can be displayed by typing masvt?.

Note that the VT Controller option displayed by config must be set to Present for either VT controller system to be active. If masvt does not exist, it can be created with the command create ('masvt', 'string', 'global').

The new Highland VT controller is autosensing, making masvt superfluous for systems with this controller.

Values: 'y' indicates the solids VT system is in use.

'n', any other value but 'n' and 'y', or if masvt does not exist, indicate

that the Oxford Varian VT controller, if present, is in use.

See also: VnmrJ Installation and Administration

Related: config Display current configuration and possibly change values (M)

create Create a new parameter in a parameter tree (C)
vttype Variable temperature controller present (P)

#### maxattench1-4 Maximum limit for attenuator setting for rf channel 1-4 (P)

Description: maxattench1, maxattench2, maxattench3, and maxattench4, are

optional global parameters for the limiting the maximum attenuator settings for rf channel 1, channel 2, channel 3, and channel 4 (respectively) from pulse sequence statements and through tpwr/dpwr/... settings on go command. If maxattench2 is present, the attenuator setting check will be carried out by SpinCAD and C psg. If the attenuator setting exceeds the limit set in

maxattench2, psg aborts with error message. This command is only applicable for check during the go command.

See also: SpinCAD

maxpen Maximum number of pens to use (P)

Description: Controls the maximum number of pens that will be used.

Values: 1 to the number of pens in the system plotter. If maxpen=x and the software

attempts to use pen x+y, it uses pen y instead.

See also: NMR Spectroscopy User Guide

Related: pen Select a pen or color for drawing (C)

setpen Set maximum number of HP plotter pens (M)

md Move display parameters between experiments (C)

Syntax: md(<from exp,>to exp)

Description: Moves the saved display parameters from one experiment to another. These

parameters must have been saved with the s command (e.g., s2).

Arguments: from exp specifies the number of the experiment, 1 through 9, from which

the parameters are to be taken. The default is that the parameters are moved

from the currently active experiment.

to exp specifies to which experiment the parameters are to be moved.

Examples: md(4)

md(2,3)

See also: NMR Spectroscopy User Guide

Related: mf Move FIDs between experiments (C)

mp Move parameters between experiments (C) s Save display parameters as a set (M)

#### menu Change status of menu system (C)

Syntax: (1) menu (menu\_name)

(2) menu< ('off') >

Description: The VNMR menu system allows up to eight buttons to be active at a time,

enabling the user to perform most actions with the mouse rather than typing in commands. All menus are stored in the library menulib in the system directory or in the user's menulib. See menuvi to change these menus.

If the menu system becomes deactivated for some reason, select the Menu On button in the Permanent Menu to reactivate it. Entering menu ('main') also

works.

Arguments: menu name is the name of the file controlling the menu (e.g., 'main').

Including this argument activates the menu system and displays the menu

controlled by menu name.

'off' is a keyword to turn off the menu system.

Examples: menu

menu('fitspec')
menu('off')

See also: User Programming

Related: menuvi Edit a menu with the vi text editor (M)

mlabel Menu label (P)

newmenu Select a menu without immediate activation (C)

### menuvi Edit a menu with vi text editor (M)

Syntax: menuvi(menu)

Description: Edits a Classic VNMR menu file using the UNIX vi text editor. On the Sun

workstation, a pop-up window contains the edit. On the GraphOn, the edit is

done on the entire terminal.

Arguments: menu is the name of file controlling a menu.

Examples: menuvi('display 1D')

See also: User Programming

Related: menu Change status of menu system (C)

newmenu Select a menu without immediate activation (C)

vi Edit text file with vi text editor (C)

### method Autoshim method (P)

Description: Selects the method for automatic shimming. Refer to the manual *NMR* 

Spectroscopy User Guide for information on how to write or alter methods.

Values: Name of file in the /vnmr/shimmethods library for one of the defined shim

methods in the system. To display all available methods, enter ls('/vnmr/shimmethods'). Standard methods include 'z1z2' (selects shimming of the Z1 and Z2 gradients) and 'allzs' (selects shimming all spinning gradients, Z1 to Z4 or Z5, depending on the magnet type). Shim methods can also be stored in a user's shimmethods directory (e.g., /home/vnmr1/

vnmrsys/shimmethods).

See also: NMR Spectroscopy User Guide

Related: 1s List files in current directory (C)

newshm Interactively create a shim method with options (M)

stdshm Interactively create a shim method (M)

# mf Move FIDs between experiments (C)

Syntax: mf(<from exp,>to exp)

Description: Moves the last acquired FID, as well as its associated parameters, from one

experiment to another. The text, the processed acquisition parameters and the current display and processing parameters are also moved to the specified

experiment.

Arguments: from exp specifies number of the experiment from which the FID is to be

taken. The default is the FID is moved from the currently active experiment.

to exp specifies to which experiment the FID is to be moved.

Examples: mf(4)

mf(3,2)

See also: NMR Spectroscopy User Guide

Related: md Move display parameters between experiments (C)

mp Move parameters between experiments (C)

# mfblk Copy FID block (C)

Syntax: mfblk(<src\_expno,>src\_blk\_no,dest\_expno,dest\_blk\_no)

Description: Copies data from a source FID block specified by src blk no to a

 ${\tt destination} \ FID \ {\tt block} \ {\tt specified} \ {\tt by} \ {\tt dest\_expno} \ {\tt and} \ {\tt dest\_blk\_no}, using$ 

memory-mapped input and output.

mfblk searches for the source and destination FID file in the directory \$vnmruser/expN/acqfil, where N is the requested experiment number or the current experiment number. If the FID file is not open, mfblk opens the file, copies the data, and closes the file. If a number of blocks need to be copied, explicitly opening and closing the files with the commands mfopen and mfclose can significantly speed up the data reformatting process.

mfblk can also be used to append blocks of data to a FID file by specifying that the dest blk no is greater than the number of blocks in a file.

Be aware that mfblk can modify data returned to an experiment with the rt command. To avoid modification, enter the following sequence of VnmrJ commands before running mfblk:

```
cp(curexp+'/acqfil/fid',curexp+'/acqfil/fidtmp')
rm(curexp+'/acqfil/fid')
mv(curexp+'/acqfil/fidtmp',curexp+'/acqfil/fid')
```

Arguments: src\_expno specifies the experiment number of the source FID file. The

default is the FID file of the current experiment.

src\_blk\_no specifies the source block of data to be copied. Block numbers start at 1 and run from 1 to the number of blocks in a file.

dest\_expno specifies the experiment number of the destination FID file.

dest\_blk\_no specifies the destination block to send the copied data.

Examples: mfblk (1,2,1) copies current experiment, block 1 to exp 2, block 1.

mfblk (3, 2, 6, 2) copies exp 2, block 2 to exp 6, block 2.

See also: User Programming

Related: mfclose Memory map close FID file (C)

mfdata Move FID data (C)

mfopen Memory map open FID file (C)

mftrace Move FID trace (C)

#### mfclose Close memory map FID (C)

Description: Closes experiment source and destination FID files that have been explicitly

opened with mfopen.

See also: User Programming

Related: mfblk Move FID block (C)

mfdata Move FID data (C)

mfopen Memory map open FID file (C)

mftrace Move FID trace (C)
rfblk Reverse FID block (C)
rfdata Reverse FID data (C)
rftrace Reverse FID trace (C)

# mfdata Move FID data (C)

Description: Copies data specified by src start loc from a FID block specified by

src\_blk\_no to a destination location specified by dest\_expno,
dest\_blk\_no, and dest\_start\_lo, using memory-mapped input and
output. The data point locations and the num\_points to be copied are
specified by data points corresponding to the np parameter, not bytes or

complex points.

mfdata searches for the source and destination FID file in th directory \$vnmruser/expN/acqfil, where N is the requested experiment number or the current experiment number. If the FID file is not open, mfdata opens the file, copies the data, and closes the file. If a number of blocks need to be copied, explicitly opening and closing the files with the commands mfopen and mfclose can significantly speed up the data reformatting process.

Be aware that mfdata can modify data returned to an experiment with the rt command. To avoid modification, enter the following sequence of VnmrJ commands before running mfdata:

```
cp(curexp+'/acqfil/fid',curexp+'/acqfil/fidtmp')
rm(curexp+'/acqfil/fid')
mv(curexp+'/acqfil/fidtmp',curexp+'/acqfil/fid')
```

Arguments:

src\_expno specifies the experiment number of the source FID file. The default is the FID file of the current experiment.

src\_blk\_no specifies the source block of data to be copied. Block numbers start at 1 and run from 1 to the number of blocks in a file.

src\_start\_loc specifies the starting data location within the specified block to copy the data. Data locations start from 0 and are specified as data points corresponding to the np parameter.

dest\_expno specifies the experiment number of the destination FID file. dest\_blk\_no specifies the destination block to send the copied data.

dest\_start\_loc specifies the starting data destination location within the specified block to send the copied data.

Examples: mfdata(1,0,2,1,(nv-1)\*np,np) copies np points of data from the

starting location 0 of block 1 of the current experiment to the data location

(nv-1) \*np of block 1 of experiment 2.

See also: User Programming

Related: mfblk Move FID block (C)

mfclose Memory map close FID file (C)

mfdata Move FID data (C)

mfopen Memory map open FID file (C)

mftrace Move FID trace (C) rfblk Reverse FID block (C) rftrace Reverse FID trace (C)

#### mfopen Memory map open FID file (C)

Syntax: mfopen<(<src expno,>dest expno)>

Explicitly opens experiment source and destination FID files for using memory-Description:

mapped input and output. Opening a file explicitly can significantly speed up

the data reformatting process.

mfopen searches for the FID file to be opened in the directory \$vnmruser/ expN/acqfil, where N is the requested experiment number or the current experiment number. Without arguments, mfopen assumes the source and destination files are the same and are in the current experiment.

After a file is open, the data reformatting commands mfblk, mfdata, mftrace, rfblk, rfdata, and rftrace can be used for moving around data. The mfclose must be used to close the file when data reformatting has been completed.

Arguments: src expno specifies the experiment number of the source FID file. The

default is the FID file of the current experiment.

dest expno specifies the experiment number of the destination FID file. The

default is the FID file of the current experiment.

If only one argument is provided, mfopen uses that as the experiment number of the destination FID file and assumes the source is the FID file of the current

experiment.

Examples: mfopen

> mfopen(3) mfopen(1,2)

See also: User Programming

Related: mfblk Move FID block (C)

> mfclose Memory map close FID file (C)

Move FID data (C) mfdata mftrace Move FID trace (C) rfblk Reverse FID block (C) Reverse FID data (C) rfdata rftrace Reverse FID trace (C)

#### mftrace Move FID trace (C)

Syntax: mftrace(<src expno,>src blk no,src trace no,

dest expno, dest blk no, dest trace no)

Description: Copies FID traces specified by src trace no from a FID block specified by

src blk no to a destination location specified by dest expno,

dest blk no, and dest trace no, using memory-mapped input and output. If a number of blocks need to be copied, explicitly opening and closing the files with the commands mfopen and mfclose can significantly speed up

the data reformatting process.

mftrace searches for the source and destination FID file in the directory \$vnmruser/expN/acqfil, where N is the requested experiment number or the current experiment number. If the FID file is not open, mftrace opens the

file, copies the data, and closes the file.

mftrace cannot be used to append data to a FID file. Its purpose is for moving around data.

Be aware that mftrace can modify data returned to an experiment with the rt command. To avoid modification, enter the following sequence of VnmrJ commands before running mftrace:

cp(curexp+'/acqfil/fid',curexp+'/acqfil/fidtmp')
rm(curexp+'/acqfil/fid')
mv(curexp+'/acqfil/fidtmp',curexp+'/acqfil/fid')

Arguments: src\_expno specifies the experiment number of the source FID file. The

default is the FID file of the current experiment.

src\_blk\_no specifies the source block of data to be copied. Block numbers

start at 1 and run to the number of blocks in a file.

src\_trace\_no specifies the source trace of data within the specified block

to be copied. Trace numbers run from 1 to number of traces in a file.

 ${\tt dest\_expno}\ specifies\ the\ experiment\ number\ of\ the\ destination\ FID\ file.$ 

dest\_blk\_no specifies the destination block to send the copied data.

src\_trace\_no specifies the destination trace of data within the specified block to be copied. Trace numbers run from 1 to the number of traces in a file.

Examples: mftrace(1,1,2,1,nv) copies trace 1 from block 1 of the current

experiment to trace nv of block 1 of experiment 2.

See also: User Programming

Related: mfblk Move FID block (C)

mfclose Memory map close FID file (C)

mfdata Move FID data (C)

mfopen Memory map open FID file (C)

rftrace Reverse FID trace (C)
rfblk Reverse FID block (C)
rfdata Reverse FID data (C)

# minsw Reduce spectral width to minimum required (M)

Description: Searches the spectrum for peaks, sets new limits accordingly, and then calls

movesw to calculate a new transmitter offset tof and spectral width sw.

See also: NMR Spectroscopy User Guide

Related: movesw Move spectral window according to cursors (M)

movetof Move transmitter offset (M)

Sw Spectral width in directly detected dimension (P) tof Frequency offset for transmitter offset (P)

# mkdir Create new directory (C)

Syntax: mkdir(directory)

Description: Creates a new UNIX directory. The function of the VnmrJ mkdir command is

similar to the UNIX mkdir command.

Arguments: directory is the name of the new directory to be created.

Examples: mkdir('tests')

mkdir('/home/george')

See also: NMR Spectroscopy User Guide

Related: rmdir Remove directory (C)

mlabel Menu label (P)

Description: Stores the label for a menu button. Usually this parameter is arrayed, with one

label for each button in the menu. This parameter is stored in a user's global file

and is set whenever a menu is called.

See also: User Programming

Related: menu Change status of menu system (C)

mstring Menu string (P)

move Move to an absolute location to start a line (C)

Syntax: move(<'graphics'|'plotter'>,x,y)

Description: Moves the start of a line to an absolute location with the coordinates given as an

argument. move is part of a line drawing capability that includes the pen and

draw commands. pen selects the pen number of the plotter ('pen1', 'pen2', etc.) or the color ('red', 'green', 'blue', etc.). move sets the point from which to start drawing the line. draw draws a line from that point to the point given by the draw arguments. Refer to the description of the draw

command for examples of using the line drawing capability.

Arguments: 'graphics' and 'plotter' are keywords selecting output to the graphics

window or a plotter device. The default is 'plotter'. The output selected is passed to subsequent pen, move, or draw commands, remaining unchanged

until different output is specified.

x, y are the absolute coordinates, in mm, of a point to move to. The range of x is 0 at the left edge of the chart and wcmax at the right edge of the chart. The

range of y is -20 at the bottom of the chart and wc2max at the top.

See also: NMR Spectroscopy User Guide

Related: draw Draw line from current location to another location (C)

gin Return current mouse position and button values (C)

pen Select a pen or color for drawing (C)

wcmax Maximum width of chart (P)

wc2max Maximum width of chart in second direction (P)

movedssw Set downsampling parameters for selected spectral region (M)

Description: Sets the parameters dslsfrq and downsamp to appropriate values for digital

filtering and downsampling in a cursor-selected spectral region. To accomplish this, Fourier transform an oversampled data set, and then run the ds program. In the resulting spectral display, enclose the desired region with the cursors, and

then run movedssw.

See also: NMR Spectroscopy User Guide

Related: downsamp Downsampling factor applied after digital filtering (P)

ds Display a spectrum (C)

dslsfrq Bandpass filter offset for downsampling (P)

moveossw Set oversampling parameters for selected spectral region (M)

Description: Sets the parameters oslsfrq and sw to appropriate values for oversampling

and digital filtering in a cursor-selected spectral region. To accomplish this, acquire a data set without digital filtering, and then run the ds program. In the resulting spectral display, enclose the desired region with the cursors, and then

run moveossw. The value of oversamp is manually set.

See also: NMR Spectroscopy User Guide

Related: ds Display a spectrum (C)

oslsfrq Bandpass filter offset for oversampling (P)
oversamp
Oversampling factor for acquisition (P)

Spectral width in directly detected dimension (P)

movesw Move spectral window according to cursors (M)

Syntax: movesw<(width)>

Description: Uses the parameters cr and delta to calculate a new transmitter offset tof

and a new spectral width sw. If referencing was used, it is also adjusted. The

movesw macro also sets sp and wp to display the spectral window.

Arguments: width specifies the spectral width sw. The default is to use a value calculated

from the parameter delta.

Examples: movesw

movesw(5000)

See also: NMR Spectroscopy User Guide

Related: cr Cursor position in directly detected dimension (P)

delta Cursor difference in directly detected dimension (P)
minsw Reduce spectral width to minimum required (M)

movetof Move transmitter offset (M)

sp Start of plot (P)

Spectral width in directly detected dimension (P)
tof
Frequency offset for observe transmitter (P)

Width of plot (P)

movetof Move transmitter offset (M)

Syntax: movetof<(frequency)>

Description: Moves the transmitter offset parameter tof so that the current cursor position,

defined by cr, becomes the center of the spectrum. If referencing was used,

movetof maintains the referencing.

Arguments: frequency specifies the transmitter frequency rather than using the cursor

position to define the frequency. This provides a convenient method of moving

the transmitter frequency outside the current spectral window.

See also: NMR Spectroscopy User Guide

Related: cr Cursor position in directly detected dimension (P)

minsw Reduce spectral width to minimum required (M)
movesw Move spectral window according to cursors (M)
tof Frequency offset for observe transmitter (P)

mp Move parameters between experiments (C)

Syntax: mp(<from exp,>to exp)

Description: Moves text and the current display, processing, and acquisition parameters from

one experiment to another. No FID is transferred.

Arguments: from exp specifies the number of the experiment from which the parameters

are to be taken; default is the parameters are moved from the currently active

experiment.

to exp specifies to which experiment the parameters are to be moved.

Examples: mp(4)

mp(2,3)

See also: NMR Spectroscopy User Guide

Related: md Move display parameters between experiments (C)

mf Move FIDs between experiments (C)

## mqcosy Set up parameters for MQCOSY pulse sequence (M)

Syntax: mqcosy<(level)>

Description: Sets up a multiple-quantum filtered COSY experiment.

Arguments: level is the desired quantum level of filtration.

Examples: mqcosy

mqcosy(3)

See also: NMR Spectroscopy User Guide

#### mref Set referencing based on a existing spectrum of the sample (M)

Syntax: mref(<source exp,>target exp)<:\$ret>

mref(source fid)<:\$ret>

Description: Use a primary referenced spectrum to reference a secondary spectrum acquired

in another work space (or experiment) at the same temperature, using the same lock sample, and either a different or the same nucleus without adding a secondary reference sample. The primary spectrum must be properly referenced using the IUPAC recommended  $\Xi$  values.  $\Xi$  is the normalized frequency such

that the <sup>1</sup>H signal from TMS is 100.00 MHz.

Begin with a source\_exp spectrum (typically a <sup>1</sup>H spectrum) and reference it using an internal reference (such as TMS, see the IUPAC recommendations).

Join a different experiment and acquire a target\_exp spectrum on a different (or same) nucleus. Enter mref (<source\_exp,>target\_exp).

Referencing of 2D data sets using mref only applies to the directly detected dimension. The indirect dimensions is referenced using reff1 and reff2 (after using mref or after manual referencing of the observe dimension). The reference frequency for the secondary spectrum, reffrq\_b, is calculated as follows:

```
reffrq b = (reffrq a / \Xi_a) * \Xi_b
```

mref also corrects for possible changes in the lock frequency:

```
reffrq b = (reffrq a / lockfreq a) * lockfreq b
```

mref works if the lock frequency changed between the two acquisitions, if the two spectra were acquired on different instruments, or at different field strengths.

mref calculates rfl and rfp after calculating reffrq:

```
rfp = 0

rfl = sw/2 - (sfrq - reffrq) * 1e6
```

The systemglobal parameters lockfreq and hlfreq must saved in the local parameters using the saveglobal mechanism when the go command is executed. The mref macro only tracks lock frequency changes if these systemglobal parameters are saved in the local parameters.

The mref macro works with earlier data if both data sets were:

• acquired at the same lock frequency (on the same instrument).

• the lockfreq (on a data station) and (on older instruments) h1freq parameters are set to the values used to acquire the data.

Referencing action from mref are reported the on line 3. Suppress the report by suppling a return argument, e.g.:

```
$ret='' mref('myfid.fid'):$ret
```

The referencing message is captured in the return argument "\$ret" and the contents of this string variable can be used to label plots with the referencing information.

Limitations: the macro works with data recalled from an archive or acquired on an other system provided the data was acquired using VNMR6.1C or newer.

Setting the global (or local) flag bioref='y' enables Bio-NMR referencing (based on nuctables/nuctabrefBio) and disables standard IUPAC / organic chemistry referencing (based on nuctables/nuctabref).

See /vnmr/nuctables/nuctabref.

# Arguments:

source\_exp — experiment containing the primary referenced spectrum or the full (or relative) path and fid file name containing the primary references spectrum.

target\_exp — experiment contining spectra to be referenced based upon the primary experiment referencing.

\$ret — return argument for output of mref.

Alternatively, the name of a FID file (with or without extension) can be given as a single argument; in this case, the data in the CURRENT experiment are referenced based on the referencing in the specified FID file.

#### Examples:

mref (3) — uses the current experiment as the source and applies the reference to the specified experiment as the target.

mref (1, 2) — experiment 1 is the source and experiment 2 is the target.

mref('myfid')

mref('/data/fids/myfid.fid')

# Related

setrefSet Frequency Referencing Based on Lock Signal Shift (M)setref1Set Frequency Referencing for f1 Evolution Dimension (M)setref2Set Frequency Referencing for f2 Evolution Dimension (M)reff1Reference f1 Indirect Dimension from Observe Dimension (M)reff2Reference f2 Indirect Dimension from Observe Dimension (M)

bioref Flag for Bio-NMR Referencing (P)

#### mrev8 Set up parameters for MREV8 pulse sequence (M)

Applicability: Systems with a solids module.

Description: Converts FLIPFLOP, BR24, or S2PUL parameter set into the MREV8 multiple-

pulse line narrowing sequence.

See also: User Guide: Solid-State NMR

Related: br24 Set up parameters for BR24 pulse sequence (M)

cylmrev Set up parameters for cycled MREV8 pulse sequence (M)

flipflop Set up parameters for FLIPFLOP pulse sequence (M)

s2pul Set up parameters for standard two-pulse sequence (M)

#### mrfb Set the filter bandwidths for multiple receivers (P)

Applicability: Systems with multiple receivers

Description: An array of fb settings to apply to individual receivers in a multiple receiver

system. The first element applies to the first receiver, the second to the second receiver, and so on. If mrfb exists and is active, these settings override the setting specified by the fb parameter; otherwise, fb is used as the filter bandwidth setting for all receivers. If there are fewer elements in mrfb than there are receivers, the remaining receivers are set to the fb value.

Note that some older multiple receiver systems do not have the hardware to provide individual receiver control. In that case, the filter setting for receiver 1 is used on receivers 1 and 2 and the setting for receiver 3 is used on receivers 3 and 4.

Also note that mrfb is not automatically set when sw is changed. Normally, you

can leave mrfb inactive and let fb be used for all receivers.

Examples: mrfb=fb/3, fb/2 sets the filter bandwidth of the first receiver to fb/3, the

second to fb/2, and of the rest to fb.

Related: fb Filter bandwidth (P)

# mrgain Set the gain for multiple receivers (P)

Applicability: Systems with multiple receivers

Description: An array of 'gain' settings to apply to individual receivers in a multiple

receiver system. If it exists and is active, these settings override the setting specified by the 'gain' parameter; otherwise, 'gain' is used as the gain

setting for all receivers.

Note that not all multiple receiver systems have the hardware set up to provide individual receiver control. In that case, the gain setting for receiver 1 is used on receivers 1 and 2 and the setting for receiver 3 is used on receivers 3 and 4.

Examples: mrgain=30, 40, 20 sets the gains of receiver 1 to 30, receiver 2 to 40 and

receivers 3 and 4 to 20.

Related: gain Receiver gain (P)

# mstat Display memory usage statistics (C)

Syntax: mstat<(program\_id)>

Description: Displays statistics on memory usage by programs that use the procedures

allocateWithId and release.

Arguments: program id is the program ID, usually the same name as the program. The

default is to display all program IDs and associated memory statistics.

Examples: mstat

mstat('proc2d')

See also: User Programming

## mstring Menu string (P)

Description: Stores command strings to be executed when a VnmrJ menu button is clicked.

Usually the mstring parameter is arrayed, with one string for each button in the menu. The string can be any string of commands that can otherwise appear in a macro or on the command line. This parameter is stored in a user's global

file and is set whenever a menu is called.

See also: User Programming

Related: menu Change status of menu system (C)

mlabel Menu label (P)

# mtune Tune probe using swept-tune graphical display (M)

Description: mtune replaces qtune on the Varian NMR System and/or Linux. mtune runs

in the spectra screen and uses VnmrJ panels. Enter mtune to retrieve

parameters and panels.

• all parameters changeable on-the-fly (exception: tune channel for the Varian NMR System).

• one or two markers are selectable to tune at the same time.

vertical autoscale button.

• number of acquired points changeable for better resolution at large spectral widths (more points will update less often).

• quit button returns user to current experiment and returns mtune to the original frequencies.

See also: NMR Spectroscopy User Guide

Related: tchan RF channel number used for tuning (P)

tugain Amount of receiver gain used by qtune (P)

tune Assign frequencies (C)

#### mv Move and/or rename a file (C)

Syntax: mv(from\_file,to\_file)

Description: Renames and/or moves a file or directory. mv functions the same as the

command rename.

Arguments: from file is the name of the file to be moved and/or renamed.

to file is the new name of the file and/or the new location. If the

from file argument has an extension such as .fid or .par, be sure the

to file argument has the same extension.

Examples: mv('/home/vnmr1/vnmrsys/seqlib/d2pul',

'/vnmr/seqlib/d2pul')

See also: NMR Spectroscopy User Guide

Related: copy Copy a file (C)

Copy a file (C)

delete Delete a file, parameter directory, or FID directory (C)

rename Move and/or rename a file (C)

rm Delete a file (C)

## mxconst Maximum scaling constant (P)

Description: Before the start of data acquisition, noise is sampled to determine the number of

bits of noise present. This number is used to set the maximum number of scaling operations on the data that can occur (essentially relevant only if dp='n').

mxconst is used to adjust this amount of scaling.

Increasing mxconst to 1, for example, permits additional scaling operations, allowing acquisition to proceed slightly longer in single-precision mode.

Decreasing mxconst to -1 allows fewer scaling operations before reaching

the message "maximum transients accumulated".

One special case exists. If mxconst is set to less than -90 and single-precision acquisition is used (dp='n'), then scaling of the data is disabled. In this mode, reports of data overflowing the 16 bits is also disabled.

mxconst does not exist in standard parameter sets. If it does not exist, its value defaults to 0. To modify mxconst, first create it by entering

create('mxconst', 'integer') and then enter the desired value.

CAUTION: Do not change mxconst unless you are fully aware of the

consequences.

See also: NMR Spectroscopy User Guide

Related: create Create new parameter in a parameter tree (C)

dp Double precision (P)

# N

n1, n2, n3 Name storage for macros (P)

newmenuSelect a menu without immediate activation (C)newshmInteractively create a shim method with options (M)

nextpl Display the next 3D plane (M)

nfni
 Number of increments in 1st indirectly detected dimension (P)
 ni2
 Number of increments in 2nd indirectly detected dimension (P)
 ni3
 Number of increments in 3rd indirectly detected dimension (P)

niter Number of iterations (P)
nimax Maximum limit of ni (P)

Position cursor at the nearest line (C)

nli Find integral values (C)

nlivastProduces a text file of integral regions without a sum region (M)nlivast2Produces a text file with normalized integral regions (M)nlivast3Produces a text file with normalized integral regions (M)

nll Find line frequencies and intensities (C)
nm Select normalized intensity mode (C)
nm2d Select Automatic 2D normalization (M)

Noesy
Convert the parameter to a NOESY experiment (M)
Noesy1d
Convert the parameter set to a Noesy1d experiment (M)

noise Measure noise level of FID (C)

noisemult Control noise multiplier for automatic 2D processing (M)

noislm Limit noise in spectrum (M)

notebook Notebook name (P)

Number of data points (P)

npoint Number of points for fp peak search (P)
nrecords Determine number of lines in a file (M)

nt Number of transients (P)

ntrigNumber of trigger signals to wait before acquisition (P)ntype3dSpecify whether f1 or f2 display expected to be N-type (P)nuctableDisplay VNMR style nucleus table for a given H1 frequency (M)

numrcvrs Number of receivers in the system (P)

numreg Return the number of regions in a spectrum (C)

numrfch Number of rf channels (P)

# n1, n2, n3 Name storage for macros (P)

Description: Stores arbitrary character strings for macros. Each experiment has these three

string parameters available.

See also: User Programming

Related: dgs Display group of special/automation parameters (M)

r1-r7 Real value storage for macros (P)

## newmenu Select a menu without immediate activation (C)

Syntax: (1) newmenu(menu\_name)

(2) newmenu:\$current menu

Description: Selects a menu but does not activate it (syntax 1). This is most useful when

picking which menu will be active when an interactive command exits. newmenu can also return the name of the currently active menu (syntax 2).

Arguments: menu name is the name of the file controlling the menu selected. For example,

the command string newmenu ('manipulate\_1D') ds causes the menu controlled by manipulate 1D to be displayed when the Return button in the

ds menu is selected.

\$current\_menu returns the file name of the currently active menu.

Examples: newmenu('display\_1D')

newmenu:\$name1

See also: User Programming

Related: menu Change status of menu system (C)

menuvi Edit a menu with the *vi* text editor (M)

## newshm Interactively create a shim method with options (M)

Syntax: newshm

Description: Interactively creates a *method* string to be used in autoshimming of the

magnetic field homogeneity. The string may consist of a series of shimming operations. The command dshim('method') describes method strings.

Any text editor may be used to make and modify the strings.

newshm provides for either lock shimming or FID shimming, permitting the user to choose whichever is best. Lock shimming is much faster, but FID shimming is frequently much more effective in improving the field. With FID shimming, the FID evaluation range limits are requested. The full range is 0 to 100. Sensitivity to higher order gradients is greatly increased by setting the finish limit to about 5 or 10 with the start limit at 0.

newshm begins by asking for the name of the user's new shim method. If the non-spin (transverse) controls are chosen for adjustment, the spinner is turned off; otherwise, it is turned on. If uncertain about the shim criteria, the "medium to medium" choice is suitable in most circumstances. The new method is found in <a href="mailto:curexp+">curexp+</a> '/.../shimmethods.

To shim after running newshm, type method='methodname' and then enter shim or set the wshim parameter to shim before the start of acquisition. 'methodname' is the name supplied to newshm. For more information on shimming, see the manual NMR Spectroscopy User Guide.

Compared to stdshm, the newshm macro is more flexible and provides for a shimming time and FID evaluation limits supplied by the user. The primary difference between the macros is that stdshm provides for determining an estimated shimming time for the selected shim controls. When no time limit is supplied, autoshim continues until the exit criteria is met or the number of cycles reaches a limit.

See also: NMR Spectroscopy User Guide

Related: curexp Current experiment directory (P)

dshim Display a shim method string (M)

method Autoshim method (P)

Submit an Autoshim experiment to acquisition (C)

stdshm Interactively create a shim method (M)

wshim Conditions when shimming is performed (P)
vi Edit text file with vi text editor (C)

nextpl Display the next 3D plane (M)

Syntax: nextpl

Description: Displays the 2D color map of the next 3D plane in the set of planes defined by

the parameters plane and path3d. If nextpl immediately follows the command dproj, nextpl results in the display of the first 3D plane within that specified set and is therefore equivalent to the command dplane (1). For example, if dplane (40) has just been executed, nextpl results in the display of 3D plane 41 of that set. The nextpl macro is more efficient than dplane or dproj because the 3D parameter set (procpar3d) is not loaded into VnmrJ—it is assumed to have already been loaded by dplane or dproj,

for example.

See also: NMR Spectroscopy User Guide

Related: dplane Display a 3D plane (M)

dproj Display a 3D plane projection (M)
dsplanes Display a series of 3D planes (M)

getplane Extract planes from a 3D spectral data set (M)

Path to currently displayed 2D planes from a 3D data set (P)

plane Currently displayed 3D plane type (P)
plplanes Plot a series of 3D planes (M)
prevpl Display the previous 3D plane (M)

nfni Number of increments in 1st indirectly detected dimension (P)

Description: Number of increments of the evolution time d2, and thus the number of FIDs

that will comprise the first indirectly detected dimension of a multidimensional data set. To create parameters ni, phase, and swl to acquire a 2D data set in

the current experiment, enter addpar ('2d').

Values: 8 is minimum; typical values range from 32 to 512. In microimaging, ni greater

than 0 is the imaging mode and ni equal to 0 is the projection mode.

See also: NMR Spectroscopy User Guide; VnmrJ Imaging NMR

Related: addpar Add selected parameters to the current experiment (M)

celem Completed FID elements (P)

d2 Incremented delay in 1st indirectly detected dimension (P)
ni2 Number of increments in 2nd indirectly detected dimension (P)

ni2 Number of increments in 2nd indirectly detected dimension (P)

Description: Number of increments of the evolution time d3, and thus the number of FIDs

that will comprise the second indirectly detected dimension of a

multidimensional data set. To create parameters d3, ni2, phase2, and sw2 to acquire a 3D data set in the current experiment, enter addpar ('3d').

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

Incremented delay in 2nd indirectly detected dimension (P)

Number of increments in 1st indirectly detected dimension (P)

par3d

Create 3D acquisition, processing, and display parameters (M)

phase 2 Phase selection for 3D acquisition (P)

Spectral width in 2nd indirectly detected dimension (P)

ni3 Number of increments in 3rd indirectly detected dimension (P)

Description: Number of increments of the evolution time d4, and thus the number of FIDs

that will comprise the third indirectly detected dimension of a multidimensional data set. To create parameters d4, ni3, phase3. and sw3 to acquire a 4D data

set in the current experiment, enter addpar ('4d').

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

Incremented delay in 3rd indirectly detected dimension (P)

Number of increments in 1st indirectly detected dimension (P)

Number of increments in 2nd indirectly detected dimension (P)

par4d Create 4D acquisition parameters (M)
phase3 Phase selection for 4D acquisition (P)

Spectral width in 3rd indirectly detected dimension (P)

niter Number of iterations (P)

Description: Sets the maximum number of iterations in an iterative simulation.

Values: 1 to 9999. The value is initialized to 20 if the Set Params button is used in setting

up spin simulation parameters.

See also: NMR Spectroscopy User Guide

nimax Maximum limit of ni (P)

Description: Maximum limit of ni. Used to prevent running an unrealistic number of

Hadamard-encoded experiments.

Values: Any positive real integer.

See also: NMR Spectroscopy User Guide

Related: sethtfrq1 Set a Hadamard frequency list from a line list (M)

Number of increments in 1st indirectly detected dimension (P)

htfrq1 Hadamard frequency in ni (P)

nl Position cursor at the nearest line (C)

Syntax: nl<:height<,frequency>>

Description: Moves the cursor to the nearest calculated line position.

Arguments: height is a return value set to the height of the line.

frequency is a return value set to the frequency of the line.

Examples: nl

nl:r1,r2

See also: NMR Spectroscopy User Guide

nli Find integral values (C)

Description: Equivalent to the dli command except that no screen display is produced. For

a list of integrals, nli stores the reset points in the parameter liftq and stores

the amplitudes in the parameter liamp.

See also: NMR Spectroscopy User Guide

Related: CZ Clear integral reset points (C)

dli Display list of integrals (C)

dlni Display list of normalized integrals (M)

liamp Amplitudes of integral reset points (P)
 lifrq Frequencies of integral reset points (P)
 Add integral reset point at cursor position (C)

nlivast Produces a text file of integral regions without a sum region (M)

Applicability: Systems with VAST accessory.

Syntax: nlivast(last)

Description: Using predefined integral regions from the spectra for each well, nlivast

writes a text file, integ.out, containing the integrals of the regions. The file is written into the current experiment. Does not add an additional region that is the

sum of all the defined regions for each well (see dlivast).

Arguments: last is the number of the last well. The default is 96.

See also: NMR Spectroscopy User Guide

nlivast2 Produces a text file with normalized integral regions (M)

Applicability: Systems with VAST accessory.

Syntax: nlivast(well)

Description: Using predefined integral regions from the spectra for each well, nlivast2

writes a text file, integ.out, containing the integrals of the regions. The file is written into the current experiment. Integrals are normalized to the integral specified by the argument well. The macro nlivast2 does not add an additional region that is the sum of all the defined regions for each well (see

dlivast). All of the spectra are integrated.

Arguments: well is the number of the reference sample well. The default reference is well

96.

See also: NMR Spectroscopy User Guide

nlivast3 Produces a text file with normalized integral regions (M)

Applicability: Systems with VAST accessory.

Syntax: nlivast(well)

Description: Using predefined integral regions from the spectra for each well, nlivast3

writes a text file, integ.out, containing the integrals of the regions. The file is written into the current experiment. Integrals are referenced to the integral specified by the argument well. The integral of spectrum from the sample specified by well is set to 1000. The macro nlivast3 does not add an additional region that is the sum of all the defined regions for each well (see

dlivast). All of the spectra are integrated.

Arguments: well is the number of the reference sample well. Reference integral set to

1000. The default reference is well 96.

See also: NMR Spectroscopy User Guide

nll Find line frequencies and intensities (C)

Syntax: nll<('pos'<,noise mult>)><:number lines,scale>

Description: Equivalent to the command dll except that the line listing is not displayed or

printed. The results of this calculation are stored in llfrq and llamp. The

frequencies are stored as Hz and are not referenced to rfl and rfp.

Amplitudes are stored as the actual data point value; they are not scaled by vs.

'pos' is a keyword that causes only positive lines to be listed.

noise mult is a numerical value that determines the number of noise peaks listed for broad, noisy peak. The default is 3. A smaller value results in more peaks, a larger value results in fewer peaks, and a value of 0.0 results in a line listing containing all peaks above the threshold th. Negative values of

noise mult are changed to 3.

number lines is a return argument with the number of lines in the line list. scale is a return argument with a scaling factor for line amplitudes. This scaling factor accounts for vs and whether the lines are listed in absolute

intensity mode or normalized mode.

Examples: nll:n1

nll('pos'):pn nll(2.5), sc

See also: User Programming

Related: Display listed line frequencies and intensities (C)

> 11amp List of line amplitudes (P) llfrq List of line frequencies (P)

#### Select normalized intensity mode (C) nm

Selects the normalized intensity mode in which spectra are scaled so that the Description:

largest peak in the spectrum is vs mm high. The alternative is the absolute intensity mode (selected by the ai command) in which the scale is kept constant from spectrum to spectrum to allow comparison of peak heights from one spectrum to another. The modes are mutually exclusive (i.e., the system is always in either nm or ai mode). Enter aig? to show which mode is currently

active.

See also: NMR Spectroscopy User Guide

Related: Select absolute intensity mode (C)

> Absolute intensity group (P) aig

Vertical scale (P) vs

#### nm2d Select Automatic 2D normalization (M)

Syntax: nm2d<(noisemult)>

Description: Sets up parameters th and vs2d automatically for a 2D contour plot and color

> map display. nm2d measures the highest signal in the spectrum and sets vs2d so that the highest signal is in the range of the highest color level. It then calculates the noise threshold so that the number of points above the noise threshold is between 10% and 30% of all the points. At the same time, the difference between the mean value of all the points above the threshold (peak points) and the mean value of all the points under the threshold (noise points) is maximized. This noise threshold is then multiplied by the noise multiplier.

> nm2d works both with absolute-value and phase-sensitive spectra. trace can be set to 'f1' or 'f2'.

Arguments:

noisemult specifies the noise multiplier number that multiplies the noise

- For <sup>1</sup>H, <sup>19</sup>F and <sup>31</sup>P (high dynamic range nuclei), and homonuclear spectra in general, the default value is 4.
- For HMQC/HSQC type spectra, the default value is also 4 but noise multipliers of 3 to 5 are often more adequate.
- For HETCOR and 2D-INADEQUATE spectra, the default value is 2.

- For "quick & dirty" COSY spectra with lots of t1 noise and other artifacts, a value of 8 and higher may be adequate for suppressing the artifacts.
- For 2D-INADEQUATE spectra, a value below 3 is appropriate to catch signals right above the noise level.
- If the multiplied noise threshold is below th=1, vs2d is scaled up; otherwise, th is increased to the desired level.
- Minimum value is 1.5 (if a lower value is entered, the value is set to 1.5).

Examples: nm2d

nm2d(3)

See also: NMR Spectroscopy User Guide

Related: dconi Interactive 2D contour display (C)

noisemult Control noise multiplier for automatic 2D processing (M)

proc2d Process 2D spectra (M)

th Threshold (P)

ws2d Mode for *n*-dimensional data display (P)
Vertical scale for 2D displays (P)

# Noesy Convert the parameter to a NOESY experiment (M)

Description: Convert the parameter to a NOESY experiment.

See also: NMR Spectroscopy User Guide

Related: foldt Fold COSY-like spectrum along diagonal axis (C)

## Noesy1d Convert the parameter set to a Noesy1d experiment (M)

Description: Convert the parameter set to a NOESY 1D experiment.

See also: NMR Spectroscopy User Guide

Related: Proton Set up parameters for <sup>1</sup>H experiment (M).

selld Selective 1D protocols to set up (M).

# noise Measure noise level of FID (C)

Syntax: noise<(excess\_noise<,last\_noise<,block\_number>>)>

:r1,r2,r3,r4,r5,r6

Description: Measures the noise level of a FID. By using pw=0 so that no real signal is

accumulated, one or more transients can be acquired. The value of np must be greater than 4096. noise then performs a statistical analysis of the noise, providing noise level, dc level, etc., for each channel. The noise level

measurement can be repeated at various settings of gain and various settings

of fb, etc., for a full system diagnosis.

Arguments: excess noise is excess noise and is used to calculate the noise figure.

last\_noise is the last measured mean square noise and is used to calculate

the noise figure.

block number is the block number. The default is 1.

r1 returns the real dc offset.

r2 returns the imaginary dc offset.

r3 returns the real rms noise.

r4 returns the imaginary rms noise.

r5 returns the average rms noise.

r6 returns the percentage channel imbalance.

r7 returns the noise figure.

See also: NMR Spectroscopy User Guide

Related: ddf Display data file in current experiment (C)

ddff Display FID file in current experiment (C)
ddfp Display phase file in current experiment (C)

fb Filter bandwidth (P)
gain Receiver gain (P)
np Number of data points (P)

Pulse width (P)

# noisemult Control noise multiplier for automatic 2D processing (M)

Syntax: noisemult<(noise\_multiplier)>

Description: Predetermines the noise multiplier used by the nm2d macro when starting

automatic 2D experiments. This multiplier determines the threshold level in 2D

spectra.

Arguments: noise\_multiplier is a noise multiplier, the same as used in the nm2d

macro. The default is 8 for homonuclear 2D spectra or 4 for other spectra.

Examples: noisemult

noisemult(10)

See also: NMR Spectroscopy User Guide

Related: nm2d Automatic 2D normalization (M)

proc2d Process 2D spectra (M)

# noislm Limit noise in spectrum (M)

Syntax: noislm<(max\_noise)>

Description: Limits the noise present in a spectrum by reducing the vertical scale vs. If the

noise is smaller than the noise limit, vs is left untouched. The noise limit is in single root-mean-square noise size; the peak-to-peak noise (width of the noise band) is about twice that value. The noise is determined by taking the smallest value from four 5% regions at the left end of the spectrum. Any filter cutoff at the end will decrease the apparent noise in the spectrum, and therefore increase the noise limit in the central part of the spectrum. Because of the particular algorithm used in this macro, signals at the left end of the spectrum should not

affect the result of noislm.

Arguments: max noise is the maximum root-mean-square size, in mm, of the noise. The

default is 2.

Examples: noislm

noislm(5)

See also: NMR Spectroscopy User Guide

Related: Vs Vertical scale (P)

vsadj Automatic vertical scale adjustment (M)

vsadjc Automatic vertical scale adjustment for <sup>13</sup>C spectra (M)
vsadjh Automatic vertical scale adjustment for <sup>1</sup>H spectra (M)

#### notebook Notebook name (P)

Description: Specifies the notebook name of a sample, which is saved with a study.

Related: cqsavestudy Macro to save study queue parameters (M)

page Name of page (P)
samplename Sample name (P)
studypar Study parameters (P)

np Number of data points (P)

Description: Sets number of data points to be acquired. Generally, np is a dependent

parameter and is calculated automatically when sw or at is changed. If a particular number of data points is desired, np can be entered, in which case at becomes the dependent parameter and is calculated based on sw and np.

Values: np is constrained to be a multiple of 2 (Acquisition Controller or Pulse

Sequence Controller board) or a multiple of 64 (Output board). (See the acquire statement in the manual *User Programming* for a description of these

boards.)

See also: NMR Spectroscopy User Guide

Related: at Acquisition time (P)

dp Double precision (P)

setlimit Set limits of a parameter in a tree (C)

Spectral width in directly detected dimension (P)

npoint Number of points for fp peak search (P)

Description: If npoint is defined in the current parameter set and has a value, it determines

the range of data points over which the fp command searches for a maximum for each peak. To create npoint and give it a value other than the default, enter create ('npoint', 'integer') npoint=x, where x is the new value.

Values: 1 to fn/4. The default is 2.

See also: NMR Spectroscopy User Guide

Related: create Create new parameter in a parameter tree (C)

fn Fourier number in directly detected dimension (P)

fp Find peak heights (C)

nrecords Determine number of lines in a file (M)

Syntax: nrecords(file):\$number\_lines

Description: Returns the number of lines (or records) in a file.

Arguments: file is the name of the file.

\$number lines returns the number of lines in the named file.

Examples: nrecords(userdir+'/mark1d.out'):\$num

See also: User Programming

nt Number of transients (P)

Description: Sets the number of transients to be acquired (i.e., the number of repetitions or

scans performed to make up the experiment or FID).

Values: 1 to 1e9. For an indefinite acquisition, set nt to a very large number such as 1e9.

See also: NMR Spectroscopy User Guide; VnmrJ Imaging NMR

ntrig Number of trigger signals to wait before acquisition (P)

Applicability: Systems with LC-NMR accessory.

Description: Sets the number of trigger signals from the LC to wait for on the external gate

line before beginning acquisition. If ntrig is 0 or the parameter does not exist, the external gate signal is ignored. If ntrig noes not exist, the parlc macro

can create it. ntrig is not normally entered by the user.

See also: NMR Spectroscopy User Guide

Related: parlc Create LC-NMR parameters (M)

# ntype3d Specify whether f<sub>1</sub> or f<sub>2</sub> display expected to be N-type (P)

Description: Indicates whether the f<sub>1</sub> or f<sub>2</sub> display is expected to be N-type, that is, opposite

to the sense of precession defined by f<sub>3</sub>, under normal 3D processing conditions.

Values: 'yn' specifies that f<sub>1</sub> is expected to have an N-type display under normal 3D

processing conditions.

'ny' specifies that f<sub>2</sub> is expected to have an N-type display under normal 3D

processing conditions.

'yy' specifies that both  $f_1$  and  $f_2$  are expected to have N-type displays under normal 3D processing conditions. Setting ntype3d='yy' changes the sense of precession in  $f_1$  and  $f_2$  by negating the imaginary portion of the  $t_1$  and  $t_2$ 

interferograms prior to Fourier transformation.

See also: NMR Spectroscopy User Guide

Related: fiddc3d 3D time-domain dc correction (P)

Perform a 3D Fourier transform on a 3D FID data set (M,U)

ptspec3d Region-selective 3D processing (P)
specdc3d 3D spectral dc correction (P)

Full bandwidth of digital filter to yield a filtered FID (P)

ssorder Order of polynomial to fit digitally filtered FID (P)

rftype Type of rf generation

# nuctable Display VNMR style nucleus table for a given H1 frequency (M)

Syntax: nuctable<(h1 freq)>

Description: The VnmrJ nucleus table is a single nucleus table, /vnmr/nuctables/

nuctable, which is calculated based on a proton frequency of 1000.000 MHz. nuctable can be used to reconstruct a traditional nucleus table, e.g., based on a proton frequency of 200.057 MHz, or to calculate a nucleus table for any given

proton frequency.

Arguments: h1 freq (optional): proton frequency on which the calculated / displayed

nucleus table will be based. Without argument, nuctable prints a nucleus table based on the proton frequency for which the current VnmrJ / VNMR

installation is configured.

Examples: nuctable(200.057)

nuctable:

Related restorenuctable Calculate and (Re-)store accurate nuctable (M)

#### numrcvrs Number of receivers in the system (P)

Applicability: Systems with multiple receivers.

Description: An integer giving the number of receivers installed in the system. numrcvrs

is set from the config panel by the vnmr1 user.

numreg Return the number of regions in a spectrum (C)

Syntax: numreg:number\_regions

Description: Returns the number of regions in a spectrum previously divided by the region

command, by manual means using the **z** command, or by the Resets button in ds. A *region* is the area between two reset points in integral mode, with every other reset point designating the start of a *baseline* region and not included in

the count of regions.

Arguments: number regions returns the number of peak regions in the spectrum.

Examples: numreg: \$num

See also: User Programming

Related: ds Display a spectrum (C)

getreg Get frequency limits of a specified region (C)

region Divide spectrum into regions (C)

Z Add integral reset point at cursor position (C)

numrfch Number of rf channels (P)

Description: Holds the number of rf channels available. The value is set with the Number of

RF Channels label in the Spectrometer Configuration window. numrfch represents the hardware in the system. For example, if the last experiment used the second decoupler, numrfch is set to 2. The software then leaves the second

decoupler on if it was on and leaves it off if it was off.

CAUTION: Do not reset numrfch to eliminate the use of a channel. See the

description of dn2 and dn3 for the method to disable channels.

Values: The fifth channel can only be used with the deuterium decoupler channel.

See also: VnmrJ Installation and Administration

Related: config Display current configuration and possibly change it (M)

dn2 Nucleus for the second decoupler (P)dn3 Nucleus for the third decoupler (P)dn4 Nucleus for the fourth decoupler (P)



off Make a parameter inactive (C)

on Make a parameter active or test its state (C)

operator Operator name (P)

operatorlogin Sets workspace and parameters for the operator (M)

opxOpen shape definition file for Pbox (M)oscoefDigital filter coefficients for over sampling (P)osfbDigital filter bandwidth for over sampling (P)osfiltOver sampling filter for real-time DSP (P)oslsfrqBandpass filter offset for over sampling (P)overrangeFrequency synthesizer overrange (P)

oversamp
Over sampling factor for acquisition (P)
owner
Operating system account owner (P)

#### off Make a parameter inactive (C)

Syntax: off(parameter<,tree>)

Description: Turns off an active parameter in any tree.

Arguments: parameter is the name of the parameter.

tree is type of parameter tree: 'current', 'global', 'processed', or 'systemglobal'. The default is 'current'. Refer to the create

command for more information on the types of trees.

Examples: off('gf')

off('n','global')

See also: User Programming

Related: create Create new parameter in a parameter tree (C)

on Make a parameter active or test its state (C) typeof Return identifier for argument type (O)

## on Make a parameter active or test its state (C)

Syntax: on(parameter<, tree>) <: \$active>

Description: Turns on an inactive parameter in any tree or tests if a parameter is active. Real

variables (not strings) can be turned on and off. This can be done in any tree with the commands on and off, and by entering name='y' or name='n' to change the active flag for variables in the current tree only. The variable trees are 'current', 'global', 'processed' and 'systemglobal'. The

default tree is 'current'.

To test the active flag of a variable, use on  $(\ldots)$ : \$x. This does not change the active flag of the variable, but sets \$x to 1, if the variable is active, or to 0, if it is not active. If the variable does not exist, a value of -1 is returned. Care should be taken if using the return value as a test for a conditional statement. For

example, in the following fragment,

```
on('var1'):$e
if $e then
  write('line3','if statement is true with value of
```

%d',\$e) endif

the write command will be executed if 'var1' is active, writing the message *if statement is true with value of 1* It will also be executed if 'var1' does not exist, writing the message *if statement is true with value of -1*.

To only execute the write command if the variable is active, use something like the following:

on('var1'):\$e
if (\$e > 0.5) then
 write('line3','var1 is active')
endif

Arguments: parameter is the name of the parameter to make active or to test.

tree is type of parameter tree: 'current', 'global', 'processed', or 'systemglobal'. The default is 'current'. Refer to the create command for more information on the types of trees.

\$active is 1 if the parameter is active, or is 0 if it is not active. Adding a return argument makes on conduct only a test of whether the specified parameter is

active and does *not* turn on the parameter if it is inactive.

Examples: on('lb'):\$ison

on('gain','global')

See also: User Programming

Related: create Create new parameter in a parameter tree (C)

off Make a parameter inactive (C)

# operator Operator name (P)

Applicability: VnmrJ Walkup

Description: Specifies the operator name. It is set when an operator logs into the Walkup

interface. Multiple operators may be defined for a single user using the VnmrJ

Administrator interface.

Related: acct Writes records for operator login and logoff (M)

operatorlogin Sets workspace and parameters for the operator (M)

vnmr accounting Open Accounting window (U)

# operatorlogin Sets workspace and parameters for the operator (M)

Syntax: operatorlogin operator email panellevel

Description: Sets the panel display level and other parameters for an operator when the

operator logs in. It also clears the new sample area in the study queue, and disables the command line if the operator has insufficient privileges. An operator may be logged in from the Switch operator dialog in the Utilities menu.

Related: acct Writes records for operator login and logoff (M)

email Email address (P)
operator Operator name (P)

panellevel Display level for VnmrJ interface pages (P)

vnmr\_accounting Open Accounting window (U)

# opx Open shape definition file for Pbox (M)

Syntax: opx<(name<.ext>)>

0

Description: Opens the pulse shape/pattern definition input file shapelib/Pbox.inp for

the Pbox software and writes the file header.

Arguments: name is the name of the output shape file.

ext is a file name extension that specifies the file type.

Examples: opx

opx('newfile.DEC')

Related: Pbox Pulse shaping software (U)

#### oscoef Digital filter coefficients for over sampling (P)

Description: Specifies number of coefficients used in the digital filter. Enter

addpar('oversamp') to add oscoef to the current experiment if
oscoef does not exist. addpar('oversamp') creates digital filtering and
oversampling parameters def\_osfilt, filtfile, oscoef, osfb,

osfilt, oslsfrq, and oversamp.

Values: The default is 7.5\*oversamp for inline DSP (dsp='i'). A larger number

of coefficients gives a filter with sharper cutoffs; a smaller number gives a filter with more gradual cutoffs. The value of oscoef does not need to be changed when oversamp is changed because oscoef is automatically adjusted by VnmrJ to give filter cutoffs that are the same regardless of the value of

oversamp.

The number of coefficients for real-time DSP (dsp='r') is determined by the

hardware and is not adjustable.

Related: addpar Add selected parameters to current experiment (M)

dspType of DSP for data acquisition (P)filtfileFile of FIR digital filter coefficients (P)osfbDigital filter bandwidth for oversampling (P)oslsfrqBandpass filter offset for oversampling (P)oversampOversampling factor for acquisition (P)

paros Create additional parameters used by oversampling (M)

#### osfb Digital filter bandwidth for oversampling (P)

Description: Specifies bandwidth of the digital filter used for oversampling. If osfb does

not exist in the current experiment, enter addpar ('oversamp') to add it.

addpar('oversamp') creates digital filtering and oversampling

parameters  ${\tt def\_osfilt}, {\tt filtfile}, {\tt oscoef}, {\tt osfilt}, {\tt oslsfrq}, {\tt and}$ 

oversamp.

Values: Number, in Hz. A value less than sw/2 rejects frequencies at the edges of the

spectrum; a value greater than sw/2 aliases noise and signals at frequencies

outside of  $\pm sw/2$ .

'n' sets the bandwidth to sw/2.

Related: addpar Add selected parameters to current experiment (M)

def osfilt Default value of osfilt (P)

filtfile File of FIR digital filter coefficients (P)

oscoef Digital filter coefficients for oversampling (P)

osfilt Oversampling filter for real-time DSP (P)

oslsfrq Bandpass filter offset for oversampling (P)

oversamp Oversampling factor for acquisition (P)

create additional parameters used by oversampling (M)
Sw Spectral width in directly detected dimension (P)

# osfilt Oversampling filter for real-time DSP (P)

Applicability: Systems with real-time DSP.

Description: Sets the type of real-time digital filter to be used on systems equipped with the

real-time DSP hardware option. osfilt is normally set automatically by the software based on the user's global parameter def\_osfilt, so that osfilt only needs to be changed if a particular experiment is to be run with a different

digital filter than the default.

Values: 'a' or 'A' for the Analog*Plus*<sup>TM</sup> digital filter.

'b' or 'B' for the brickwall digital filter.

'' (null string) causes osfilt to be set to the value contained in the def osfilt when an acquisition is initiated (with go, for example).

Related: def\_osfilt Default value of osfilt (P)

dsp Type of DSP for data acquisition (P)

# oslsfrq Bandpass filter offset for oversampling (P)

Description: Selects a bandpass filter that is not centered about the transmitter frequency. In

this way oslsfrq works much like lsfrq. If oslsfrq does not exist in the current experiment, add it with addpar ('oversamp'), which creates digital filtering and oversampling parameters, the same as the paros macro.

Values: Number, in Hz. A positive value selects a region upfield from the transmitter

frequency. A negative value selects a downfield region.

Related: addpar Add selected parameters to current experiment (M)

def osfilt Default value of osfilt (P)

filtfile

file of FIR digital filter coefficients (P)

fsq

Frequency-shifted quadrature detection (P)

lsfrq

Frequency shift of the fn spectrum in Hz (P)

oscoef

Digital filter coefficients for oversampling (P)

osfb

Digital filter bandwidth for oversampling (P)

osfilt

Oversampling filter for real-time DSP (P)

oversamp

Oversampling factor for acquisition (P)

paros Create additional parameters used for oversampling (M)

# overrange Frequency synthesizer overrange (P)

Applicability: Systems with optional version X46 of the PTS frequency synthesizer.

Description: Configures whether an rf channel has version X46 of the PTS frequency

synthesizer. The value for each channel is set using the label Frequency

Overrange in the Spectrometer Configuration window.

Values: Not Present, 10000 Hz, or 100000 Hz

Not Present indicates that this rf channel does not have the frequency

overrange option.

10000 or 100000 indicate that this rf channel has the frequency overrange option. The **10000 Hz** or **100000 Hz** choices are determined by the letters *H*, *J*, or *K* found in the PTS Synthesizers model number. The normal value for overrange is 10000 Hz. If **Frequency Overrange** is set to 10000 Hz or 100000 Hz, the **Latching** value for that RF channel must also be set to **Present**.

When set to either 10000 Hz or 100000 Hz, overrange guarantees a range of phase-continuous frequency jumps of at least 10 kHz or 100 kHz in each jump

direction.

See also: VnmrJ Installation and Administration

Related: config Display current configuration and possibly change it (M)

latch Frequency synthesizer latching (P)

# oversamp Oversampling factor for acquisition (P)

Description:

Specifies the oversampling factor for the acquisition. With inline digital filtering (dsp='i'), np\*oversamp data points are acquired at a rate of sw\*oversamp. The data is then transferred to the host computer, digitally filtered, and downsampled to give np points and a spectral width of sw.

With real-time digital filtering (dsp='r'), the oversampling, digital filtering, and down sampling all occur as each data point is collected, so that only np data points are ever stored in the acquisition computer memory and subsequently transferred to the host computer.

If oversamp does not exist in the current experiment, enter the command addpar('oversamp') to add it. addpar('oversamp') creates digital filtering and oversampling parameters def\_osfilt, filt, osfb, osfilt, oslsfrq, and oversamp.

If oversamp is set to a number, then that number represents the amount of oversampling to apply when collecting the data. The oversamp value is automatically calculated whenever sw is changed, provided oversamp is not set to 'n'. That is the distinction between oversamp='n' and oversamp=1. In both cases, no oversampling will be used. This occurs, for example, if the sw parameter is greater than half the maximum spectral width. However, if sw is reduced so that oversampling is possible, then if oversamp is set to 'n', oversamp will remain set to 'n' and oversampling will not occur. On the other hand, if oversamp is set to 1, then oversamp parameter accurately represents whether oversampling is performed for a data set. When oversamp is automatically determined based on a change to sw, it is set to the maximum possible oversampling factor. The value of oversamp can be manually reset.

Note that setting oversamp greater than 1 means oversampling is selected for the experiment. However, if the oversampling facility is not present in the system (i.e., dsp='n'), then the oversamp parameter is automatically reset to 1, indicating that no oversampling will be performed.

Two other experiment local parameters reflect whether DSP is used during the acquisition of a data set:

- fb is set to Not Active if DSP is used.
- oscoef reflects whether real-time (dsp='r') or inline (dsp='i') DSP was used. If real-time, oscoef is set to Not Active. If inline, oscoef is set to the value used by the inline algorithm.

Values: Number less than or equal to 68. For inline DSP, sw\*oversamp and np\*oversamp are limited by the values in the following table:

Maximum sw*oversamp	Maximum np*oversamp
500 kHz	2M
100 kHz	128K

The maximum np\*oversamp is given for double precision data (dp='y'). For dp='n', multiply this value by 2.

'n' causes normal acquisition to be done without digital filtering.

Related: addpar Add selected parameters to current experiment (M)

def\_osfilt Default value of osfilt parameter (P)

dp Double precision (P)

dsp Type of DSP for data acquisition (P)

Filter bandwidth (P)

filtfile File of FIR digital filter coefficients (P)

fsq Frequency-shifted quadrature detection (P)

np Number of data points (P)

oscoef
Digital filter coefficients for oversampling (P)
osfb
Digital filter bandwidth for oversampling (P)
osfilt
Oversampling filter for real-time DSP (P)
oslsfrq
Bandpass filter offset for oversampling (P)

paros Create additional parameters used by oversampling (M)
Sw Spectral width in directly detected dimension (P)

# owner Operating system account owner (P)

Description: Set to the Unix or Linux account owner. It is set when VnmrJ is started.

# P

Enter pulse width for p1 in degrees (C) p1 First pulse width (P) p1 Set up sequence for PFG testing (M) p2pul p31 Automated phosphorus acquisition (M) Process 1D phosphorus spectra (M) p31p Set phase angle mode in directly detected dimension (C) рa Set phase angle mode in 1st indirectly detected dimension (C) pa1 Plot automatic COSY analysis (C) pacosy Preacquisition delay (P) pad padept Perform adept analysis and plot resulting spectra (C) Submit plot and change plotter page (C) page Name of page (P) page Display level for VnmrJ interface pages (P) panellevel pap Plot out "all" parameters (C) par2d Create 2D acquisition, processing, and display parameters (M) Create 3D acquisition, processing, and display parameters (M) par3d par3rf Get display templates for 3rd rf channel parameters (M) par4d Create 4D acquisition parameters (M) paramedit Edit a parameter and its attributes with user-selected editor (C) Edit a parameter and its attributes with vi editor (M) paramvi pards Create additional parameters used by down sampling (M) Create parameters for time-domain solvent subtraction (M) parfidss parfix Update parameter sets (M) Create parameters for LC-NMR experiments (M) parlc parl12d Create parameters for 2D peak picking (M) Create parameters for linear prediction (M) parlp Parameter maximum values (P) parmax Parameter minimum values (P) parmin Create additional parameters used by over sampling (M) paros Parameter step size values (P) parstep parversion Version of parameter set (P) path3d Path to currently displayed 2D planes from a 3D data set (P) paxis Plot horizontal LC axis (M) Pbox Pulse shaping software (U) Define excitation band (M) pbox bw pbox\_bws Define excitation band for solvent suppression (notch) pulses (M) pbox dmf Extract dmf value from pbox.cal or Pbox shape file (M) pbox\_dres Extract dres value from pbox.cal or Pbox shape file (M) pbox\_name Extract name of last shape generated by Pbox from pbox.cal (M) pbox pw Extract pulse length from pbox.cal or Pbox shape file (M) pbox pwr Extract power level from Pbox.cal or Pbox shape file (M) pbox pwrf Extract fine power level from pbox.cal or Pbox shape file (M)

pboxget

Extract Pbox calibration data (M)

Add parameter definition to the Pbox.inp file (M) pboxpar pboxrst Reset temporary Pbox variables (M) Converts to Pbox default units (M) pboxunits pcon Plot contours on a plotter (C) pcss Calculate and show proton chemical shifts spectrum (M) Find tallest peak in specified region (C) peak Return information about maximum in 2D data (C) peak2d pen Select a pen or color for drawing (C) pexpl Plot exponential or polynomial curves (C) pexpladd Add another diffusion analysis to current plot (M) Pulsed field gradient amplifiers on/off control (P) pfgon pfww Plot FIDs in whitewash mode (C) Convert parameter set to PGE pulse sequence (M) pge Calibrate gradient strengths for PGE pulse sequence (M) pge\_calib pge data Extract data from single element of PGE pulse sequence (M) Output results from PGE pulse sequence (M) pge output Automated processing of data from PGE pulse sequence (M) pge\_process Calculate diffusion constant for integral region (M) pge results pge\_setup Set up gradient control parameters for PGE pulse sequence (M) Set phased mode in directly detected dimension (C) ph Set phased mode in 1st indirectly detected dimension (C) ph1 ph2 Set phased mode in 2nd indirectly detected dimension (C) phase Change frequency-independent phase rp (M) Phase selection (P) phase Phase of first pulse (P) phase1 phase2 Phase selection for 3D acquisition (P) Phase selection for 4D acquisition (P) phase3 Control update region during interactive phasing (P) phasing Zero-order phasing constant for the np FID (P) phfid phfid1 Zero-order phasing constant for ni interferogram (P) Zero-order phasing constant for ni2 interferogram (P) phfid2 Set up parameters for <sup>31</sup>P experiment (M) Phosphorus pi3ssbsq Set up pi/3 shifted sinebell-squared window function (M) pi4ssbsq Set up pi/4 shifted sinebell-squared window function (M) pin Pneumatics Router Interlock ((P) pintvast Plots of integral regions (M) Plot integral amplitudes below spectrum (C) pir pirn Plot normalized integral amplitudes below spectrum (M) Plot integral amplitudes below spectrum (M) piv Plot normalized integral amplitudes below spectrum (M) pivn p1 Plot spectra (C) pl2d Plot 2D spectra in whitewash mode (C) plt2Darg Plot 2D arguments (P) Currently displayed 3D plane type (P) plane plapt Plot APT-type spectra automatically (M)

Plotting macro for arrayed 1D spectra (M)
Define a glue order for plotting and display (U)

plarray

plate\_glue

Plot a carbon spectrum (M)

Plot COSY- and NOESY-type spectra automatically (M)

pldept Plot DEPT data, edited or unedited (M)

plfid Plot FIDs (C)

plfit Plot deconvolution analysis (M)
plgrid Plot a grid on a 2D plot (M)
plh Plot proton spectrum (M)

plhet2dj Plot heteronuclear J-resolved 2D spectra automatically (M)
plhom2dj Plot homonuclear J-resolved 2D spectra automatically (M)

Plot X,H-correlation 2D spectrum (M)

Plot a line list (M)

pll2d Plot results of 2D peak picking (C)

Port number to use to lock out multiple ProTune processes (P)

Plot Automatically plot spectra (M)

Plotting macro for simple (non-arrayed) 1D spectra (M)

plot2D Plot 2D spectra (M)
plotfile Plot to a file (M)

plothiresprep High resolution plot output preparation (M)

plotmanualPlot manually (M)plotlogoPlots a logo (M)

plotside Plot spectrum on side (M)

plotter Plotter device (P)

plottop Plot spectrum on top (M)

plottopside Plot spectrum on top and side (M)
plp Plot phosphorus spectrum (M)
plplanes Plot a series of 3D planes (M)

pltext Plot 2D arguments (P)
pltext Plot text file (M)
pltmod Plotter display mode (P)

plvast Plot VAST data in a stacked 1D-NMR matrix format (M)
plvast2d Plot VAST data in a stacked pseudo-2D format (M)

Plot spectra in whitewash mode (C)

pmode

Processing mode for 2D data (P)

poly0 Display mean of the data in regression.inp file (M)

pp Decoupler pulse length (P)

ppa Plot a parameter list in plain English (M)
ppcal Proton decoupler pulse calibration (M)
ppf Plot peak frequencies over spectrum (C)

pph Print pulse header (M)

ppmm Resolution on printers and plotters (P)
pprofile Plot pulse excitation profile (M)

pps Plot pulse sequence (C)

prealfa Specify a delay for longer ring down (P)
prep Run prepare acquisition macro (M)

Presat Set up parameters for presat <sup>1</sup>H experiment (M)

prescan Study queue prescan (P)

prevpl Display the previous 3D plane (M)

prescan CoilTable Read or update the CoilTable File (M)

prescan tn Return tn string for a given atomic number (M)

**printer** Printer device (P)

printfile Path to the print-to-file image (P)
printformat Format of saved-to-file image (P)
printlayout Layout of printed image (P)

**printoff** Stop sending text to printer and start print operation (C)

printonDirect text output to printer (C)printregionScreen region to be printed (P)printsizeSize of printed image (P)

printsend Defines where image will print (P)

probe Probe type (P)

probeConnect Specify which nucleus can be acquired on each RF channel (P)

Probe\_edit Edit probe for specific nucleus (U)
probe\_edit Edit probe for specific nucleus (M)

probe\_protection Probe protection control (P)
proc Type of processing on np FID (P)

Type of processing on ni interferogram (P)

proc1d Processing macro for simple (non-arrayed) 1D spectra (M)

Type of processing on ni2 interferogram (P)

proc2d Process 2D spectra (M)

process arrayed 1D spectra (M)
process
Generic automatic processing (M)
procplot
Automatically process FIDs (M)

profile Set up pulse sequence for gradient calibration (M)

proj Project 2D data (C)

Proton Set up parameters for <sup>1</sup>H experiment (M)

protune Macro to start ProTune (M)

Shell script to start ProTune operation (U)

protunegui Macro to start ProTune in graphical user interface (M)

prune Prune extra parameters from current tree (C)
pscale Plot scale below spectrum or FID (C)

**Set** default parameters for pseudo-echo weighting (M)

psgDisplay pulse sequence generation errors (M)psggenCompile a user PSG object library (M,U)

psgset Set up parameters for various pulse sequences (M)
psgupdateon Enable update of acquisition parameters (C)

psgupdateoffPrevent update of acquisition parameters (C)pshapePlot pulse shape or modulation pattern (M)pshapefPlot the last created pulse shape (M)

PostScript High Resolution plotting control (P)

pslabel
Pulse sequence label (P)

psslw PostScript Line Width control (P)
pssl Plot Arrayed Numbers (C)
ptext Print out a text file (M)

ptspec3d Region-selective 3D processing (P)
ptsval PTS frequency synthesizer value (P)

pulseinfo Shaped pulse information for calibration (M)

pulsetoolRF pulse shape analysis (U)purgeRemove macro from memory (C)puttxtPut text file into a data file (C)putwaveWrite a wave into Pbox.inp file (M)pwEnter pulse width pw in degrees (C)

pw Pulse width (P) pw90 90° pulse width (P)

Display current working directory (C)

pwr Set power mode in directly detected dimension (C)
pwr1 Set power mode in 1st indirectly detected dimension (C)
pwr2 Set power mode in 2nd indirectly detected dimension (C)

pwsadj Adjust pulse interval time (M)
pwxcal Decoupler pulse calibration (M)

pxbss Bloch-Siegert shift correction during Pbox pulse generation (P)

**pxrep** Flag to set the level of Pbox reports (P)

Assign Pbox calibration data to experimental parameters (M)

pxshapeGenerates a single-band shape file (M)PxsimSimulate Bloch profile for a shaped pulse (U)

Pxspy Create shape definition using Fourier coefficients (U)

# p1 Enter pulse width for p1 in degrees (C)

Syntax: p1(flip angle<,90 pulse width>)

Description: Calculates the flip time, in µs, given a desired flip angle and the 90° pulse. The

value is entered into the pulse width parameter p1.

Arguments: flip angle is the desired flip angle, in degrees.

90 pulse width is the  $90^{\circ}$  pulse, in  $\mu s$ . The default is the value of

parameter pw90 if it exists.

Examples: p1(30)

p1(90,12.8)

See also: NMR Spectroscopy User Guide

Related: ernst Calculate the Ernst angle pulse (C)

p1 First pulse width (P) pw90 90° pulse width (P)

# p1 First pulse width (P)

Description: Length of first pulse in the standard two-pulse sequence.

Values: 0, 0.2 μs to 150,000 μs, in 0.1 μs steps

0.1 µs to 8190 sec, smallest value possible is 0.1 µs, finest increment possible

is 12.5 ns.

See also: NMR Spectroscopy User Guide

Related: p1 Enter pulse width p1 in degrees (C)

## plpat Shape of excitation pulse (P)

Applicability: Systems with imaging capabilities.

Description: Specifies the shape of pulse p1 when used in imaging experiments.

Values: 'hard', 'sinc', 'gauss', 'sech', 'sine', or any shape resident in the

system pulse shape library or libraries.

See also: VnmrJ Imaging NMR

Related: p1 First pulse width (P)

pwpat Shape of refocusing pulse (P)

#### p2pul Set up sequence for PFG testing (M)

Applicability: Systems with the pulsed field gradient (PFG) module. This sequence is not for

NMR applications.

Description: Sets up the PFG two-pulse sequence, a system checkout sequence for PFG

installation. Several modes are controlled by the cmd parameter.

• cmd='twinkle' sequentially addresses DACs 0 through 4. On the gradient channel interface, lights become a slow binary counter.

• cmd='pulse' makes a pulse of value gzlvl1 for a time gt1.

• cmd='bipulse' makes a pulse of value gzlvl1 for a time gt1 followed by a pulse of value -qzlvl1 for a time gzlvl1.

For other modes, see the PFG installation manual.

See also: Pulsed Field Gradient Modules Installation

# p31 Automated phosphorus acquisition (M)

Syntax: p31<(solvent)>

Description: Prepares parameters for automatically acquiring a standard <sup>31</sup>P spectrum. The

parameter wexp is set to 'procplot' for standard processing. If p31 is used as the command for automation via the enter command, then the macro au is supplied automatically and should not be entered on the MACRO line of the enter program. However, it is possible to customize the standard p31 macro on the MACRO line by following it with additional commands and parameters. For example, p31 nt=1 will use the standard p31 setup but with only one

transient.

Arguments: solvent is the name of the solvent. The default is CDC13. In automation

mode, the solvent is supplied by the enter program.

Examples: p31

p31('DMSO')

See also: NMR Spectroscopy User Guide

Related: au Submit experiment to acquisition and process data (M)

enter Enter sample information for automation run (C)

p31p Process 1D phosphorus spectra (M)

proc1d Processing macro for simple, non-arrayed 1D spectra (M)

procplot Automatically process FIDs (M)
wexp When experiment completes (P)

## p31p Process 1D phosphorus spectra (M)

Syntax: p31p

Description: Processes non-arrayed 1D <sup>31</sup>P spectra using a set of standard macros. p31p is

called by the procld macro but can also be used directly. Fully automatic processing (up to a point where a spectrum could be plotted) is provided:

Fourier transformation (using preset weighting functions), automatic phasing (aphx macro), automatic integration (integrate macro, if required only), vertical scale adjustment (vsadjc macro), avoiding excessive noise (noislm macro), threshold adjustment (thadj macro), and referencing to the TMS signal, if present (tmsref macro).

See also: NMR Spectroscopy User Guide

Related: aphx Perform and check automatic phasing (M)

> integrate Automatically integrate 1D spectrum (M)

noislm Avoids excessive noise (M)

p31 Automated phosphorus acquisition (M)

Automatically process non-arrayed 1D fids (M) proc1d

thadj Adjust threshold (M)

Reference spectrum to TMS line (M) tmsref vsadic Adjust vertical scale for carbon spectra (M)

#### Set phase angle mode in directly detected dimension (C) рa

Description: Selects the phase angle mode by setting the parameter dmg='pa'. In the phase

angle display mode, each real point in the displayed spectrum is calculated from the phase angle of the real and imaginary points comprising each respective complex data point. The phase angle also takes into account the phase

parameters rp and lp.

For 2D data, if pmode='partial' or pmode=' (two single quotes with no space in between), pa has an effect on the data prior to the second Fourier transform. If pmode='full', pa acts in concert with the commands pa1, av1, pwr1, or ph1 to yield the resultant contour display for the 2D data.

See also: NMR Spectroscopy User Guide

Related: Set abs. value mode in directly detected dimension (C)

Data display mode in directly detected dimension (P) dma

ft. Fourier transform 1D data (C)

ft1d Fourier transform along f<sub>2</sub> dimension (C)

ft2d Fourier transform 2D data (C)

1p First-order phase in directly detected dimension (P)

Set phase angle mode in 1st indirectly detected dimension (C) pa1

Set phased mode in directly detected dimension (C) ph

Processing mode for 2D data (P) pmode

Set power mode in directly detected dimension (C) pwr Set power mode in 1st indirectly detected dimension (C) pwr1 Zero-order phase in directly detected dimension (P) rp

wft Weight and Fourier transform 1D data (C) wft1d Weight and Fourier transform f2 of 2D data (M) wft2d Weight and Fourier transform 2D data (M)

#### Set phase angle mode in 1st indirectly detected dimension (C) pa1

Selects the phase angle spectra display mode along the first indirectly detected Description: dimension by setting the parameter dmq1 to the string value 'pa1'. If the

parameter dmg1 does not exist, pa1 will create it and set it to 'pa1'.

In the phase angle mode, each real point in the displayed trace is calculated from the phase angle of the real and imaginary points comprising each respective complex data point. For hypercomplex data, the phase angle uses the real-real and imaginary-real points from each respective hypercomplex data point. The phase angle also takes into account the phase parameters rp1 and lp1.

The pal command is only needed if mixed-mode display is desired. If the parameter <code>dmgl</code> does not exist or is set to the null string, the display mode along the first indirectly detected dimension defaults to the display mode of the directly detected dimension (characterized by the parameter <code>dmg</code>). For the contour display of multidimensional data, the result of <code>pal</code> is the same as for traces provided that <code>pmode='partial'</code> or <code>pmode=''</code>.

See also: NMR Spectroscopy User Guides

Related: av1 Set abs. value mode in 1st indirectly detected dimension (C)

dmg1 Data display mode in 1st indirectly detected dimension (P)

lp1 First-order phase in 1st indirectly detected dimension (P)

pa Set phase angle mode in directly detected dimension (C)

ph1 Set phased mode in 1st indirectly detected dimension (C)

pmode Processing mode for 2D data (P)

pwr1 Set power mode in 1st indirectly detected dimension (C)
rp1 Zero-order phase in 1st indirectly detected dimension (P)

# pacosy Plot automatic COSY analysis (C)

Description: Automatically analyzes and plots a COSY data set with fn=fn1 and sw=sw1.

Symmetrization of the data with the command <code>foldt</code> is recommended, but not required. First, select a proper threshold and perform a 2D line listing with the command <code>ll2d</code>. Next, plot the 2D data with the contour plot command <code>pcon</code>; leaving enough room at the left side of the plot for the connectivity table. Then, <code>pacosy</code> will analyze the data and plot the connectivities on the plotter. <code>pacosy</code> gets its input from the file <code>ll2d.out</code> in the current experiment directory. The command <code>acosy</code> performs the same analysis and displays the

connectivities on the screen.

See also: NMR Spectroscopy User Guide

Related: acosy Automatic analysis of COSY data (C)

fn Fourier number in directly detected dimension (P)
fn1 Fourier number in 1st indirectly detected dimension (P)
foldt Fold COSY-like spectrum along diagonal axis (C)
hcosy Automated proton and COSY acquisition (M)
l12d Automatic and interactive 2D peak picking (C)

Plot contours on plotter (C)

relayh Set up parameters for COSY pulse sequence (M)
sw Spectral width in directly detected dimension (P)
sw1 Spectral width in 1st indirectly detected dimension (P)

## pad Preacquisition delay (P)

Description: Each NMR experiment starts with a single delay time equal to pad over and

above the delay d1 that occurs before each transient. Normally, pad is set to a small, nominal time (0.5 seconds) to allow any hardware changes that may be required at the start of the acquisition to "settle in." During experiments in which the temperature is changed, the acquisition starts pad seconds after the temperature regulation system comes to regulation. Since the sample temperature does not actually come to equilibrium for some time after that, it is generally desirable to increase pad to perhaps 300 seconds. This is especially true when running experiments involving arrays of temperatures. The pad parameter is most useful for running kinetics experiments. For example, pad=0,3600,3600,3600,3600 will run an experiment immediately when go is typed (pad=0), then wait an hour (3600 seconds), run the second experiment, etc.

experiment, etc.

Values:  $0.0.1 \,\mu s$  to 8190 sec in 12.5 ns steps

 $0.0.2 \,\mu s$  to  $150.000 \, sec$  in  $0.1 \,\mu s$  steps.

See also: NMR Spectroscopy User Guide; VnmrJ Walkup

Related: d1 First delay (P)

go Submit experiment to acquisition (C)

# padept Perform adept analysis and plot resulting spectra (C)

Syntax: padept<(<'noll'><,'coef'><,'theory'>)>

Description: Performs the adept analysis and plots the resulting spectra with a scale and the

assigned line listing. Leave enough space at the left end of the display for the

line list.

Arguments: The following arguments can be supplied in any order:

'noll' is a keyword that specifies no line listing.

'coef' is a keyword that causes the combination coefficients to be printed.

 $\mbox{'theory'}$  is a keyword that causes the theoretical coefficients rather than

optimized coefficients to be used.

Examples: padept('noll','coef')

See also: NMR Spectroscopy User Guide

Related: adept Automatic DEPT analysis and spectrum editing (C)

autodeptAutomated complete analysis of DEPT data (M)cdeptAutomated carbon and DEPT acquisition (C)DeptSet up parameters for DEPT experiment

deptproc Process DEPT data (M)

hcdept Automated proton, carbon, and DEPT acquisition (C)

pldept Plot DEPT data, edited or unedited (M)

# page Submit plot and change plotter page (C)

Syntax: page<(number pages<,'clear'|file>)>

Description: Submits the current plotter file, which has been created by all previous plotter

commands, and changes the paper after the plot has been completed. Actual plotting is controlled by the vnmrplot script in the bin subdirectory of the system directory. The page command can also clear the current plotter file or

save the data to a specified file name.

Arguments: number pages is the number of pages to move the plotter forward. The

default is 1. If number\_pages is 0, page submits the plot but does not

change the paper.

'clear' is a keyword to clear the plot made thus far; that is, clear the data in

the current plotter file.

file is the name of a file to save the plot for import into a document. If the file

already exists, it is overwritten.

Examples: page

page(0)

page('clear')

page('myplotfile')

See also: NMR Spectroscopy User Guide

Related: vnmrplot Plot files (U)

page Name of page (P)

Description: Specifies the page of a sample. It is saved with a study.

Related: cqsavestudy Macro to save study queue parameters (M)

notebookNotebook name (P)samplenameSample name (P)studyparStudy parameters (P)

panellevel Display level for VnmrJ interface pages (P)

Description: Determines which VnmrJ interface pages are available under the tabs in the

parameter page area. The higher the number, the more pages are available. The only time panellevel is changed is during the login process of an operator in the Walkup interface. For the Walkup interface, the value is set by the VnmrJ

Administrator (default is 10).

Values: **0-9** — shows the minimum number of pages.

No shim, lock, or processing, and minimal parameter control is available. This

may be used for routine automation users.

10-29 — typical for a basic Walkup user.

Shim and lock are available only if there is a sample changer. Basic processing is available. Pages are not fully populated, allowing control of a few basic

parameters.

**30-100** — typical for the system owner.

All pages are available and fully populated.

See also: VnmrJ Installation and Administration

Related: operator Operator name (P)

 $\begin{tabular}{ll} \textbf{Operatorlogin} & \textbf{Sets workspace and parameters for the operator } \textbf{(M)} \\ \end{tabular}$ 

pap Plot out "all" parameters (C)

Syntax: pap<(<template><, x><, y><, character\_size>)>

Description: Plots a parameter list containing "all" parameter names and values.

Arguments: template is the name of a template that controls the display. The default is

the string parameter ap, which can be modified using paramvi ('ap'). See

the manual User Programming for rules on building a template.

x is the starting position in the x direction of the plot on the paper, in mm. The

default is a preset value.

y is the starting position in the y direction of the plot on the paper, in mm. If y

is specified, the x position must be also. The default is a preset value.

character size is the character size of the list and is specified as a

multiplier. The default is 0.70 (not available on all plotters or printers acting as

plotters).

Examples: pap

pap(wcmax-40)

pap(10,wc2max\*.9)

pap('newpap',wcmax-50,100,1.4)

See also: NMR Spectroscopy User Guide, User Programming

Related: ap Print out "all" parameters (C)

"All" parameters display control (P)

hpa Plot parameters on special preprinted chart paper (C)

paramvi Edit a variable and its attributes using vi text editor (M)

ppa Plot a parameter list in "English" (M)

# par2d Create 2D acquisition, processing, and display parameters (M)

Description: Creates the acquisition parameters ni, swl, and phase, which can be used to

acquire a 2D data set. par2d also creates any missing processing and display parameters for the ni (or second) dimension, including flcoef, reffrql, refpos1, and refsource1. The par2d macro is functionally the same as

addpar('2d').

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

flcoef Coefficient to construct F1 interferogram (P)

Number of increments in 1st indirectly detected dimension (P)

phase Phase selection (P)

reffrq1 Reference frequency of reference line in 1st indirect dimension (P)

refpos1 Position of reference line in 1st indirect dimension (P)
refsource1 Center frequency in 1st indirect dimension (P)
set2d General setup for 2D experiments (M)

Spectral width in 1st indirectly detected dimension (P)

#### par3d Create 3D acquisition, processing, and display parameters (M)

Description: Creates the acquisition parameters ni2, sw2, d3, and phase2 that can be used

to acquire a 3D data set. par3d also creates any missing processing or display parameters for the ni2 (or third) dimension, including f2coef, fiddc3d, specdc3d, and ptspec3d. The par3d macro is functionally the same as

addpar('3d').

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

d3 Incremented delay in 2nd indirectly detected dimension (P)

f2coef Coefficient to construct F2 interferogram (P)

fiddc3d 3D time-domain dc correction (P)

Number of increments in 2nd indirectly detected dimension (P)

phase2 Phase selection for 3D acquisition (P)
 ptspec3d Region-selective 3D processing (P)
 specdc3d 3D spectral dc correction (P)

Spectral width in 2nd indirectly detected dimension (P)

#### par3rf Get display templates for 3rd rf channel parameters (M)

Applicability: Systems with a second decoupler.

Description: Retrieves the dg2 and modified ap display templates from the parameter set

s2pul3rf in the system parlib directory. These two templates support the display of second decoupler acquisition parameters and 3D acquisition and

processing parameters.

See also: User Programming

Related: ap "All" parameters display control (P)

dg2 Control dg2 parameter group display (P)

# par4d Create 4D acquisition parameters (M)

Applicability: Systems with a third decoupler.

Description: Creates the acquisition parameters ni3, sw3, d4, and phase3 that can be used

to acquire a 4D data set. The par4d macro is functionally the same as

addpar('4d').

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

d4 Incremented delay for 3rd indirectly detected dimension (P)

ni3 Number of increments in 3rd indirectly detected dimension (P)

phase3 Phase selection for 4D acquisition (P)

Spectral width in 3rd indirectly detected dimension (P)

## paramedit Edit a parameter and its attributes with user-selected editor (C)

Syntax: paramedit(parameter<,tree>)

Description: Opens a parameter file for editing with a user-selected text editor. The default

editor is vi. If vi is used as the editor, paramedit is functionally the same

as the paramvi command. To select another editor, set the UNIX

environmental variable  ${\tt vnmreditor}$  to the editor name (change  ${\tt .login}$ 

line setenv vnmreditor old editor to become setenv

vnmreditor new\_editor (e.g., setenv vnmreditor emacs) and make sure a script with the prefix vnmr\_ followed by the name of the editor is placed in the bin subdirectory of the system directory (e.g., vnmr\_emacs). The script file makes adjustments for the type of graphic interface in use.

Scripts in the software release include vnmr\_vi and vnmr\_textedit. To create other scripts, refer to the vnmr\_vi script for non-window editor interfaces and to vnmr\_textedit for window-based editor interfaces. The

vnmreditor variable must be set before starting VnmrJ.

Arguments: parameter is the name of the parameter file to be edited.

tree is a keyword for one of the parameter trees 'current', 'global', or

'processed'. The default is 'current'.

Examples: paramedit('ap')

paramedit('b','global')

See also: NMR Spectroscopy User Guide; User Programming

Related: paramvi Edit a parameter and its attributes with vi editor (M)

vi Edit text file with the vi text editor (C)

## paramvi Edit a parameter and its attributes with vi editor (M)

Syntax: paramvi(parameter<,tree>)

Description: Opens a parameter file for editing using the UNIX vi text editor. The parameter

file contains various attributes of the parameter in a format documented in the manual *User Programming*. Be sure you understand the format before

modifying the parameter because if an error in the format is made, the parameter will not load. When the editor is exited, the modified parameter is reloaded into

the system.

Arguments: parameter is the name of the parameter file to be edited.

tree is a keyword for one of the parameter trees 'current', 'global',

or 'processed'. The default is 'current'.

Examples: paramvi('ap')

paramvi('b','global')

See also: NMR Spectroscopy User Guide, User Programming

Related: create Create new parameter in a parameter tree (C)

destroy Destroy a parameter (C)

destroygroup Destroy parameters of a group in a tree (C)
display Display parameters and their attributes (C)

**fread** Read parameters from file and load them into a tree (C)

**fsave** Save parameters from a tree to a file (C)

groupcopy Copy parameters of group from one tree to another (C)

paramedit Edit a parameter and its attributes with user-selected editor (C)

prune Prune extra parameters from current tree (C)
setgroup Set group of a parameter in a tree (C)
setlimit Set limits of a parameter in a tree (C)
setprotect Set protection mode of a parameter (C)
vi Edit text file with the vi text editor (C)

## pards Create additional parameters used by downsampling (M)

Description: Creates the parameters downsamp, dscoef, dsfb, dslsfrq, and

filtfile necessary for digital filtering and downsampling. The pards

macro is functionally the same as addpar ('downsamp').

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to current experiment (M)

downsampDownsampling factor applied after digital filtering (P)dscoefDigital filter coefficients for downsampling (P)dsfbDigital filter bandwidth for downsampling (P)dslsfrqBandpass filter offset for downsampling (P)filtfileFile of FIR digital filter coefficients (P)

movedssw Set downsampling parameters for selected spectral region (M)

#### parfidss Create parameters for time-domain solvent subtraction (M)

Description:

Creates solvent subtraction parameters ssfilter, sslsfrq, ssntaps, and ssorder. Entering addpar('ss') is functionally equivalent to parfidss.

In a 1D transform, subtraction of the zero-frequency component from the time-domain data, usually in the context of solvent subtraction, is selected by setting ssorder and ssfilter to desired values and entering wft:

- The zfs (zero-frequency suppression) option is selected if both ssfilter and ssorder are set to a value other than "Not Used."
- The lfs (low-frequency suppression) option is selected if ssfilter is set to a value other than "Not Used" and ssorder is set to "Not Used."
- The zfs and lfs options are both turned off if ssfilter is set to "Not Used."

The zfs option leads to the following series of processing events: (1) the raw FID is frequency-shifted by sslsfrq Hz, (2) the raw FID is subjected to a low-pass digital filter, (3) the filtered FID is fit to a polynomial of order ssorder, (4) the polynomial function is subtracted from the raw FID, and (5) the resulting FID is frequency-shifted by -sslsfrq Hz.

The Ifs option does not include a polynomial fit (step 3 of the zfs option), which leads to the following series of processing events: (1) the raw FID is frequency-shifted by sslsfrq Hz, (2) the raw FID is subjected to a low-pass digital filter, (3) the filtered FID is directly subtracted from the raw FID, (4) the resulting FID is frequency-shifted by -sslsfrq Hz.

The quality of filtering with zfs diminishes rapidly as the solvent peak moves off the exact center of the digital filter. It may be necessary to adjust lsfrq or sslsfrq to move the solvent peak to within  $\pm 0.2$  Hz of the center of the filter to obtain optimal solvent suppression. The lfs option is less sensitive to small offsets, but typically removes or distorts peaks near to the solvent peak.

In a 2D transform, solvent correction to the t<sub>2</sub> FIDs is invoked in the same manner with the ftld, ftld, wftld, and wftld commands and with the ftlda, wftlda, and wftlda macros.

In a 3D transform, solvent suppression works on t<sub>3</sub> FIDs of 3D spectra just like in the 1D and 2D cases.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

ft Fourier transform 1D data (C)

ftld Fourier transform along f<sub>2</sub> dimension (C)

ft2d Fourier transform 2D data (C)

Perform a 3D Fourier transform on a 3D FID data set (M,U)

lsfrq Frequency shift of the fn spectrum in Hz (P)

ntype3d N-type peak selection in  $f_1$  or  $f_2$  (P)

ssfilter
Full bandwidth of digital filter to yield a filtered FID (P)

sslsfrq
Center of solvent-suppressed region of spectrum (P)

ssorder
Order of polynomial to fit digitally filtered FID (P)

ssntaps
Number of coefficients to be used in the digital filter (P)

wft Weight and Fourier transform 1D data (C)

#### parfix Update parameter sets (M)

Description: Corrects upper limits, lower limits, and step sizes of a number of parameters in

the current experiment. In addition, the template parameter dgs is updated.

This is automatically done via the macro fixpar if the parameter

parversion is less than 4.3. parfix is used by the macro updatepars to correct saved data. This macro has been applied to all parameters as of VNMR version 4.3 and should be run on older parameter sets (e.g., rtp('pars')

svp('pars') update a parameter set named pars).

See also: NMR Spectroscopy User Guide

Related: ap "All" parameters display control (P)

dgs Control dgs parameter group display (P)

fixpar Correct parameter characteristics in experiment (M)

parversion Version of parameter set (P)

updatepars Update all parameter sets saved in a directory (M)

#### parlc Create parameters for LC-NMR experiments (M)

Applicability: Systems with LC-NMR accessory.

Description: Creates the following parameters used for a variety of LC-NMR experiments:

curscan, dtrig, inject, ntrig, and savefile. The parlc macro also creates ni and sw1 (if they don't exist) for use in isocratic runs. Finally, it creates a display parameter dglc, so that the dg('dglc') command (or the equivalent macro dglc) can be used to display all the LC-related parameters.

Note that parlc can be used without worrying about losing existing values or attributes; if the parameters already exist, they are left untouched.

See also: NMR Spectroscopy User Guide

Related: curscan Scan currently in progress (P)

dglc Control LC-NMR parameter display (P)

dtrig Delay to wait for another trigger or acquire a spectrum (P)

inject Trigger the injection of a sample (P)

ntrig Number of trigger signals to wait before acquisition (P)

Base file name for saving FIDs or data sets (P)

#### parl12d Create parameters for 2D peak picking (M)

Description: Creates additional parameters th2d and xdiag for use with 112d 2D peak

picking program. parl12d is functionally the same as addpar('112d').

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

Automatic and interactive 2D peak picking (C)

th2d Threshold for integrating peaks in 2D spectra (P)

xdiag Threshold for excluding diagonal peaks when peak picking (P)

#### parlp Create parameters for linear prediction (M)

Syntax: parlp<(dimension)>

Description: Creates parametrized options for linear prediction (LP) in the current

experiment. The display template for the dglp macro is also created if

necessary. parlp is functionally the same as addpar('lp').

Arguments: dimension is the dimension of a multidimensional data set. The default is to

create the LP parameters lpalg, lpopt, lpfilt, lpnupts, strtlp,

lpext, strtext, lptrace, and lpprint.

parlp(1) creates LP parameters lpalg1, lpopt1, lpfilt1,

lpnupts1, strtlp1, lpext1, strtext1, lptrace1, and lpprint1.

 ${\tt addpar('lp',1)} \ \ is \ functionally \ equivalent \ to \ {\tt parlp(1)}.$ 

parlp(2) creates LP parameters lpalg2, lpopt2, lpfilt2,
lpnupts2, strtlp2, lpext2, strtext2, lptrace2, and lpprint2.

addpar('lp', 2) is functionally equivalent to parlp(2).

Examples: parlp

parlp(1)

See also: NMR Spectroscopy User Guide

Related: lpalg LP algorithm for np dimension (P)

lpext LP data extension for np dimension (P)

lpfilt LP coefficients to calculate for np dimension (P)
lpnupts LP number of data points for np dimension (P)
lpopt LP algorithm data extension for np dimension (P)

proc1 Type of processing on ni interferogram (P)
proc2 Type of processing on ni2 interferogram (P)

Starting point for LP data extension for np dimension (P)
Starting point for LP calculation for np dimension (P)

#### parmax Parameter maximum values (P)

Description: An array that holds the maximum values of other parameters. The maximum

value of a parameter is an index into the array, and more than one parameter can have the same index into parmax. Several global parameters set in the Spectrometer Configuration window are part of parmax. To display all parmax values, enter display ('parmax', 'systemglobal').

See also: *User Programming* 

Related: config Display current configuration and possibly change it (M

display Display parameters and their attributes (C)

paramedit Edit a parameter and its attributes with user-selected editor (C)
paramvi Edit a parameter and its attributes using vi text editor (M)

parmin Parameter minimum values (P)
parstep Parameter step size values (P)

#### parmin Parameter minimum values (P)

Description: An array that holds the minimum values for other parameters. The minimum

value of a parameter is the index into the parmin array. More than one parameter may have the same index into the array. To display all the values in

parmin, enter display('parmin', 'systemglobal').

See also: User Programming

Related: paramvi Edit a parameter and its attributes using vi text editor (M)

display Display parameters and their attributes (C)

paramedit Edit a parameter and its attributes with user-selected editor (C)

parmax Parameter maximum values (P)
parstep Parameter step size values (P)

#### paros Create additional parameters used by oversampling (M)

Description: Creates the parameters def osfilt, filtfile, oscoef, osfb,

osfilt, oslsfrq, and oversamp for oversampling and digital filtering.

paros is functionally the same as addpar ('oversamp').

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to current experiment (M

def\_osfiltDefault value of osfilt parameter (P)filtfileFile of FIR digital filter coefficients (P)oscoefDigital filter coefficients for oversampling (P)osfbDigital filter bandwidth for oversampling (P)osfiltOversampling filter for real-time DSP (P)oslsfrqBandpass filter offset for oversampling (P)oversampOversampling factor for acquisition (P)

## parstep Parameter step size values (P)

Description: An array that holds the step size values for other parameters. The step size value

of a parameter is the index into the array. More than one parameter can have the same index into parstep. Several configuration parameters set in the Spectrometer Configuration window are part of parstep. To display all parstep values, enter display ('parstep', 'systemglobal').

See also: User Programming

Related: config Display current configuration and possibly change it (M)

display Display parameters and their attributes (C)

paramedit Edit a parameter and its attributes with user-selected editor (C)
paramvi Edit a parameter and its attributes using vi text editor (M)

parmax Parameter maximum values (P)
parmin Parameter minimum values (P)

parversion Version of parameter set (P)

Description: Stores the version of a parameter set. When a parameter set is updated with

updatepars or parfix, parversion is set to 4.3 to indicate that fact. When a parameter set is retrieved into an experiment, fixpar checks parversion to determine if other parameters need to be updated using

parfix.

See also: NMR Spectroscopy User Guide

Related: fixpar Correct parameter characteristics in experiment (M)

parfix Update parameter sets (M)

updatepars Update all parameter sets saved in a directory (M)

## path3d Path to currently displayed 2D planes from a 3D data set (P)

Description: Stores the absolute path to the current 3D data directory tree. If path3d does

not exist, it is created by the macro par3d. The command select, as well as the many macros that make use of select, require path3d in order to know

where the 2D planes extracted from a 3D data set can be found.

path3d is set automatically by the macros ft3d and getplane:

• ft3d sets path3d to curexp/datadir3d if ft3d is not supplied with a directory path for the transformed 3D data. If ft3d is supplied with such a directory path (e.g., /home/data/test3D), path3d is set equal to that directory path. In this case, the 3D spectral data would reside in the directory /home/data/test3D/data.

• getplane sets path3d to curexp/datadir3d if getplane is not supplied with a directory path to the transformed 3D data. If getplane is supplied with such a directory path (e.g., /home/data/test3D), path3d is set equal to that directory path. In this case, the extracted 3D planes would reside in the directory /home/data/test3D/extr.

See also: NMR Spectroscopy User Guide

Related: dplane Display a 3D plane (M)

dproj Display a 3D plane projection (M)
dsplanes Display a series of 3D planes (M)

Perform a 3D Fourier transform on a 3D FID data set (M)

getplane Extract planes from a 3D spectral set (M)

nextpl Display the next 3D plane (M)

create 3D acquisition, processing, display parameters (C)

Currently displayed 3D plane type (P)

plplanes Plot a series of 3D planes (M)
prevpl Display the previous 3D plane (M)

select Select a spectrum or 2D plane without displaying it (C)

# paxis Plot horizontal LC axis (M)

Applicability: Systems with the LC-NMR accessory.

Syntax: paxis(time, major tic, mino tic)

Description: Plots a horizontal LC axis. Horizontal axes are assumed to be used with "LC

plots" of an entire LC run are labeled accordingly. It is assumed that relevant parameters (e.g., sc, wc, vo, vp) have not been changed after plotting the data.

Arguments: time is the time scale, in minutes (decimal values are fine), of the axis.

major\_tic is spacing, in minutes (decimal values are fine), of major tics.
minor\_tic is spacing, in minutes (decimal values are fine), of minor tics.

See also: NMR Spectroscopy User Guide

Pbox Pulse shaping software (U)

Syntax: Pbox file options

Description: Main Pbox (Pandora's Box) program for the generation of shape files for

RF and gradients. (See NMR Spectroscopy User Guide manual for description

of interactive Pbox usage).

Arguments: file is the name of a shape file.

options is any of the Pbox parameters initialized by the ' - ' sign and followed by the parameter value. The following options can be in any order and

combinations:

-b time Activates Bloch simulator, sets simtime, in sec.

-c Calibrate only, do not create a shape file.

f file
Set name of the output file.
h wave
Print wave file header.
i wave
Print wave file parameters.

-1 ref pw90 Length, in µs, of reference pw90 pulse.

-o List options.

-p ref pwr Reference power level, in dB.

-r file Reshape Pbox pulse.

-s stepsize Define length, in µs, of a single step in waveform.

-t wave Print wave title.
-w wavestr Set wave data string.

-v Run in verbose mode. Also print Pbox version.

-value Sets reps to value.

Examples: Pbox -i eburp2

Pbox newshape -wc 'eburp1 450 -1280.0' -1

Pbox sel.RF -w 'eburp1 420 -800' 'eburp1 420 1200' Pbox -w 'eburp1 200 -1200' -attn e -p1 45 54.2 -b Pbox tst -w 'esnob 20p 170p' -sfrq 150.02 -refofs 55p

-ref pwr 45 -ref pw90 54.2

See also: NMR Spectroscopy User Guide

Related: Cpx Create Pbox shape file (M)

dprofile Display pulse excitation profile from Pbox software (M)

dshape Display pulse shape (M)

dshapefDisplay last generated pulse shape (M)dshapeiDisplay pulse shape interactively (M)opxOpen shape definition file for Pbox (M)

pbox\_bw Define excitation band (M)

pbox\_bws Define excitation band for solvent suppression (notch) pulses (M)

pbox\_dmfExtract dmf value from Pbox shape file (M)pbox\_dresExtract dres value from Pbox shape file (M)

pbox\_nameExtract name of last shape file generated by Pbox (M)pbox\_pwExtract pulse length from Pbox shape file (M)pbox\_pwrExtract pulse power from Pbox shape file (M)pbox\_pwrfExtract pulse fine power from Pbox (M)

pboxget Extract all calibration data from a Pbox shape file (M)

pboxpar Add parameter definition to the pbox.inp file (M)
pboxrst Reset temporary Pbox/VnmrJ variables (M)

pboxunits Converts to Pbox default units (M)

Print pulse header (M)

pprofile Plot pulse excitation profile from Pbox software (M)

pshape Plot pulse shape (M)

pshapef Display pulse shape or modulation pattern interactively (M)

putwave Write a wave into Pbox.inp file (M)

Assign Pbox calibration data to experimental parameters (M)

pxshape Generates a single-band shape file (M)

Pxsim Simulate Bloch profile for a shaped pulse (M)

Pxspy Create shape definition using Fourier coefficients (U)

selex Defines excitation band (M)

Setwave Sets a single excitation band in Pbox.inp file (M) shdec Shaped observe excitation sequence (M)

## pbox bw Define excitation band (M)

Syntax: pbox\_bw<(shapename)>

Description: Defines the excitation band from the position of cursors in the graphics window

and reports them to user. It also sets r1 to excitation bandwidth and r2 to offset.

This macro is used mainly in Pbox menus and macros.

Arguments: shapename is the name of a shape as in wavelib; mainly for use with

menus.

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

#### pbox bws Define excitation band for solvent suppression (notch) pulses (M)

Syntax: pbox bws<(shapename)>

Description: Defines the excitation band from the position of cursors in the graphics window

and reports them to user. It also sets r1 to excitation bandwidth and r2 to offset. Note, the left cursor should be placed on the left side of the excitation band and the right cursor on resonance of the solvent signal. This macro is mainly used in

Pbox menus and macros.

Arguments: shapename is the name of a shape file as in wavelib, mainly for use with

menus.

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

## pbox dmf Extract dmf value from pbox.cal or Pbox shape file (M)

Syntax: pbox dmf<(shapefile.DEC)>:exp param

Description: Extracts the dmf value from the file shapefile.DEC created by Pbox or, if

file name is not provided, from the pbox.cal file containing parameters of the

last created Pbox shape file.

Arguments: shapefile.DEC is the name of a shape file.

exp\_param is a dmf type experiment parameter.

Examples: pbox\_dmf('myfile.DEC'):mydmf

pbox\_dmf:dmf2

See also: NMR Spectroscopy User Guide

Related: dmf Decoupler modulation frequency for first decoupler (P)

Pbox Pulse shaping software (U)

pbox dres Extract dres value from pbox.cal or Pbox shape file (M)

Syntax: pbox\_dres<(shapefile.DEC)>:exp\_param

Description: Extracts the dres value from the file shapefile.DEC created by Pbox or,

if file name is not provided, from the Pbox. cal file containing parameters of

the last created Pbox shape file.

Arguments: shapefile.DEC is the name of a shape file.

exp param is a dres type experiment parameter.

Examples: pbox dres('myfile.DEC'):mydres

pbox dres:dres2

See also: NMR Spectroscopy User Guide

Related: dres Tip-angle resolution for first decoupler (P)

Pbox Pulse shaping software (U)

pbox name Extract name of last shape generated by Pbox from pbox.cal (M)

Syntax: pbox\_name:exp\_name

Description: Extracts name of the last shape file generated by Pbox and stored in the

Pbox.cal file. Note, that the file name extension is not stored explicitly and

is not provided by this macro.

Arguments: exp name returns the name of last shape file.

Examples: pbox\_pw:shname

pbox pw:pwpat

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

pbox pw Extract pulse length from pbox.cal or Pbox shape file (M)

Syntax: pbox\_pw<(shapefile.RF)>:exp\_param

Description: Extracts pulse length from the file shapefile.RF generated by Pbox or, if

file name is not provided, from pbox.cal file containing parameters of the

last created Pbox shape file. Returns the pulse length, in  $\mu s. \label{eq:pbox}$ 

Arguments: shapefile.RF is the shape file name, including the extension.

exp param is a pw type experiment parameter.

Examples: pbox\_pw('myfile.RF'):softpw

pbox\_pw:selpw

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

pbox pwr Extract power level from Pbox.cal or Pbox shape file (M)

Syntax: pbox\_pwr<(shapefile.ext)>:exp\_param

Description: Extracts the power lever from the file shapefile.ext generated by Pbox

or, if file name is not provided, from the pbox.cal file containing parameters

of the last created Pbox shape file. Returns the power level, in dB. The

exp param parameter will not be changed by this macro if the parameter is

previously set to 'n' (not used).

Arguments: shapefile.ext is the name of the shape file.

exp param is a power type experiment parameter.

Examples: pbox pwr('myfile.DEC'):mypwr

pbox pwr:dpwr2

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

#### pbox pwrf Extract fine power level from pbox.cal or Pbox shape file (M)

Syntax: pbox\_pwrf<(shapefile.ext)>:exp\_param

Description: Extracts the fine power lever from the file shapefile.ext generated by

Pbox or, if file name is not provided, from the pbox.cal file containing parameters of the last created Pbox shape file. Returns the value of fine power, in dB. Note that the parameter will not be changed by this macro if it was

previously set to 'n' (not used).

Arguments: shapefile.ext is the name of the shape file.

exp param is a fine power type experiment parameter.

Examples: pbox pwrf('myfile.DEC'):mypwrf

pbox pwrf:dpwrf

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

#### pboxget Extract Pbox calibration data (M)

Syntax: pboxget<(shfile.ext)>:\$name,\$pw,\$pwr,\$pwrf,\$dres,\$dmf

Description: Extracts calibration data from the file shfile.ext generated by Pbox or, if

a file name is not provided, from the pbox.cal file containing parameters of the last created Pbox shape file. Returns shape name and the values of total pulse length (in  $\mu$ s), power (dB), fine power, dres, and dmf. The parameter will not be changed by this macro if the parameter was previously set to 'n'

(not used).

Arguments: shfile.ext is the name of the shape file, including the extension.

name is the experiment parameter receiving the shape name (without the

extension).

pw is the experiment parameter receiving the total pulse length, in  $\mu$ s. pwr is the experiment parameter receiving the power level, in dB. pwrf is the experiment parameter receiving the fine power level. dres is the experiment parameter receiving the decoupler resolution.

dmf is the experiment parameter receiving the decoupler modulation frequency.

Examples: pboxget('myfile.DEC'):dseq,r1,dpwr,dpwrf,dres,dmf

pboxget('selshape.RF'):pwpat,selpwr
pboxget:dseq2,r1,dpwr2,dpwrf2,dres2,dmf2

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

pboxpar Add parameter definition to the Pbox.inp file (M)

Syntax: pboxpar(param, value)

Description: Adds a parameter definition to the Pbox.inp file.

Arguments: param is the parameter name

value is the value of the parameter.

Examples: pboxpar('name','myfile.DEC')

pboxpar('bsim','y')
pboxpar('T1', 0.24)

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

pboxrst Reset temporary Pbox variables (M)

Description: Resets r1=0, r2=0, r3=0, r4=0, n2='n', n3='', and adds some standard

comment lines to the Pbox.inp file. This macro is used in menus and other

Pbox macros.

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

pboxunits Converts to Pbox default units (M)

Syntax: pboxunits

Description: Used by Pbox menus to scale parameters related to time or frequency down to

Pbox default units (Hz or seconds) before the parameter is stored in the

Pbox.inp file.

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

pcon Plot contours on a plotter (C)

Syntax: pcon<(<'pos'|'neg'><,'noaxis'><,levels><,spacing>)>

Description: Plots positive and negative peaks of a contour plot display using different

colors. Specifically, if maxpen is set for n pens, positive peaks are plotted using colors 1 through (n+1)/2, and negative peaks are plotted using colors ((n+1)/2)+1 through n (i.e., half the colors for each, plus one extra for positive if an odd number of pens is specified). Pen 1 is always used for the axes, and the lowest contour of the positive peaks is also plotted with pen1. In all cases, the pen colors are cycled if more contours are to be plotted than there are pens available.

To plot both negative and positive contours of a phase-sensitive spectrum on a monochrome device such as a LaserJet or a plotter with a single pen, different numbers of contours may be plotted for the different sign. For example, pcon('pos',10,1.4) pcon('neg',1) will plot ten closely spaced

positive contours and one negative contour.

Arguments: 'pos' is a keyword specifying that phase-sensitive spectra plot positive peaks

only. The default is to plot both positive and negative peaks.

'neg' is a keyword specifying that phase-sensitive spectra plot negative peaks

only. The default is to plot both positive and negative peaks.

'noaxis' is a keyword to omit outlining the plot and omit plotting the

horizontal and vertical axes.

levels is maximum number of contour levels to plot. The default is 4.

spacing is relative intensity of successive contour levels. The default is 2.

Examples: pcon

pcon(4,1.4)

pcon('pos','noaxis')

pcon('neg',3)

See also: NMR Spectroscopy User Guide

Related: dpcon Display plotted contours (C)

maxpen Maximum number of pens to use (P)

## pcss Calculate and show proton chemical shifts spectrum (M)

Syntax: pcss<(<threshold><,max cc><,max width>)>

Description: Calculates and shows the proton chemical shifts spectrum. The dsp command

is used to display the results. The list of chemical shifts is saved in the file pcss.outpar. The original spectrum can be calculated by the wft

command.

Arguments: threshold sets the level whether a point belongs to a peak or is noise. The

default is that pcss automatically calculates the threshold.

max cc is the maximum allowable coupling constant in the spectrum. The

default is 20 Hz.

max width is the maximum width of a spin multiplet in the spectrum. The

default is 60 Hz.

Examples: pcss

pcss(10)

pcss(9,20,80)

See also: NMR Spectroscopy User Guide

Related: do\_pcss Calculate proton chemical shifts spectrum (C)

dsp Display pulse sequence (C)

wft Weight and Fourier transform 1D data (C)

#### peak Find tallest peak in specified region (C)

Syntax: peak<(min freq, max freq) ><:height, freq>

Description: Returns the height and frequency of the tallest peak in the selected region,

including any referencing (i.e., the same frequency that you would measure by placing a cursor on the peak). A spectrum need not actually be displayed for

peak to work.

Arguments: With no return arguments, peak displays on the screen information about peak

height and frequency. If two cursors are displayed, peak without arguments

finds the tallest peak between the cursors.

min\_freq is minimum frequency limit of the region to be searched. The

default value is sp.

max freq is maximum frequency limit, in Hz, of the region to be searched.

The default value is sp + wp.

height returns the height, in mm, of the tallest peak in the selected region. freq returns the frequency, in Hz, of the tallest peak in the selected region.

Examples: peak: \$ht, \$freq

peak(0,2000):r3
peak:\$ht,cr

See also: *User Programming* 

Related: sp Start of plot (P)

WP Width of plot (P)

## peak2d Return information about maximum in 2D data (C)

Syntax: peak2d:\$maximum\_intensity<,\$trace,\$point>

Description: Searches the area defined by sp, wp, sp1, and wp1 in a 2D data set for a

maximum intensity.

Arguments: \$maximum intensity returns the maximum intensity value found.

\$trace returns the trace number of the maximum. The parameter trace

defines whether f<sub>1</sub> or f<sub>2</sub> traces are counted.

\$point returns the data point number of the maximum on that trace.

See also: NMR Spectroscopy User Guide

Related: sp Start of plot (P)

Start of plot in 1st indirectly detected dimension (P)

trace Mode for *n*-dimensional data display (P)

wp Width of plot (P)

wp1 Width of plot in 1st indirectly detected dimension (P)

#### pen Select a pen or color for drawing (C)

Syntax: pen(<'graphics'|'plotter',><'xor'|'normal',>

pen | color)

Description: Selects the pen number for a plotter or the color for the graphics screen. This

command is part of a line drawing capability that includes the move and draw commands. move sets the coordinates from which the line starts. draw draws a line from that point to the new coordinates specified by draw. Refer to the description of draw for examples of using the line drawing capability.

Arguments: 'graphics' and 'plotter' are keywords selecting the output device. The

default is 'plotter'. The output selected is passed to subsequent pen, move, or draw commands and remains active until a different output is

specified.

'xor' and 'normal' are keywords selecting the drawing mode for the 'graphics' output device. In the 'xor' mode, if a line is drawn such that one or more points of the line are in common with a previously drawn line, the common points are erased. In the 'normal' mode, the common points remain. The mode selected is passed to subsequent pen, draw, or move commands and remains active until a different mode is specified. The default

mode is 'normal'.

pen is the plotter pen number: 'pen1', 'pen2', 'pen3', etc.

color is the active color for the graphics screen: 'red', 'green',
'blue', 'cyan', 'magenta', 'yellow', 'black', or 'white'.

Examples: pen('pen2')

pen('graphics','red')

See also: NMR Spectroscopy User Guide

Related: draw Draw line from current location to another location (C)

move Move to an absolute location (C)

## pexpl Plot exponential or polynomial curves (C)

Syntax: pexpl<(<options,><line1,line2, ...)>

Description: Plots exponential curves resulting from  $T_1$ ,  $T_2$ , or kinetics analysis. Also plots

polynomial curves from diffusion or other types of analysis. The

analyze.out file is the data input file used to make the plot. Refer to the expl entry for the format of this file. The parameters sc, wc, sc2, and wc2

control the size of the plot.

Arguments: options are any of the following keywords:

- 'linear', 'square', and 'log' provide for plotting of the data points against the square or log of the data. 'linear' controls x-axis scale, 'square' controls the y-axis. The default is 'linear'.
- 'link' causes the data points to be connected rather than a plot of the theoretical curve.
- 'nocurve' produces a plot of data points only.
- 'oldbox' plots an additional curve on an existing plot. Only the first data set in analyze.out is plotted. It causes the program to get box and scale description from expfit.out in the current experiment.
- 'file' followed by a file name replaces analyze.out as the input.

line1, line2, ... specify curves to be plotted. The default is to plot the first six curves (if that many exist) along with the data points.

Examples: pexpl

pexpl(1,3,6)

See also: NMR Spectroscopy User Guide, User Programming

Related: expl Display exponential or polynomial curves (C)

Start of chart (P)

Start of chart in second direction (P)

WC Width of chart (P)

wc2 Width of chart in second direction (P)

#### pexpladd Add another diffusion analysis to current plot (M)

Applicability: Systems with the diffusion option.

Syntax: pexpladd(integral\_region)

Description: Adds results of another diffusion analysis to the currently plotted results.

Arguments: integral region specifies the number of the region whose results are to

be added to the existing plot.

Examples: pexpladd(1)

See also: NMR Spectroscopy User Guide

Related: expl Display exponential or polynomial curves (C)

pexpl Plot exponential or polynomial curves (C)

expladd Add another diffusion analysis to current display (M)

# pfgon Pulsed field gradient amplifiers on/off control (P)

Applicability: Systems with pulsed field gradient (PFG) modules.

Description: A global string parameter controlling the X, Y, and Z gradients for the PFG

current amplifiers. Entering su or go sets the amplifiers at the current value of pfgon. For pfgon to take effect, gradtype must equal p, q, 1, t, or u for the corresponding X, Y, or Z gradient, and a su or a go must be issued.

Values: A three-character string, with the first character controlling the X gradient, the

second the Y gradient, and the third the Z gradient. For each gradient, setting the value to y turns on an amplifier and setting the value to n turns it off. For example, pfgon='nny' turns on only the PFG amplifier on the Z channel,

and pfgon='nnn' turns off the PFG amplifiers on all channels.

See also: NMR Spectroscopy User Guide

Related: go Submit experiment to acquisition (M)

gradtype Gradients for X, Y, and Z axes (P)

setupSet up parameters for basic experiments (M)suSubmit a setup experiment to acquisition (M)

## pfww Plot FIDs in whitewash mode (C)

Syntax: pfww<(<start><,finish><,step><,'all'|'imag'>)>

Description: Plots FIDs in whitewash mode (after the first FID, each FID is blanked out in

regions in which it is behind an earlier FID). The position of the first FID is

governed by parameters wc, sc, and vpf.

Arguments: start is the index of a particular FID for arrayed 1D or 2D data sets. For

multiple FIDs, start is the index of the first FID.

finish is the index of the last FID for multiple FIDs.

 $\ensuremath{\mathfrak{step}}$  specifies the increment for the FID index. The default is 1.

'all' is a keyword to plot all of the FIDs. This is the default.

'imag' is a keyword to plot only the imaginary FID channel. The default is

'all'.

Examples: pfww

pfww(4,10,2,'imag')

See also: NMR Spectroscopy User Guide

Related: dfs Display stacked FIDs (C)

dfww Display FIDs in whitewash mode (C)

plfid Plot FIDs (C) sc Start of chart (P)

vpf Current vertical position of FID (P)

wc Width of chart (P)

#### pge Convert parameter set to PGE pulse sequence (M)

Applicability: Systems with the diffusion option.

Description: Adds all necessary parameters to perform the PGE (Pulse Gradient Experiment)

pulse sequence, taking those parameters from the file /vnmr/parlib/pge.

See also: NMR Spectroscopy User Guide

Related: pge calib Calibrate gradient strengths for PGE pulse sequence (M)

pge data Extract data from single element of PGE pulse sequence (M)

pge output Output results from PGE pulse sequence (M)

pge\_process Automated processing of data from PGE pulse sequence (M)

pge\_results Calculate diffusion constant for integral region (M)

pge\_setup Set up gradient control parameters for PGE pulse sequence (M)

## pge calib Calibrate gradient strengths for PGE pulse sequence (M)

Applicability: Systems with the diffusion option.

Description: Calibrates the parameters grad cw coef and grad p coef, which relate

the DAC values (in DAC units) to the gradient strengths (in gauss/cm). Given a diffusion constant measurement (made with pge\_results) for a known diffusion constant, pge\_calib then adjusts the calibration parameters to

produce the correct diffusion constant.

See also: NMR Spectroscopy User Guide

Related: pge Calibrate gradient strengths for PGE pulse sequence (M)

pge results Calculate diffusion constant for integral region (M)

## pge data Extract data from single element of PGE pulse sequence (M)

Applicability: Systems with the diffusion option.

Syntax: pge data(array index)

Description: Extracts integral information from a currently displayed element of a PGE

(Pulse Gradient Experiment) and writes the results in the current experiment directory as the file info\_#, where # is the value of the array\_index

argument (e.g., if array\_index is 5, the file is info\_5)

Arguments: array index is the number of the array element from which the data is

extracted.

Examples: pge data(5)

See also: NMR Spectroscopy User Guide

Related: pge Calibrate gradient strengths for PGE pulse sequence (M)

## pge\_output Output results from PGE pulse sequence (M)

Applicability: Systems with the diffusion option.

Description: Prints the calculated results from the PGE (Pulse Gradient Experiment) pulse

sequence on a printer and plots the graphs of calculated decay curves.

See also: NMR Spectroscopy User Guide

Related: pge Calibrate gradient strengths for PGE pulse sequence (M)

#### pge process Automated processing of data from PGE pulse sequence (M)

Applicability: Systems with the diffusion option.

Syntax: pge\_process

Description: Performs full automated processing of data from a PGE (Pulse Gradient

Experiment) pulse sequence.

See also: NMR Spectroscopy User Guide

Related: pge Calibrate gradient strengths for PGE pulse sequence (M)

#### pge results Calculate diffusion constant for integral region (M)

Applicability: Systems with the diffusion option.

Syntax: pge\_results(integral\_region<,reference\_region>)

Description: Calculates a diffusion coefficient based on a single integral region in the

spectrum (if one input argument) or calculates diffusion coefficient of an integral region consisting of two components (if two input arguments).

Arguments: integral\_region is the number of the integral region on which to perform

the analysis

reference\_region is the number of the integral region used to get the

value of the diffusion coefficient.

Examples: pge\_results(2)

pge results(1,3)

See also: NMR Spectroscopy User Guide

Related: pge Calibrate gradient strengths for PGE pulse sequence (M)

## pge setup Set up gradient control parameters for PGE pulse sequence (M)

Applicability: Systems with the diffusion option.

Syntax: pge setup<('no')>

Description: Prompts the user for the values of the g\_max, g\_min, g\_steps, g\_array,

nt\_first, nt\_aray, and other parameters for the PGE (Pulse Gradient Experiment) pulse sequence. These parameters are then used to calculate the

grad\_p1 and nt arrays.

Arguments: 'no' is a keyword to turn off prompting the user and instead use the current

values of the parameters to calculate the grad p1 and nt arrays.

Examples: pge setup

pge\_setup('no')

See also: NMR Spectroscopy User Guide

Related: pge Calibrate gradient strengths for PGE pulse sequence (M)

## ph Set phased mode in directly detected dimension (C)

Description: Selects the phased mode by setting the parameter dmg='ph'. In the phased

*spectra display mode*, each real point in the displayed spectrum is calculated from a linear combination of the real and imaginary points comprising each respective complex data point. The coefficients for this linear combination are

derived from the phase parameters rp and lp.

For 2D data, if <code>pmode='partial'</code> or <code>pmode=''</code> (two single quotes with no space in between), ph has an effect on the data prior to the second Fourier transform. If <code>pmode='full'</code>, ph acts in concert with the commands <code>ph1</code>,

av1, or pwr1 to yield the resultant contour display for the 2D data.

See also: NMR Spectroscopy User Guide

Related: av Set abs. value mode in directly detected dimension (C)

av1 Set abs. value mode in 1st indirectly detected dimension (C)

dmg Data display mode in directly detected dimension (P)

ft Fourier transform 1D data (C)

ftld Fourier transform along  $f_2$  dimension (C)

ft2d Fourier transform 2D data (C)

pa First-order phase in directly detected dimension (P)

Set phase angle mode in directly detected dimension (C)

pal Set phase angle mode in 1st indirectly detected dimension (C)

phl Set phased mode in 1st indirectly detected dimension (C)

ph2 Set phased mode in 2nd indirectly detected dimension (C)

pmode Processing mode for 2D data (P)

pwr Set power mode in directly detected dimension (C)
pwr1 Set power mode in 1st indirectly detected dimension (C)
rp Zero-order phase in directly detected dimension (P)

wft Weight and Fourier transform 1D data (C)
wftld Weight and Fourier transform f2 of 2D data (M)
wftld Weight and Fourier transform 2D data (M)

#### Set phased mode in 1st indirectly detected dimension (C) ph1

Description:

Selects the phased spectra display mode along the first indirectly detected dimension by setting the parameter dmq1 to the string value 'ph1'. If the parameter dmg1 does not exist, ph1 will create it and set it to 'ph1'.

In the phased mode, each real point in the displayed trace is calculated from a linear combination of the real and imaginary points comprising each respective complex data point. For hypercomplex data, the linear combination uses the real-real and imaginary-real points from each respective hypercomplex data point. The coefficients for this linear combination are derived from the phase parameters rp1 and lp1.

The ph1 command is only needed if mixed-mode display is desired. If the parameter dmg1 does not exist or is set to the null string, the display mode along the first indirectly detected dimension defaults to the display mode of the directly detected dimension (characterized by the parameter dmg). For the contour display of multidimensional data, the result of ph1 is the same as for traces provided that pmode='partial' or pmode=''.

See also: NMR Spectroscopy User Guide

Related:

рa

Set abs. value mode in 1st indirectly detected dimension (C) Data display mode in 1st indirectly detected dimension (P) dmg1 First-order phase in 1st indirectly detected dimension (P) lp1 Set phase angle mode in directly detected dimension (C) pa1 Set phase angle mode in 1st indirectly detected dimension (C)

Set phased mode in directly detected dimension (C)

Processing mode for 2D data (P) pmode

Set power mode in 1st indirectly detected dimension (C) pwr1 Zero-order phase in 1st indirectly detected dimension (P) rp1

#### ph2 Set phased mode in 2nd indirectly detected dimension (C)

Description:

Selects phased spectrum display mode processing along the second indirectly detected dimension by setting the parameter dmg2='ph2'. If dmg2 does not exist or is set to the null string, ph2 creates dmg2 and sets it to 'ph2'.

In the phased mode, each real point in the displayed trace is calculated from a linear combination of the real and imaginary points comprising each respective complex data point. For hypercomplex data, the linear combination uses the real-real and imaginary-real points from each respective hypercomplex data point. The coefficients for this linear combination are derived from the phase parameters rp2 and lp2.

The ph2 command is only needed if mixed-mode display is desired. If the parameter dmq2 does not exist or is set to the null string, the display mode along the second indirectly detected dimension defaults to the display mode of the directly detected dimension (characterized by the parameter dmg). For the contour display of multidimensional data, the result of ph2 is the same as for traces provided that pmode='partial' or pmode=''.

See also: NMR Spectroscopy User Guide

Related:

av2 Set abs. value mode in 2nd indirectly detected dimension (C) Data display mode in 2nd indirectly detected dimension (P) dmq2 ft1d Fourier transform along f<sub>2</sub> dimension (C) ft2d Fourier transform 2D data (C) First-order phase in 2nd indirectly detected dimension (P) 1p2 ph Set phased mode in directly detected dimension (C) Processing mode for 2D data (P) pmode

Set power mode in 2nd indirectly detected dimension (C) pwr2 Zero-order phase in 2nd indirectly detected dimension (P) rp2

phase Change frequency-independent phase rp (M)

Syntax: phase (phase change)

Description: Changes the phase of all peaks in the spectrum by adding a value to the current

rp value. Any excess over 360° is removed.

Arguments: phase change is the value to be added to the current rp value (i.e.,

new rp = old rp + phase change).

Examples: phase(45)

See also: NMR Spectroscopy User Guide

Related: rp Zero-order phase in directly detected dimension (P)

phase Phase selection (P)

Description: Selects the phase cycling that determines the experiment type. To create the

parameters phase, ni, and sw1 for acquisition of a 2D data set in the current

experiment, enter addpar ('2d').

Values: The following values are generally used in experiments with phase cycling. For

more details, see the specific pulse sequence.

phase=0 selects an absolute-value 2D experiment.

phase=1, 2 selects the required two components of a hypercomplex (States-

Haberkorn) experiment.

phase=3 selects TPPI (Time Proportional Phase Incrementation).

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

Cosyps Set up parameters for phase-sensitive COSY (M)

Dqcosy Set up parameters for double quantum filtered COSY (M)

Hmqc Set up parameters for HMQC pulse sequence (M)
hmqcr Set up parameters for HMQCR pulse sequence (M)
inadqt Set up parameters for INADEQUATE pulse sequence (M)
mqcosy Set up parameters for MQCOSY pulse sequence (M)
Noesy Set up parameters for NOESY pulse sequence (M)
Roesy Set up parameters for ROESY pulse sequence (M)

Tocsy
Set up parameters for TOCSY pulse sequence (M)

phase1 Phase of first pulse (P)

Applicability: Systems with a solids NMR module.

Description: Controls the first pulse phase in the cycle, in multipulse experiments.

See also: NMR Spectroscopy User Guide

Related: br24 Set up BR24 multiple pulse experiment (M)

flipflop Set up sequences for multipulse (M)

phase 2 Phase selection for 3D acquisition (P)

Description: Selects phase cycling type for 3D data acquisitions. Also selects the phase of the

second pulse in the sequence set up by flipflop. To create the parameters phase2, d3, ni2, and sw2 for acquisition of a 3D data set in the current

experiment, enter addpar('3d').

See also: NMR Spectroscopy User Guide; User Guide: Solid-State NMR

Related: addpar Add selected parameters to the current experiment (M)

d3 Incremented delay for 2nd indirectly detected dimension (P)

flipflop Set up sequences for multipulse (M)

ni2Number of increments in 2nd indirectly detected dimension (P)par3dCreate 3D acquisition, processing, display parameters (C)sw2Spectral width in 2nd indirectly detected dimension (P)

## phase3 Phase selection for 4D acquisition (P)

Description: Selects phase cycling type for 4D data acquisitions. To create the parameters

phase3, d4, ni3, and sw3 for acquisition of a 4D data set in the current

experiment, enter addpar ('4d').

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

Incremented delay for 3rd indirectly detected dimension (P)

Number of increments in 3rd indirectly detected dimension (P)

par4d Create 4D acquisition parameters (C)

Spectral width in 3rd indirectly detected dimension (P)

#### phasing Control update region during interactive phasing (P)

Description: Controls the percentage of the spectrum updated during interactive phasing

using the ds command.

Values: 10 to 100, in percent, where 100 causes the entire spectrum to be updated, and

20 causes the area between the two vertical cursors to be updated.

See also: NMR Spectroscopy User Guide

Related: ds Display a spectrum (C)

#### phfid Zero-order phasing constant for the np FID (P)

Description: Specifies the angle of zero-order rotation. This zero-order rotation is executed

as a part of retrieving the time-domain data into the active region of the memory and can be used instead of the parameter rp applied to the frequency-domain

data. phfid is used only in a complex phase rotation.

phfid (and related parameters lsfid and lsfrq) operate on complex np FID data, referred to as the  $t_2$  dimension in a 2D experiment or as the  $t_3$  dimension in a 3D experiment. phfid is in the processing group and is

properly handled through the wti display.

Values: -360.0 to +360.0, in degrees; 'n'

See also: NMR Spectroscopy User Guide

Related: dfid Display a single FID (C)

ds Display a spectrum FID (C)
ft Fourier transform 1D data (C)

ftld Fourier transform along f<sub>2</sub> dimension (C)

ft2d Fourier transform 2D data (C)

 ${\color{red} \textbf{lsfid}} \qquad \qquad \text{Number of complex points to left-shift the np FID (P)}$ 

lsfrq Frequency shift of the fn spectrum in Hz (P)

Number of data points (P)

phfid1 Zero-order phasing constant for ni interferogram (P)
phfid2 Zero-order phasing constant for ni2 interferogram (P)

Zero-order phase in directly detected dimension (P)

wft Weight and Fourier transform 1D data (C)
wftld Weight and Fourier transform f<sub>2</sub> of 2D data (M)
wftld Weight and Fourier transform 2D data (M)

wti Interactive weighting (C)

#### phfid1 Zero-order phasing constant for ni interferogram (P)

Description: Specifies the angle of zero-order rotation. This zero-order rotation is executed

as a part of retrieving the time-domain data into the active region of the memory and can be used instead of the parameter rpl applied to the frequency-domain

data. phfid1 is used in a complex phase rotation for complex  $t_1\!/t_2$ 

interferograms and in a hypercomplex phase rotation for hypercomplex  $t_1/t_2$ 

interferograms.

phfid1 (and related parameters <code>lsfid1</code> and <code>lsfrq1</code>) operate on <code>ni</code> interferogram data, both hypercomplex and complex. <code>ni</code> interferogram data are referred to as the t<sub>1</sub> dimension in both a 2D and a 3D experiment. <code>phfid1</code> is in the processing group and is properly handled through the <code>wti</code> display; that is, a <code>wti</code> operation on an <code>ni</code> interferogram applies the parameters <code>phfid1</code>, <code>lsfid1</code>, and <code>lsfrq1</code>, if selected, to the time-domain data prior to the Fourier transformation.

transformation.

Values: -360.0 to +360.0, in degrees; 'n'.

See also: NMR Spectroscopy User Guide

Related: lsfid1 Number of complex points to left-shift the ni interferogram (P)

lsfrq1 Frequency shift of the fn1 spectrum in Hz (P)

Number of increments in 1st indirectly detected dimension (P)

phfid Zero-order phasing constant for np FID (P)

phfid2 Zero-order phasing constant for ni2 interferogram (P)
rp1 Zero-order phase in 1st indirectly detected dimension (P)

wti Interactive weighting (C)

# phfid2 Zero-order phasing constant for ni2 interferogram (P)

Description: Specifies the angle of zero-order rotation. This zero-order rotation is executed

as a part of retrieving the time-domain data into the active region of the memory and can be used instead of the parameter  $\tt rp2$  applied to the frequency-domain

data. phfid2 is used in a complex phase rotation for complex  $t_1 \! / \! t_2$ 

interferograms and in a hypercomplex phase rotation for hypercomplex  $t_1 \! / \! t_2$ 

interferograms.

phfid2 (and related parameters <code>lsfid2</code> and <code>lsfrq2</code>) operate on ni2 interferogram data, both hypercomplex and complex. ni2 interferogram data are referred to as the t<sub>2</sub> dimension in a 3D experiment. phfid2 is in the processing group and is properly handled through the <code>wti</code> display.

Values: -360.0 to +360.0, in degrees; 'n'.

See also: NMR Spectroscopy User Guide

Related: lsfid2 Number of complex points to left-shift ni2 interferogram (P)

lsfrq2 Frequency shift of the fn2 spectrum in Hz (P)

Number of increments in 2nd indirectly detected dimension (P)

phfid Zero-order phasing constant for np FID (P)

phfid1 Zero-order phasing constant for ni interferogram (P)

rp2 Zero-order phase in 2nd indirectly detected dimension (P)

wti Interactive weighting (C)

# Phosphorus Set up parameters for <sup>31</sup>P experiment (M)

Description: Set up parameters for <sup>31</sup>P experiment.

## pi3ssbsq Set up pi/3 shifted sinebell-squared window function (M)

Syntax: pi3ssbsq<(<t1 inc><,t2 inc>)>

Description: Sets up a pi/3 unshifted sinebell-squared window function in 1, 2, or 3

dimensions. The macro checks whether the data is 1D, 2D, and 3D.

Arguments: t1 inc is the number of t1 increments. The default is ni.

t2 inc is the number of t2 increments. The default is ni2.

See also: NMR Spectroscopy User Guide

Related: gaussian Set up unshifted Gaussian window function (M)

niNumber of increments in 1st indirectly detected dimension (P)ni2Number of increments in 2nd indirectly detected dimension (P)pi4ssbsqSet up pi/4 shifted sinebell-squared window function (M)sqcosineSet up unshifted cosine-squared window function (M)sqsinebellSet up unshifted sinebell-squared window function (M)

#### pi4ssbsq Set up pi/4 shifted sinebell-squared window function (M)

Syntax: pi4ssbsq<(<t1\_inc><,t2\_inc>)>

Description: Sets up a pi/4 unshifted sinebell-squared window function in 1, 2, or 3

dimensions. The macro checks whether the data is 1D, 2D, and 3D.

Arguments: t1 inc is the number of t1 increments. The default is ni.

t2 inc is the number of t2 increments. The default is ni2.

See also: NMR Spectroscopy User Guide

Related: gaussian Set up unshifted Gaussian window function (M)

niNumber of increments in 1st indirectly detected dimension (P)ni2Number of increments in 2nd indirectly detected dimension (P)pi3ssbsqSet up pi/3 shifted sinebell-squared window function (M)sqcosineSet up unshifted cosine-squared window function (M)sqsinebel1Set up unshifted sinebell-squared window function (M)

# pin Pneumatics Router Interlock ((P)

Description: This parameter controls the effect of a Pneumatics Router Fault. The Pneumatic

Router can fault in four ways:

• Intake pressure < 20 psi

• Solids narrow bore stack temperature fault

• VT air flow exceeded.

• Power supply fault

When either of these fault occur, and interrupt alerts the console of the problem and this parameter determines how the fault is handled. Once a fault is registered, all subsequent acquisitions will see the error according to 'pin'. The error must be cleared and re-armed with sethw('pneufault','clear')

Values: 'n' -- the fault is ignored

'w' -- a warning msg is printed, acquisition continues 'y' -- an error msg is printed, acquisition is aborted

Related: tin Temperature interlock (P)

vtairflow VT air flow (P)
vtairlimits VT air flow limits (P)

## pintvast Plots of integral regions (M)

Applicability: Systems with VAST accessory.

Syntax: pintvast (last)

Description: pintvast plots the integrals of the partial regions of each spectra from wells

 $0 \ {\sf to} \ {\sf last}.$ 

Arguments: last is the number last sample well. The default is 96.

See also: NMR Spectroscopy User Guide

Related: intvast Builds text file the integral regions (M)

#### pir Plot integral amplitudes below spectrum (C)

Description: Plots integral amplitudes below the appropriate spectral regions.

See also: NMR Spectroscopy User Guide

Related: dpf Display peak frequencies over spectrum (C)

dpir Display integral amplitudes below spectrum (C)

dpirn Display normalized integral amplitudes below spectrum (M)
pirn Plot normalized integral amplitudes below spectrum (M)

Plot peak frequencies over spectrum (M)

#### pirn Plot normalized integral amplitudes below spectrum (M)

Description: Equivalent to the command pir except that the sum of the integrals is

normalized to the value of the parameter ins.

See also: NMR Spectroscopy User Guide

Related: dpirn Display normalized integral amplitudes below spectrum (M)

ins Integral normalization scale (P)

Plot integral amplitudes below spectrum (C)

#### piv Plot integral values below spectrum (M)

Syntax: piv<(vertical position)>

Description: Labels integrals with a bracket below the spectrum and a vertical number

indicating the integral value. See dpiv for description and use.

Related: dpir Display integral amplitudes below spectrum (C)

dpiv Display integral amplitudes below spectrum (M)

dpirn Display normalized integral amplitudes below spectrum (C)
dpivn Display normalized integral amplitudes below spectrum (M)
pirn Plot normalized integral amplitudes below spectrum (C)

Plot integral amplitudes below spectrum (C)

Plot normalized integral amplitudes below spectrum (M)

## pivn Plot normalized integral values below spectrum (M)

Syntax: pivn<(vertical position)>

Description: Labels integrals with a bracket below the spectrum and a vertical number

indicating the integral value. See dpiv for description and use.

Related: dpir Display integral amplitudes below spectrum (C)

dpiv Display integral amplitudes below spectrum (M)

dpirn Display normalized integral amplitudes below spectrum (C)
dpivn Display normalized integral amplitudes below spectrum (M)

pirn	Plot normalized integral amplitudes below spectrum (C)
pir	Plot integral amplitudes below spectrum (C)
piv	Plot integral amplitudes below spectrum (M)

# pl Plot spectra (C)

Description:

Plots one or more spectra. When a single spectrum is plotted, integral plotting is controlled by the parameter intmod as follows: intmod='off' turns off the integral plot, intmod='full' plots the entire integral, and intmod='partial' plots every other integral region.

For arrayed 1D spectra or for 2D spectra, a particular trace can be plotted by supplying the index number as an argument. For 2D data sets, spectra can be plotted from either the  $f_1$  or  $f_2$  domain by setting the parameter <code>trace</code> to <code>'f1'</code> or <code>'f2'</code>, respectively. After the command <code>ft1d</code>, interferogram can be plotted by setting <code>trace='f1'</code> and then typing <code>p1</code>. Multiple spectra can be plotted by supplying the indexes of the first and last spectra.

The position of the first spectrum is governed by the parameters wc, sc, and vp. For 1D data, subsequent spectra are positioned relative to the preceding spectrum by the vertical and horizontal offset parameters vo and ho. For 2D data, ho defines the total horizontal offset between the first and last spectrum. Also for 2D data, vo is inactive while the parameter wc2 defines the total vertical offset between the first and last spectrum.

The parameter <code>cutoff</code>, if it exists and is active, defines the distance above and below the current vertical position <code>vp</code> at which peaks are truncated. By arraying <code>cutoff</code> to have two different values, truncation limits above and below the current vertical position can be controlled. For example, <code>cutoff=50</code> truncates peaks at <code>vp+50</code> mm and <code>vp-50</code> mm. <code>cutoff=50</code>, <code>10</code> truncates peaks at <code>vp+50</code> mm and <code>vp-10</code> mm.

Arguments:

start is the index of a particular trace for arrayed 1D or 2D spectra. For multiple spectra, start is the index of the first spectrum.

finish is the index of the last spectrum for multiple spectra.

step specifies the increment for the spectral index. The default is 1.

'int' is a keyword that specifies displaying only the integral, independently of the value of intmod.

'all' is a keyword to plot all of the spectra. This value is the default.

options can be any of the following keywords:

- 'top' or 'side' cause the spectrum to be plotted either above or at the left edge of a contour plot. This assumes that the parameters sc, wc, sc2, and wc2 are those used to position the contour plot.
- 'dodc' causes all spectra to be drift corrected independently.
- 'pen1', 'pen2', 'pen3', etc. specify a pen number on a plotter.

Examples: pl

pl(1,6,2)

See also: NMR Spectroscopy User Guide

Related: cutoff Data truncation limit (P)

 $\begin{array}{ll} \textbf{dssa} & \text{Display stacked spectra automatically (C)} \\ \textbf{dsww} & \text{Display spectra in whitewash mode (C)} \\ \textbf{ftld} & \text{Fourier transform along } f_2 \, \text{dimension (C)} \\ \end{array}$ 

ho Horizontal offset (P)
intmod Integral display mode (P)

Plot spectra in whitewash mode (C)

pshr PostScript High Resolution plotting control (P)

PostScript Line Width control (P)

Start of chart (P)

Start of chart in second direction (P)

shownumx x position counting from bottom left of every spectrum (P) y position counting from bottom left of every spectrum (P)

trace Mode for 2D data display (P)

vo Vertical offset (P)

vp Vertical position of spectrum (P)

WC Width of chart (P)

wc2 Width of chart in second direction (P)

#### pl2d Plot 2D spectra in whitewash mode (C)

Syntax: pl2d<('nobase'|'fill'|'fillnb')>

Description: Plots a stacked plot of 2D spectra in whitewash mode (after the first spectra,

each spectra is blanked out in regions in which it is behind an earlier spectra). Color does not represent intensity (unlike dcon), since intensity can be seen visually, but instead successive traces are displayed in different colors so that color represents frequency. The horizontal offset parameter ho is not active for

this command.

Arguments: 'nobase' is a keyword to activate th to suppress intensity below th.

'fill' is a keyword to fill in the peaks. Note that if 'fill' (or 'fillnb') is used, th operates linearly and not logarithmically (with factors of 2) as it does

in contour or color intensity displays.

'fillnb' is a keyword to combine base suppression and peak filling.

Examples: pl2d

pl2d('nobase')

See also: NMR Spectroscopy User Guide

Related: dcon Display noninteractive color intensity map (C)

ds2d Display 2D spectra in whitewash mode (C)
dsww Display spectra in whitewash mode (C)

ho Horizontal offset (P)

plww Plot spectra in whitewash mode (C)

th Threshold (P)

#### plane Currently displayed 3D plane type (P)

Description: Stores the type of 3D plane currently displayed within VnmrJ. If plane does

not exist, it is created by the macro par3d. The command select, as well as the many macros that make use of select, requires the parameter plane to

exist for 3D data sets and to contain an appropriate value.

plane is set automatically by the macro <code>getplane</code>; it can also be set by the macro <code>ft3d</code> if automatic plane extraction is requested at the end of the 3D FT. The order of priority for the plane types is <code>'f1f3'</code>, <code>'f2f3'</code>, and then <code>'f1f2'</code>. In other words, if <code>getplane</code> is requested to extract the  $f_1f_3$  and the  $f_2f_2$  planes, plane will be set to <code>'f1f3'</code>. plane can also be set manually.

Values: 'f1f3', 'f3f1', 'f2f3', 'f3f2', 'f1f2', or 'f2f1'

See also: NMR Spectroscopy User Guide

Related: dplane Display a 3D plane (M)

dproj Display a 3D plane projection (M)

dsplanes Display a series of 3D planes (M)

Perform a 3D Fourier transform on a 3D FID data set (M,U)

getplane Extract planes from a 3D spectral set (M)

nextpl Display the next 3D plane (M)

par3d Create 3D acquisition, processing, display parameters (C)

path3d Number of complex points to left-shift np FID (P)

plplanes Plot a series of 3D planes (M)
prevpl Display the previous 3D plane (M)

select Select a spectrum or 2D plane without displaying it (C)

#### plapt Plot APT-type spectra automatically (M)

Syntax: plapt<(13Cexp\_number)>

Description: Automatically plots APT spectra. The APT spectrum is plotted on top of a

standard carbon spectrum if either an experiment with such data is specified or if a file C13 is found in curexp+'/subexp'. If neither such a subfile is found nor an experiment with standard carbon data is specified, the APT

spectrum is plotted alone.

Arguments: 13Cexp number specifies the number, from 1 to 9, of an experiment with a

standard <sup>T3</sup>C spectrum.

Examples: plapt

plapt(2)

See also: NMR Spectroscopy User Guide

Related: current experiment directory (P)

## plarray Plotting macro for arrayed 1D spectra (M)

Description: A generic macro for plotting arrayed 1D spectra. plarray is called by the

plot macro, but can also be used directly. For the plot layout, procarray distinguishes between arrays with few elements (6 or less), which will be stacked vertically (no horizontal offset), and spectra with many (greater than 6) elements. Those are stacked horizontally by default, unless there are too many lines, in which case a diagonally stacked display is chosen. Horizontal stacking is mostly adequate for pulse and power calibrations, where there are usually few lines only; diagonally stacked displays/plots are frequently chosen for  $T_1$  and  $T_2$  experiments on entire spectra, often with many lines.

The automatic stacking mode can be overridden by creating and setting a string parameter **stackmode** in the startup macro or before calling **procplot** or **procarray**. Possible values for **stackmode** are 'horizontal', 'vertical', or 'diagonal'. DEPT-type spectra can, in principle, also be processed with **procarray**, but no DEPT editing occurs, of course.

See also: NMR Spectroscopy User Guide

Related: aexppl Automatic expansion plot (M)

plc Plot carbon spectrum (M)
plh Plot proton spectrum (M)
plot Automatically plot spectra (M)
procarray Process arrayed 1D spectra (M)

stackmode Stack control for processing arrayed 1D spectra (P)

## plate\_glue Define a glue order for plotting and display (U)

Applicability: Systems with VAST accessory

Description: In a Unix terminal or shell window type plate\_glue. The glue order is

determined by clicking on the wells to be displayed. Save the glue order file in

the user's vnmrsys/templates/glue directory.

See also: NMR Spectroscopy User Guide

Related: dsvast2d Display VAST data in a pseudo-2D format (M)

plvast Plot VAST data in a stacked 1D-NMR matrix (M)
plvast2d Plot VAST data in a pseudo-2D format (M)

#### plc Plot a carbon spectrum (M)

Syntax: plc<(pltmod)>

Description: Plots a carbon spectrum based on the parameters pltmod (the options 'off',

'full', and 'fixed' are implemented) and intmod('off', 'full', and 'partial' are implemented). Peak frequency labels, in ppm, are usually

plotted.

Arguments: pltmod is an alternate value of pltmod for this macro only. The value of the

pltmod parameter is not changed.

Examples: plc

plc('full')

See also: NMR Spectroscopy User Guide

Related: intmod Integral display mode (P)

pltmod Plotter display mode (P)

## plcosy Plot COSY- and NOESY-type spectra automatically (M)

Syntax: plcosy(<'pos'|'neg'><,><levels<,spacing<,exp1D>>>)

Description: Automatically plots 2D COSY- and NOESY-type spectra (homonuclear

correlated spectra). Features include the following:

- Keeps the orientation (f<sub>1</sub>, f<sub>2</sub>) of the spectrum on the screen.
- Plot area is optimized.
- Number of contour levels and their spacing can be selected.
- Negative or positive contours can be suppressed.
- 1D traces can be plotted along both axes; such 1D traces are taken from a full (or reduced) 1D spectrum in an other experiment, or from a subfile from within the current experiment.
- Works correctly for expansions.
- 1D traces can be suppressed, allowing a larger area for the 2D spectrum.
- 1D spectrum can be in any experiment.
- With phase-sensitive spectra using a plotter with one pen or a printer such as a LaserJet, if 'pos' or 'neg' are not selected, seven positive levels (or the specified number of positive contours) and one negative level are plotted, to distinguish positive and negative signals.

In multiexperiment mode, for the first plot, the experiment with the 1D spectrum should be specified (at least if it is not in exp1). From then on, the 1D spectrum will be stored *within* the experiment with the 2D spectrum, which allows much faster switching between spectra and also frees the other (1D) experiment for other tasks. Because of this internal storage, the exp1D argument is not required for subsequent plots.

Arguments: 'pos' is a keyword to plot only positive contours.

'neg' is a keyword to plot only negative contours.

levels is the number of contour levels. The default is 7.

spacing is the spacing between the contours. The default is 2.

exp1D is the experiment in which the proton 1D spectrum resides. This can be a full 1D spectrum, but the referencing must be the same as for the 2D. A negative number suppresses the proton trace. The default is from a subfile.

Examples: plcosy

plcosy(12,1.5)
plcosy('pos',7,2,3)
plcosy(7,2,-1)
plcosy('neg')

See also: NMR Spectroscopy User Guide

#### pldept Plot DEPT data, edited or unedited (M)

Description: Plots out DEPT data, either edited or not edited.

See also: NMR Spectroscopy User Guide

Related: adept Automatic DEPT analysis and spectrum editing (C)

autodept Automated complete analysis of DEPT data (M)

deptproc Process DEPT data (M)

padept Perform adept analysis and plot resulting spectra (C)

## plfid Plot FIDs (C)

Syntax: plfid<(<start><,finish><,step><,'all'|'imag'>

<,pen>)>

Description: Plots one or more FIDs. The position of the first FID is governed by the

parameters wc, sc, and vpf. A subsequent FID is positioned relative to the preceding FID by the vertical and horizontal offset parameters vo and ho.

Arguments: start is the index of a particular FID for arrayed 1D or 2D data sets. For

multiple FIDs, start is the index of the first FID.

finish is the index of the last FID for multiple FIDs. To include all FIDs, set start to 1 and finish to the parameter arraydim (see example).

step specifies the increment for the FID index. The default is 1.

'all' is a keyword to plot all of the FIDs. This is the default.

'imag' is a keyword to plot the imaginary FID channel only. The default is

'all'.

pen is a keyword with the plotter pen number: 'pen1', 'pen2', 'pen3',

etc. The default is 'pen1'.

Examples: plfid(1,arraydim,3)

See also: NMR Spectroscopy User Guide

Related: arraydim Dimension of experiment (P)

dfs Display stacked FIDs (C)

dfww Display FIDs in whitewash mode (C)

ho Horizontal offset (P)
sc Start of chart (P)
vo Vertical offset (P)

vpf Current vertical position of FID (P)

wc Width of chart (P)

plfit Plot deconvolution analysis (M)

Description: Produces a complete output plot of a deconvolution analysis, plotting the

observed spectrum, the full calculated spectrum, each individual component, as

well as the numerical results of the analysis.

See also: NMR Spectroscopy User Guide

Related: fitspec Perform spectrum deconvolution (C)

showfitDisplay numerical results of deconvolution (M)usemarkUse "mark" output as deconvolution starting point (M)

plgrid Plot a grid on a 2D plot (M)

Syntax: (1) plgrid<(<spacing><,><pen>) >

(2) plgrid<(start\_f2,incr\_f2,start\_f1,incr\_f1<,pen>)>

Description: Plots grid lines over a 2D plot.

Arguments: spacing specifies the approximate spacing of the grid lines, in cm. The

default is intervals of approximately 1 cm, rounded so that the intervals fall at a

multiple of 1, 2, or 5 (in Hz) or 1p, 2p, or 5p (in ppm).

pen is a keyword with the plotter pen number: 'pen1', 'pen2', 'pen3',

etc. The default is 'pen1'.

start\_f2, incr\_f2, start\_f1, incr\_f1 define the starting and increment frequencies in both f<sub>2</sub> and f<sub>1</sub> for a grid. Add the p suffix to a value to

enter it in ppm (see last example below).

Examples: plgrid

plgrid(2)
plgrid('pen5')
plgrid(1.5, 'pen2')
plgrid(1p, 0.5p, 3p, 0.5p)

See also: NMR Spectroscopy User Guide

Related: grid Draw a grid on a 2D display (C)

plh Plot proton spectrum (M)

Syntax: plh<(pltmod)>

Description: Plots a proton spectrum based on the parameters pltmod (the options 'off',

'fixed', 'full', and 'variable' are implemented) and intmod

('off', 'full', and 'partial' are implemented).

Arguments: pltmod is an alternate value of the parameter pltmod for this macro only.

The value of the pltmod parameter is not changed.

Examples: plh

plh('full')

See also: NMR Spectroscopy User Guide

Related: intmod Integral display mode (P)

pltmod Plotter display mode (P)

sp Start of plot (P)
wp Width of plot (P)

plhet2dj Plot heteronuclear J-resolved 2D spectra automatically (M)

Syntax: plhet2dj<('pos'|'neg'<,levels<,spacing<,exp1D>>>)>

Description: Automatically plots 2D spectra of type HET2DJ (heteronuclear J-resolved 2D spectra) with the following features:

- Displayed portion of the spectrum is plotted in f2-mode
- Plot area is optimized
- Number of contour levels and their spacing can be selected
- Negative or positive contours can be suppressed
- A 1D trace can be plotted along the f<sub>2</sub> axis; such a 1D trace is taken from a full (or reduced) 1D spectrum in an other experiment, or from a file from within the current experiment.
- Expansions are handled correctly
- The 1D trace can be suppressed, which allows using a larger area for the 2D spectrum
- The 1D spectrum can be in any experiment
- With phase-sensitive spectra, if 'pos' or 'neg' are not selected and the plotter has only one pen (also for printers like the LaserJet), the specified number of positive contours are plotted (default is 7), but only one negative level, to distinguish positive and negative signals.

In multiexperiment mode, for the first plot the experiment with the 1D spectrum should be specified (at least if it is not in exp1). From then on, the 1D spectrum is stored *within* the experiment with the 2D spectrum, which allows much faster switching between the spectra and also frees the other 1D experiment for other tasks. Because of this internal storage, the exp1D argument is not required for subsequent plots.

Arguments:

'pos' is a keyword to only plot positive contours

'neq' is a keyword to only plot negative contours

levels is the number of contour levels. The default is 7.

spacing is the spacing between the contours. The default is 2.

exp1D is the number from 1 to 9 of the experiment in which the 1D spectrum resides. This can be a full 1D spectrum, but the referencing must be the same as for the 2D. A negative number will suppress the 1D trace. The default is 1 (for exp1).

expr).

Examples: plhet2dj

plhet2dj(12,1.5)
plhet2dj('pos',7,2,3)
plhet2dj(7,2,-1)

See also: NMR Spectroscopy User Guide

#### plhom2dj Plot homonuclear J-resolved 2D spectra automatically (M)

Syntax: (1) plhom2dj < (levels < , spacing < , exp1D >>) >

(2) plhom2dj('pos'|'neg'<,levels<,spacing<,exp1D>>>)

Description:

Automatically plots 2D spectra of type HOM2DJ (homonuclear J-resolved 2D spectra). Features include the following:

- The displayed portion of the spectrum is plotted in f2-mode
- The plot area is optimized
- Number of contour levels and their spacing can be selected
- Negative or positive contours can be suppressed

- A 1D trace can be plotted along the f<sub>2</sub> axis; such a 1D trace is taken from a full (or reduced) 1D spectrum in an other experiment, or from a file from within the current experiment.
- It also works correctly for expansions
- The 1D trace can be suppressed, which allows using a larger area for the 2D spectrum
- The 1D spectrum can be in any experiment
- With phase-sensitive spectra, if 'pos' or 'neg' are not selected and the plotter has only 1 pen (also for printers like the LaserJet) 7 or the specified number of positive contours are plotted, but only one negative level, to distinguish positive and negative signals.

In multiexperiment mode, for the first plot the experiment with the 1D spectrum should be specified (at least if it is not in exp1). From then on, the 1D spectrum will be stored within the experiment with the 2D spectrum, which allows much faster switching between the spectra and also frees the other (1D) experiment for other tasks. Because of this internal storage, the exp1D argument is not required for subsequent plots.

Arguments: levels is the number of contour levels. The default is 7.

spacing is the spacing between the contours. The default is 2.

exp1D is a number from 1 to 9 for the experiment in which the 1D spectrum resides. The spectrum can be a full 1D spectrum but the referencing must be the same as for the 2D. A negative number will suppress the 1D trace. The default is 1 (for exp1).

'pos' specifies only plot positive contours.

'neg' specifies only plot negative contours.

Examples: plhom2dj

plhom2dj(25,1.2) plhom2dj('pos',7,2,3) plhom2dj(7,2,-1)

See also: NMR Spectroscopy User Guide

## plhxcor

## Plot X,H-correlation 2D spectrum (M)

Syntax: plhxcor(<'pos'|'neg'><,><levels<,spacing <,exp1D H<,exp1D X>>>>)

Description: Automatically plots 2D spectra of type HETCOR, COLOC, HMQC, HMBC (direct and indirect detection). Features include the following:

- Keeps the orientation  $(f_1, f_2)$  of the spectrum on the screen.
- Plot area is optimized.
- Number of contour levels and their spacing can be selected.
- Negative or positive contours can be suppressed.
- 1D proton and X traces can be plotted along both axes; such 1D traces are taken from full (or reduced) 1D spectra in other experiments or subfile within the current experiment.
- Works correctly for expansions.
- 1D traces can be suppressed, allowing a larger area for the 2D spectrum.
- 1D spectra can be in any experiment.

Arguments: 'pos' is a keyword to plot only positive contours.

'neg' is a keyword to plot only negative contours.

levels is the number of contour levels. The default is 7.

spacing is the spacing between the contours. The default is 2.

exp1D\_H is a number from 1 to 9 of the experiment in which the proton 1D spectrum resides; this can be a full 1D spectrum, but the referencing must be the same as for the 2D. A negative number will suppress the proton trace. The default is a subfile in the current experiment.

exp1D\_X is a number from 1 to 9 of the experiment in which the X 1D spectrum resides. A negative number suppresses the X trace. the default is a subfile in the current experiment.

Examples: plhxcor(12,1.5)

plhxcor(7,2,3) plhxcor(7,2,1,3) plhxcor('pos',7,2,-1,3) plhxcor(7,2,-1,-1) plhxcor('neq')

See also: NMR Spectroscopy User Guide

Related: hetcor Set up parameters for HETCOR pulse sequence (M)

## pll Plot a line list (M)

Syntax:  $pll < (x, y, minimum_y) >$ 

Description: Produces a columnar line list on a plotter, similar to what would appear on a

printer. pll is quite different from the alternative method of plotting peak frequencies using ppf. The output of pll is automatically formatted into

multiple columns, depending on the number of lines.

Arguments: x is the x position of the upper left of the line list.

y is the y position of the upper left of the line list.

minimum y is the minimum y at which to reset back to top.

Examples: pll

pll(20,150)

pll(5,wc2max\*.8,wc2max\*.5)

See also: NMR Spectroscopy User Guide

Related: ppf Plot peak frequencies over spectrum (M)

#### pll2d Plot results of 2D peak picking (C)

Syntax: pll2d<(options)>

Description: Plots the results of applying the 112d command to pick 2D peaks in a 2D

spectrum or a 2D plane of a 3D spectrum. Refer to the description of 112d for

a description of the process and the options available.

See also: NMR Spectroscopy User Guide

Related: 112d Automatic and interactive 2D peak picking (C)

### plockport Port number to use to lock out multiple ProTune processes (P)

Syntax: plockport=<value>

Description: The parameter must be created as a real local parameter before it can be used.

The parameter is used to override a default port number that is used internally in ProTune to prevent two Java ProTune process from running simultaneously.

Related: protune Macro to start ProTune (M)

Create new parameter in a parameter tree (C)

## plot Automatically plot spectra (M)

Description: A unive

A universal plotting macro normally called through the procplot macro (which by itself serves as processing and plotting facility for automatic experiments). plot can also be used directly by the user who then doesn't have to remember specific plotting macros. Of course, the specialized macros can still be called directly if the user know their names.

The main purpose of plot is to automatically call the correct specialized plotting macro, depending on the user definition or otherwise on the type of data in the experiment. A plotting macro is selected automatically as follows:

APT spectra: plapt
other, non-arrayed 1D data: plot1d
DEPT type arrayed spectra: pldept
other arrayed 1D spectra: plarray
J-resolved 2D spectra: pl2dj
homonuclear correlation 2D spectra: plcosy
heteronuclear correlation 2D spectra: plhxcor

Other types of 2D spectra (mostly multiple-quantum 2D spectra such as 2D-INADEQUATE) are not plotted automatically at this time. For phase-sensitive 2D spectra, automatic plotting is only provided if they were acquired using the method described by States, Haberkorn, and others; TPPI spectra are not covered.

Note that plot macros in general should not adjust the phase, the vertical scale, or change the integral size and reset points; these are assumed to be adjusted either by hand or by a suitable processing macro like procplot and the macros called therein. The plotting macros only make adjustments in order to make spectrum and parameters fit onto the page the desired way.

See also: NMR Spectroscopy User Guide

Related: apptype Application type (P)

execpars
Set up the exec parameters (M)
execplot
Execute plotting macro (P)
plapt
Plot APT spectra (M)
plarray
Plot arrays (M)

plcosy Plot homonuclear 2D correlation spectra (M)

pldept Plot DEPT type spectra (M)

plhxcor Plot heteronuclear correlation spectra (M)

plot1d Plot 1D spectra (M)
plt2Darg Plot 2D arguments (P)

procplot Automatically process FIDs (M)

#### plot1d Plotting macro for simple (non-arrayed) 1D spectra (M)

Description:

A generic macro for plotting non-arrayed 1D spectra using a set of standard macros. plot1d is called by the plot macro, but can also be used directly. plot1d first tries to find a specific macro (e.g., plh, plc, plp) for the current observe nucleus. If such a macro exists, it is called. If a nucleus-specific macro is not found in the command path, a "minimal" 1D plot is produced.

See also: NMR Spectroscopy User Guide

Related: plc Plot carbon spectrum (M)

plhPlot proton spectrum (M)plpPlot phosphorus spectrum (M)plotAutomatically plot spectra (M)

#### plot2D Plot 2D spectra (M)

Description: Checks for the presence of appropriate proton or carbon high-resolution spectra

in the directory userdir+'/data/'+sample and decides to plot high resolution spectra or a projection depending on whether or not the proton or

carbon spectrum exists.

Arguments: The plot2D macro accepts the following arguments:

'pos' keyword to plot positive contours 'neg' keyword to plot negative contours.

'both' keyword to plot both positive and negative contours.

levels number of levels to be plotted.
spacing spacing between contour levels.

'top' keyword to plot a high-resolution spectrum on the top.
'notop' keyword to plot a non-high-resolution spectrum or projection.

'proj' keyword to plot a projection on top.

'side' keyword to plot a high-resolution spectrum on the side.
'noside' keyword to plot a non-high-resolution spectrum or projection.

'proj' keyword that plots a projection on the side.

Examples: plot2D('pos',2,5,'top','side')

See also: NMR Spectroscopy User Guide

Related: plot Automatically plot spectra (M)

plotside Plot spectrum on side (M)
plottop Plot spectrum on top (M)
plottopside Plot spectrum on top and side (M)

#### plotfile Plot to a file (M)

Syntax: plotfile('argument')

Description: plots automatically to a file. Supported output formats are: ps, pdf, jpq, pcl,

hpgl. and png.

Arguments: auto — plots automatically.

manual — plots contents of printer queue to a file.

Path and file name — plots to specified file in the directory specified. Plots to the data directory using the supplied name if no path is specified.

Examples: plotfile('xxx.fid/myplotfile.PDF') plots will go into saved

data directory.

plotfile('myplotfile.PDF') - plots will go to vnmrsys/plots if

FID has not been saved.

plothiresprep High resolution plot output preparation (M)

Description: Required for the operation of the "Plot HiRes..." popup window to interactively

use plottop/plotside of spectra in work spaces EXPn - creates necessary

variables.

plotmanual Plot manually (M)

Description: Makes correct choice of printer (for preview) and correct alignment with respect

to parameter output, resets back screen to original size & position based on

selections made on the Plot page.

plotlogo Plots a logo (M)

Description: Plots a logo Varian logo using image file located in /vnmr/iconlib/

varianlogo.gif or a custom logo from location specified in the parameter

plotlogo.

Reads value for doplotlogo (n/y), plotlogox (x dimension image), and

plotlogoy (y dimensions image), and image file in iconlib.

plotpreview Creates temporary plots of the current plot output (M)

Syntax: plotpreview<('argument')>

Description: Creates preview of the output from auto-plotting the current spectrum and starts

an Acrobat PDF reader. The preview output can be saved in PS, PDF, PCL,

HPGL, JPG or PNG formats.

Arguments: no argument — creates preview of whatever is ready to send to the plotter.

auto — creates preview of auto-plot based upon plot macro manual — creates preview of the contents of the print queue.

plotside Plot spectrum on side (M)

Description: Plots projection or high-resolution spectrum on the side of a 2D spectrum.

plotside is used with plot2D and is not useful by itself.

See also: NMR Spectroscopy User Guide

Related: plot2D Plot 2D spectra (M)

plotter Plotter device (P)

Description: Sets the plotter in use on the system.

Values: A string with entries such as 'DraftPro', 'ThinkJet 96',

'LaserJet 300', 'jim', 'varian1', and 'Laser1'.

See also: NMR Spectroscopy User Guide

Related: setplotdev Return characteristics of a named plotter (C)

showplotter Show list of currently defined plotters and printers (M)

plottop Plot spectrum on top (M)

Description: Plots projection or high resolution spectra on the top of a 2D spectrum.

plottop is used with plot2D and is not useful by itself.

See also: NMR Spectroscopy User Guide

Related: plot2D Plot 2D spectra (M)

## plottopside Plot spectrum on top and side (M)

Description: Plots projection or high-resolution spectrum on the top and side of a

2D spectrum. plottopside is used with plot2D and is not useful by itself.

See also: NMR Spectroscopy User Guide

Related: plot2D Plot 2D spectra (M)

## plp Plot phosphorus spectrum (M)

Syntax: plp<(pltmod)>

Description: Plots a phosphorus spectrum based on the parameters pltmod (the options

'off', 'full', and 'fixed' are implemented) and intmod ('off', 'full', and 'partial' are implemented). Peak frequency labels, in ppm,

are usually plotted.

Arguments: pltmod is an alternate value of pltmod for this macro only. The value of the

pltmod parameter is not changed.

Examples: plp

plp('full')

See also: NMR Spectroscopy User Guide

Related: intmod Integral display mode (P)

plh Plot proton spectrum (M)
pltmod Plotter display mode (P)

## plplanes Plot a series of 3D planes (M)

Syntax: plplanes(start\_plot,stop\_plot<,'pos'|'neg'>

<,number levels><,spacing>)

Description: Creates the 2D contour plots for a subset of the 3D planes specified by the

parameter plane.

Arguments: start\_plot specifies the number, greater than 0, of the 3D plane with which

plotting is to begin.

stop\_plot specifies the number of the 3D plane with which plotting is to end. If start\_plot is greater than stop\_plot, only the first plane, whose number is start\_plot, is plotted. The range of stop\_plot depends on the value of the parameter plane:

- if plane='f1f3', stop plot is between 0 and fn2/2
- if plane='f2f3', stop plot is between 0 and fn1/2
- if plane='f1f2', stop\_plot is between 0 and fn/2

'pos' is a keyword specifying that phase-sensitive spectra plot positive peaks only. The default is to plot both positive and negative peaks.

'neg' is a keyword specifying that phase-sensitive spectra plot negative peaks only. The default is to plot both positive and negative peaks.

levels is maximum number of contour levels to plot. The default is 4.

spacing is relative intensity of successive contour levels. The default is 2.

Note that the optional arguments 'pos' | 'neg', number\_levels, and spacing are for the VnmrJ plotting command pcon.

Examples: plplanes(1,3)

plplanes(2,3,'pos',4)

Related: dplane Display a 3D plane (M)

dproj Display a 3D plane projection (M)
dsplanes Display a series of 3D planes (M)

getplane Extract planes from 3D spectral data set (M)

nextpl Display the next 3D plane (M)

path3d Path to currently displayed 2D planes from a 3D data set (P)

Plot contours on a plotter (C)

plane Currently displayed 3D plane type (P)
prevpl Display the previous 3D plane (M)

# plt2Darg Plot 2D arguments (P)

Applicability: Liquids

Description: Specifies options for contours and 1D projections on 2D plots, used by the

plot2D macro. The plot options are selected on the Defaults page in the Acquire

folder for most 2D sequences.

Related: plot 2D spectra (M)

### pltext Plot text file (M)

Description: Plots a text file.

Arguments: file is the name of a text file. The default is the current experiment text file.

x and y are coordinates, in mm, of the first line of text. This positions the location of the output. The default is the upper left-hand corner of the page. width is the maximum column text width, in characters. pltext uses a word

wrap to make the text fit into the width specified.

\$x\_next and \$y\_next are the coordinates where the start of the next line would have been plotting. This is useful for subsequent character plotting.

\$y\_increment is the vertical increment between lines.

Examples: pltext

pltext(wcmax-70)

pltext(userdir+'/exp3/text')

pltext(100,100)

pltext(userdir+'/exp4/text',200,200,24)

pltext: \$x, \$y, \$dy

See also: NMR Spectroscopy User Guide

Related: dtext Display a text file in the graphics window (C

ptext
Print out a text file (M)

Display text or set new text for current experiment (C)

userdir User directory (P)

### pltmod Plotter display mode (P)

Description: Controls plotting of a proton, carbon, or phosphorus spectrum.

Values: 'off' sets no plotting.

'fixed' takes sp and wp as is.

'full' adjusts sp and wp to plot the full spectrum.

'variable' adjusts sp and wp to plot only the region of interest.

See also: NMR Spectroscopy User Guide

Related: plc Plot carbon spectrum (M)

plh Plot proton spectrum (M)
plp Plot phosphorus spectrum (M)

sp Start plot (P)
wp Width of plot (P)

### plvast Plot VAST data in a stacked 1D-NMR matrix format (M)

Applicability: Systems with the VAST accessory.

Syntax: plvast<(display order, number of columns plotted)>
Description: plvast arranges and plots the traces from a reconstructed 2D data set (see

vastglue) as an array of 1D spectra in a convenient format (as a matrix of 1D spectra). If no arguments are provided, the number of rows and columns are

spectra). If no arguments are provided, the number of rows and columns are determined by the periodicity of the display order. For example, if a block of 96 spectra, as is typical for a microtiter-plate, have been acquired using VAST automation, the spectra is plotted in a matrix 8 rows and 12 columns.

The default is to plot the spectra from 1 through arraydim (the number of spectra in the 2D data set). An optional argument (plvast(##)) allows one to specify that only spectra from 1 through ## should be plotted.

Arguments: display order is optional and its default value is the glue order as listed in

glueorderarray.

number of columns plotted. The default value of is deduced by examining the periodicity of the requested display order. The number of columns plotted can entered as the second argument or as the first

argument if the default display order is used.

Examples: plvast

plvast(12)

plvast('glue file', 4)

See also: NMR Spectroscopy User Guide

Related: dsast2d Display VAST data in a pseudo-2D format (M)

dsvast Display VAST data in a stacked 1D-NMR matrix (M)

plvast2d Plot VAST data in a pseudo-2D format (M)

plate\_glue define a display order (U)

# plvast2d Plot VAST data in a stacked pseudo-2D format (M)

Applicability: Systems with the VAST accessory.

Syntax: plvast2d<(number) >

Description: If an array of 1D spectra have been acquired (in particular if a block of 96

spectra has been acquired using VAST automation, especially in a microtiter-plate format) and if these spectra have been glued into a reconstructed 2D dataset (see vastglue), plvast2d will arrange and plot them (on the plotter) in a convenient pseudo-2D format (almost like an LC-NMR chromatogram). Well labels are not attached to the spectra and spectra are

plotted with 12 spectra per row.

Arguments: number specifies that only spectra from 1 through number should be plotted.

The default is to plot all the spectra (from 1 through arraydim).

Related: dsast2d Display VAST data in a pseudo-2D format (M)

dsvast Display VAST data in a stacked 1D-NMR matrix (M)

plvast Plot VAST data in a stacked 1D-NMR matrix (M)

### plww Plot spectra in whitewash mode (C)

Syntax: plww<(start,finish,step><,'all'>)>

Description: Plots one or more spectra in whitewash mode (after the first spectra, each

spectra is blanked out in regions in which it is behind an earlier spectra).

Arguments: start — index of the first spectra when plotting multiple spectra. It is also the

index number of a particular trace to be plotted when plotting arrayed 1D

spectra or 2D spectra. The default is to plot all spectra.

finish — index of the last spectra when plotting multiple spectra. step — increment for the spectral index when plotting multiple spectra,

default is 1.

'all' — (default) keyword to plot all spectra in the array.

See also: NMR Spectroscopy User Guide

Related: dss Display stacked spectra (C)

dsww Display spectra in whitewash mode (C)

pl Plot spectra (C)

### pmode Processing mode for 2D data (P)

Description: Specifies the type of 2D spectral data that the 2D Fourier transform (FT) will

yield. pmode is in the processing group.

Values: ' ' (null string, shown by two single quotes with no space in between) specifies a processing mode in which it is not possible to change either the  $f_2$  or  $f_1$  display

mode after the 2D FT. If the  $f_2$  display mode has been set to phased (dmg='ph'), each  $f_2$  spectrum is phase rotated using the phase constants rp and lp prior to the FT along the second dimension. If the  $f_2$  display mode has been set to power (dmg='pwr') or absolute-value (dmg='av'), however, the  $f_2$  spectrum is not processed any further after the first FT. The complex  $t_1$  interferograms are handled in a similar manner. If the  $f_1$  display mode has been set to phased (dmg1='ph1'), each  $f_1$  spectrum is phased using the phase constants rp1 and lp1. If the display mode has been set to power (dmg1='pwr1') or to absolute value (dmg1='av1'), the appropriate

(dmg1='pwr1') or to absolute value (dmg1='av1'), the appropriate magnitude calculation is performed, with the result being placed in the real part of the appropriate complex datum and a 0 being placed in the imaginary part. At the end of the 2D transform, the spectral data file datdir/data is reduced from complex data to real data ("VnmrJ REDUCE" display message).

'partial' specifies a processing mode in which it is not possible to change the  $f_2$  display mode after the 2D FT. It is possible, however, to select between the three  $f_1$  display modes without having to reprocess the 2D data. If the  $f_2$  display mode has been set to phased (dmg='ph'), each  $f_2$  spectrum is phase rotated using the phase constants rp and lp prior to FT along the second dimension. If the  $f_2$  display mode is set to power (dmg='pwr') or absolute value (dmg='av'), the  $f_2$  spectrum is not processed any further after the first FT. Regardless of the requested  $f_1$  display mode, no further processing is performed by ft2d on the  $f_1$  spectra after the second FT. The calculations on 2D spectral data necessary to achieve the requested  $f_1$  display mode are performed by dcon or dconi. If pmode does not exist, it is assigned a value of 'partial' internal to VnmrJ.

'full' specifies a processing mode in which it is possible to select between the three display modes for each dimension without having to reprocess the 2D data. Regardless of any requested display mode, no display mode processing is performed by ft2d on the f<sub>2</sub> spectra after the first or second FT.

The hypercomplex data structure for the 2D time domain data is:

```
{Re(t1)Re(t2), Re(t1)Im(t2), Im(t1)Re(t2), Im(t1)Im(t2)}
```

and is experimentally composed by the pulse sequence generation arraying mechanism. The hypercomplex data structure for the  $t_1$  interferograms is:

```
{Re(t1)Re(F2), Re(t1)Im(F2), Im(t1)Re(F2), Im(t1)Im(F2)}
```

where Re represents the real part and Im represents the imaginary part. A hypercomplex FT along  $t_1$  yields a hypercomplex 2D spectrum with the following data structure per hypercomplex point:

```
{Re(F1)Re(F2), Re(F1)Im(F2), Im(F1)Re(F2), Im(F1)Im(F2)}
```

Note that if pmode='full', the ft2d program will require an array index or coefficients for the construction of the  $t_1$  interferograms.

See also: NMR Spectroscopy User Guide

Related: av Set abs. value mode in directly detected dimension (C)

av1 Set abs. value mode in 1st indirectly detected dimension (C)

dcon Display noninteractive color intensity map (C)

dconi Interactive 2D data display (C)

dmg Data display mode in directly detected dimension (P)
dmg1 Data display mode in 1st indirectly detected dimension (P)

ftld Fourier transform along f<sub>2</sub> dimension (C)

ft2d Fourier transform 2D data (C)

ph Set phased mode in directly detected dimension (C)
ph1 Set phased mode in indirectly detected dimension (C)
pwr Set power mode in directly detected dimension (C)
pwr1 Set power mode in 1st indirectly detected dimension (C)

wft1d Weight and Fourier transform 2D data (C)
wft2d Weight and Fourier transform 2D data (C)

### poly0 Display mean of the data in regression.inp file (M)

Description: Calculates and displays the mean of data in the file regression.inp.

See also: User Programming

 $Related: \quad \ \ \text{average and standard deviation of input (C)}$ 

expl Display exponential or polynomial curves (C)

# pp Decoupler pulse length (P)

Description: Sets the decoupler pulse length for use by pulse sequences such as DEPT,

HET2DJ, and HETCOR.

See also: NMR Spectroscopy User Guide

Related: AC1-AC9 Automatic calibration (M)

Dept Set up parameters for DEPT experiment

dhpDecoupler high-power control with class C amplifier (P)dpwrPower level for first decoupler with linear amplifier (P)hetcorSet up parameters for HETCOR pulse sequence (M)

p1 First pulse width (P)
pw Pulse width (P)

# ppa Plot a parameter list in plain English (M)

Syntax: ppa<(x<,y>)>

Description: Plots parameters in plain English (instead of in a table with parameter names

and their values as plotted by the parameter pap).

Arguments: x controls the x offset, in mm, from the lower left of the plot to the starting

position (upper left) of the parameter list. The default is a preset position on the

page (upper left corner).

y controls the y offset, in mm, from the lower left of the plot to the starting position (upper left) of the parameter list. Default is a preset position on the page

(upper left corner).

Examples: ppa

ppa(10)

ppa(wcmax-80,wc2max\*.9)

See also: NMR Spectroscopy User Guide

Related: bpa Plot boxed parameters (M)

hpa Plot parameters on special preprinted chart paper (C)

pap Plot out "all" parameters (C)

pltext Plot a text file (M)

### ppcal Proton decoupler pulse calibration (M)

Description: Proton decoupler pulse calibration for DEPT, HETCOR, INEPT, etc.

See also: NMR Spectroscopy User Guide

Related: AC1S-AC11S Automatic calibration (M)

d2pulSet up parameters for D2PUL pulse sequence (M)DeptSet up parameters for DEPT experimenthetcorSet up parameters for HETCOR pulse sequence (M)ineptSet up parameters for INEPT pulse sequence (M)

# ppf Plot peak frequencies over spectrum (C)

Description: Plots peak frequencies, in units specified by the axis parameter, in the plotter

device. Only those peaks greater than th high are selected. Two basic modes of label positioning are available: labels placed at the top, with long "leaders" extending down to the tops of the lines (syntax 1 using the 'top' keyword), or labels positioned just above each peak, with short leaders (syntax 2 using the

'leader' keyword). The default is short leaders.

Arguments: 'noll' is a keyword to plot frequencies using the last previous line listing.

'pos' is a keyword to plot positive peaks only ('noneg' is the same as

'pos').

noise\_mult is a numerical value that determines the number of noise peaks plotted for broad, noisy peaks. The default is 3. A smaller value results in more peaks, a larger value results in fewer peaks, and a value of 0.0 results in a line listing containing all peaks above the threshold th. Negative values of

noise\_mult default to 3. The noise\_mult argument is inactive when the 'noll' keyword is specified.

'top' is a keyword to plot labels at the top with long leaders. In this mode, the height of labels is varied by changing the parameter wc2.

'leader' is a keyword to plot labels positioned just above each peak with short leaders.

length specifies the leader length, in mm, if labels are positioned just above each peak. The default length is 20 mm.

Examples: ppf('pos')

ppf('leader',30)
ppf('top','noll')

ppf('pos', 0.0, 'leader', 30)

See also: NMR Spectroscopy User Guide

Related: axis Axis label for displays and plots (P)

dpf Display peak frequencies over spectrum (C)
dpir Display integral amplitudes below spectrum (C)

dpirn Display normalized integral amplitudes below spectrum (M)

Plot integral amplitudes below spectrum (C)

Plot normalized integral amplitudes below spectrum (M)

th Threshold (P)

# pph Print pulse header (M)

Syntax: pph(file)

Description: Prints out the shape file header (i.e., all lines starting with #).

Arguments: file is the name of the shape file, including the extension.

Examples: pph('shgrad.GRD')

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

### ppmm Resolution on printers and plotters (P)

Description: An internal software parameter, selected automatically based on the plotter

configuration, that contains the resolution in dots/mm on raster graphics printers. On pen plotters, ppmm contains the resolution of points drawn. On

PostScript printers, ppmm adjusts linewidths.

## pprofile Plot pulse excitation profile (M)

Syntax: pprofile<(axisflag<,profile<,shapefile>>)>

Description: Plots the X, Y and Z excitation (inversion) profile for a pulse shape that has been

generated with the Pbox software. If shape names is not provided, the last simulation data stored in the shapelib/pbox.sim file are plotted.

Arguments: The axisflag and profile arguments can be given in any order.

axisflag is 'y' to display the full spectrum and a frequency scale, or 'n'

to suppress the scale and spectrum. The default is 'n'.

profile is a character string identifying the desired profile. 'xyz' selects X, Y, and Z (inversion) profiles; 'xy' selects only the excitation (transverse) profiles; 'x' selects only the X transverse excitation profile; and 'z' selects

only the inversion profile. The default is 'xyz'.

shapefile is the name of a \*.RF or \*.DEC file, including the extension.

Examples: pprofile

pprofile('y','x')

pprofile('xy','n','softpls.RF')

See also: NMR Spectroscopy User Guide

Related: dprofile Display pulse excitation profile (M)

Pbox Pulse shaping software (U)

### pps Plot pulse sequence (C)

Syntax: pps<(file<,x,y,width,height>)>

Description: Plots pulse sequences. The plotted picture consists of three to five parts. At the

top is the transmitter pulse sequence. Below that is the decoupler pulse sequence. Next is the second decoupler pulse sequence or gradients, depending

on the program. At the bottom is the status.

The parameter of each pulse is plotted if its length is less than 30 letters. The value of each pulse is also plotted. If its value is less than zero, a question mark "?" is plotted. The time units are displayed as letters (s, m, or u). The height of

pulses are plotted according to their power level.

Arguments: file specifies the pulse sequence to be plotted. The default is seqfil.

 $\boldsymbol{x}$  ,  $\boldsymbol{y}$  specifies the start of the plotting position with respect to the lower-left

corner of the plotter.

width, height are in proportion to wcmax and wc2max.

Examples: pps

pps('s2pul')

pps(3,50)

See also: NMR Spectroscopy User Guide

Related: dps Display pulse sequence (C)

seqfilPulse sequence name (P)wcmaxMaximum width of chart (P)

wc2max Maximum width of chart in second direction (P)

# prealfa Specify a delay for longer ring down (P)

Applicability: Systems with Varian, Inc. Cold Probes

Description: Specify a delay to be used in situations when there is a longer ring down of rf

following the last rf pulse.

This parameter is only active when qcomp='y'. prealfa should be created as a local parameter of type pulse or delay. This parameter must be created

as a local parameter of the type pulse for SpinCad Sequences.

If it is desired to use the software computed value for this delay, destroy the

prealfa parameter.

Values: User set prealfa value that may be slightly adjusted by the software to better

optimize the DSP parameters.

# prep Run prepare acquisition macro (M)

Applicability: Imaging

Description: Run the prepare acquisition macro specified by the execprep parameter.

Usually only called from panels.

Related: execute prepare macro (P)

# Presat Set up parameters for presat <sup>1</sup>H experiment (M)

Description: Set up parameters for presat <sup>1</sup>H experiment with solvent suppression.

# prevpl Display the previous 3D plane (M)

Description: Displays 2D color map of the previous 3D plane in the set of planes defined by

the parameters <code>plane</code> and <code>path3d</code>. For example, if <code>dplane(40)</code> has just been executed, <code>prevpl</code> results in the display of 3D plane 39 of that set. (If <code>prevpl</code> immediately follows the command <code>dproj</code>, an error results because there is no 3D plane whose number is <code>-1.</code>) <code>prevpl</code> is more efficient than <code>dplane</code> or <code>dproj</code> because the 3D parameter set (<code>procpar3d</code>) is not loaded into <code>VnmrJ</code>. It is assumed to have already been loaded by, for example, <code>dplane</code>

or dproj.

See also: NMR Spectroscopy User Guide

Related: dplane Display a 3D plane (M)

dproj Display a 3D plane projection (M)
dsplanes Display a series of 3D planes (M)

getplane Extract planes from a 3D spectral data set (M)

nextpl Display the next 3D plane (M)

path3d Path to currently displayed 2D planes from a 3D data set (P)

plane Currently displayed 3D plane type (P)
plplanes Plot a series of 3D planes (M)

# prescan Study queue prescan (P)

Description: This parameter keeps track of the type and status of the prescans in the study

queue.

Related: cqexp Load experiment from protocol (M)

cqrset Reset study queue parameters (M)
sqexp Load experiment from protocol (M)

sqreset Reset study queue parameters for imaging (M)

# prescan\_CoilTableRead or update the CoilTable File (M)

Syntax: prescan\_CoilTable(action,rfcoil)

Description: Manages the CoilTable file in ~/vnmrsys. Reads information about rfcoil

into the global parameter coil\_param; updates/adds information for rfcoil

from coil\_param; removes the rfcoil entry from CoilTable.

Arguments: actions for the specified rfcoil are:

read add update remove

Examples: prescan\_CoilTable('read','main')

# prescan\_tn Return tn string for a given atomic number (M)

Syntax: prescan tn(number):str

Description: Returns tn string for a given atomic number; for H1, C13, F19, P31, Na23,

Xe129 only.

Arguments: Number is the atomic number.

str is a string that can be assigned to tn.

Examples: prescan\_tn(23):tn

printer Printer device (P)

Description: Selects the printer in use on the system.

Values: A string with entries such as 'ThinkJet\_96', 'LaserJet\_300',

'jim', 'varian1', and 'Laser1'.

See also: NMR Spectroscopy User Guide

Related: showplotter Show list of currently defined plotters and printers (M)

printfile Path to the print-to-file image (P)

Description: Defines the path where an image is saved if it is printed to a file.

printformat Format of saved-to-file image (P)

Description: The format of the image to be printed to a file.

Values: 'jpeg', 'gif', 'tiff', 'bmp'

printlayout Layout of printed image (P)

Description: The layout of the printed image.

Values: 'portrait' or 'layout'

printoff Stop sending text to printer and start print operation (C)

Syntax: printoff<('clear'|file)>

Description: Stops redirection of output to printer caused by the printon command and

starts the print operation. The command printoff must be entered to

**obtain output on the printer.** Actual printing is controlled by the **vnmrprint** script in the bin subdirectory of the system directory. printoff can also clear the data in the current print file or save data to a

specified file name (i.e., print or plot to a file).

Arguments: 'clear' is a keyword to clear the print file made so far.

file specifies the name of a file to save the printout. If the file already exists,

it is overwritten.

Examples: printoff

printoff('clear')

printoff('vnmrsys/papers/peaks.list')

See also: NMR Spectroscopy User Guide

Related: printon Direct text output to printer (C)

vnmrprint Print text files (U)

printon Direct text output to printer (C)

Description: Sends information to the printer that is normally displayed in the text window.

After using printon, output from commands that use the text window, such as dg and cat, is sent to the printer and does not appear on the screen. The value of the parameter printer is used to select which printer is used.

Related: cat Output one or more files to output text window (C)

dg Display group of acquisition/processing parameters (C)

printer
Printer device (P)

# printregion Screen region to be printed (P)

Description: The region of the screen to be printed or saved to a file.

Values: 'vnmrj' -- entire VnmrJ interface.

'graphics' -- the graphics area of the VnmrJ interface.
'frames' -- selected frames from the graphics area.

# printsize Size of printed image (P)

Description: The size of the printed image.

Values: 'quarterpage', 'halfpage', 'page'

### printsend Defines where image will print (P)

Description: Defines whether the selected image will sent to a file or a printer.

Values: 'file' or 'printer'

### probe Probe type (P)

Description: Contains a string with the name of the probe currently in the magnet. This

parameter is set automatically when the addprobe macro is entered. The getparam and setparams macros use probe to retrieve and write

parameters into the current probe file.

See also: NMR Spectroscopy User Guide

Related: addnucleus Add new nucleus to existing probe file (M)

addprobe Create new probe directory and probe file (M)
getparam Receive parameter from probe file (M)
setparams Write parameter to current probe file (M)

## probeConnect Specify which nucleus can be acquired on each RF channel (P)

Applicability: VNMRS and 400 MR

Syntax: probeConnect = 'nuc1 nuc2 nuc3...'

Description: Global string parameter that does not exist by default. If present, PSG uses it to

determine which RF channel to connect to a given nucleus. The string consists of a series of space-separated nuclei. A nucleus 'X' may be used only once in the

string to match any nucleus. The parameter must match the hardware

connections. If the parameter does not match the hardware connections or does not exist, default settings are used. Default settings are to use the first channel for the for high band observe, and the second channel for the for low band observe.

Values: Any nucleus name used for tn, or 'X'.

Examples: create('probeConnect','string','global')

 $\label{eq:probeConnect} \begin{tabular}{ll} probeConnect = "H1 C13" maps H1 to channel 1, C13 to channel 2 \\ probeConnect = "H1 P31 X" maps H1 to channel 1, P31 to channel 2, \\ \end{tabular}$ 

any nucleus to channel 3.

See also: VnmrJ User Programming

Related: tn Nucleus for observe transmitter (P)

dm Nucleus for first decoupler (P)
 dm2 Nucleus for second decoupler (P)
 dm3 Nucleus for third decoupler (P)

# Probe edit Edit probe for specific nucleus (U)

Syntax: (UNIX) Probe edit probe nucleus

Description: Opens a dialog box showing all the parameters related to a specific nucleus from

the probe table.

Arguments: probe is the name of the probe.

nucleus is the specified nucleus from the probe table.

Examples: Probe edit 5mmSW H1

Related: probe edit Edit probe for specific nucleus (M)

# probe edit Edit probe for specific nucleus (M)

Syntax: probe\_edit(probe,nucleus)

Description: Opens a dialog box showing all the parameters related to a specific nucleus from

the probe table.

Arguments: probe is the name of the probe.

nucleus is the specified nucleus from the probe table.

Examples: probe edit('5mmSW','H1')

probe\_edit(probe,tn)

Related: Probe edit Edit probe for a specific nucleus (U)

# probe protectionProbe protection control (P)

Description: Controls the power check for probe protection.

See also: NMR Spectroscopy User Guide

### proc Type of processing on np FID (P)

Description: Specifies the type of data processing to be performed upon the np (t<sub>2</sub>) FID.

Similarly, parameters proc1 and proc2 specify the type of data processing on

the  $ni(t_1)$  and ni2 interferograms, respectively.

All Varian data must be processed along np with a complex Fourier transform (FT). Sequentially sampled Bruker data (the usual case) must be processed along this dimension with a real FT, while simultaneously sampled Bruker data must be processed with a complex FT.

Pure absorptive 2D data collected by the States-Haberkorn (hypercomplex) method must be processed along ni or ni2 with a complex FT.

Pure absorptive 2D data collected by the TPPI method on a Varian spectrometer can be processed in one of two ways, depending upon how the data was

collected:

```
phase=3 Complex FT, i.e., proc1='ft' (standard way)
phase=1,4 Real FT, i.e., proc1='rft' (new way)
phase2=3 Complex FT, i.e., proc2='ft'
phase2=1,4 Real FT, i.e., proc2='rft'
```

Pure absorptive 2D data collected by TPPI method on a Bruker spectrometer must be processed along ni with a real FT (i.e., proc1 = 'rft').

Values: 'ft' specifies complex FT data processing.

'rft' specifies real FT data processing.

'lp' specifies linear prediction processing on complex data. If 'lp' is selected, additional parameters must be set to fully define how the time-domain data is to be processed; see the description of the addpar command.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

Number of increments in 1st indirectly detected dimension (P)

Number of data points (P)

parlp Create parameters for linear prediction (C)

phase Phase selection (P)

phase2 Phase selection for 3D acquisition (P)
proc1 Type of processing on ni interferogram (P)
proc2 Type of processing on ni2 interferogram (P)

# proc1 Type of processing on ni interferogram (P)

Description: Specifies the type of data processing to be performed upon the ni  $(t_1)$ 

interferogram (2D). Refer to the description of proc for further information.

Values: 'ft' specifies complex Fourier transform (FT) data processing.

'rft' specifies real FT data processing.

'lp' specifies linear prediction processing on complex data. If 'lp' is selected, additional parameters must be set to fully define how the time-domain data is to be processed; see the description of the addpar command.

'ht' specifies Hadamard transform processing. If 'ht' is selected, additional parameters must be set with the addpar command. In addition, the data set

must be acquired using a Hadamard pulse sequence.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

Number of increments in 1st indirectly detected dimension (P)

Type of processing on np FID (P)

# proc1d Processing macro for simple (non-arrayed) 1D spectra (M)

Description: A generic macro for processing non-arrayed 1D spectra using a set of standard

macros. procld is called by the procplot macro, but can also be used directly. procld first tries to find a macro of the form {tn}p with the name of the observe nucleus in lower case (e.g., hlp, cl3p). If such a macro exists, it is called. If such a nucleus-specific macro is not found in the command path, minimal 1D processing is performed (the intent is to provide a well-processed spectrum in most cases): Fourier transformation (using pre-set weighting functions), automatic phasing (aphx macro), automatic integration (integrate macro), vertical scale adjustment (vsadj macro), avoiding excessive noise (noislm macro), and threshold adjustment (thadj macro).

proc1d does not work with arrayed 1D spectra: use deptproc (for DEPT-type spectra) or procarray (for all other arrayed 1D data).

See also: NMR Spectroscopy User Guide

Related: aphx Perform optimized automatic phasing (M)

c13p Process 1D carbon spectra (M)
deptproc Process arrayed dept type spectra (M)
h1p Process 1D proton spectra (M)

integrate Automatically integrate 1D spectrum (M)

noislmAvoids excessive noise (M)procarrayProcess arrayed 1D spectra (M)procplotAutomatically process FIDs (M)

thadj Adjust threshold (M)
vsadj Adjust vertical scale (M)

# proc2 Type of processing on ni2 interferogram (P)

Description: Specifies the type of data processing to be performed upon the ni2

interferogram (3D). Refer to the description of proc for further information.

Values: 'ft' specifies complex Fourier transform (FT) data processing.

'rft' specifies real FT data processing.

'lp' specifies linear prediction processing on complex data. If 'lp' is selected, additional parameters must be set to fully define how the time-domain

data is to be processed; see the description of the addpar command.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

Number of increments in 2nd indirectly detected dimension (P)

Type of processing on np FID (P)

### proc2d Process 2D spectra (M)

Description: A general 2D processing macro that tries to do the appropriate processing for as

many types of 2D experiments as possible. It uses wft2da for phase-sensitive spectra, wft2d for absolute-value 2D spectra, wft2d('ptype') for

HOM2DJ and COSYPS (absolute value). Symmetric homonuclear correlation spectra (fn=fn1, sw=sw1) in absolute-value mode is symmetrized using foldt. The resulting spectrum is then normalized (adjustment of vs and th) using nm2d and displayed (if not in background mode). proc2d is called as part of the procplot macro, but can also be used directly by the user.

See also: NMR Spectroscopy User Guide

Related: fn Fourier number in the directly detected dimension (P)

fn1 Fourier number in 1st indirectly detected dimension (P)
foldt Fold COSY-like spectrum along diagonal axis (C)

nm2d Normalize intensity of 2D spectrum (M)

procplot Automatically process FIDs (M)

Sw Spectral width in the directly detected dimension (P)
Sw1 Spectral width in the 1st indirectly detected dimension (P)

th Threshold (P)
vs Vertical scale (P)

wft2d Weight and Fourier transform 2D data (C)

wft2da Weight and Fourier transform for pure absorption 2D data (M)

### procarray Process arrayed 1D spectra (M)

Description: A generic macro for processing arrayed 1D data. It is called within the

procplot macro, but can also be called directly. It transforms all traces, phase the trace with the largest signal, scale the traces appropriately, and set up the display parameters such that the data can be plotted directly. The plotting is done in a separate macro plarray that is also called in the procplot macro.

For the display setup, procarray distinguishes between arrays with 6 or less elements, which are stacked vertically (no horizontal offset), and spectra with greater than 6 elements, which are stacked horizontally by default, unless there are too many lines, in which case a diagonally stacked display is chosen.

Horizontal stacking is mostly adequate for pulse and power calibrations, where there are usually only a few lines. Diagonally stacked displays and plots are frequently chosen for  $T_1$  and  $T_2$  experiments on entire spectra, often with many lines. The automatic stacking mode can be overridden by creating and setting a string parameter stackmode in the startup macro, or before calling procplot or procarray. Possible values for stackmode are

'horizontal', 'vertical', and 'diagonal'. DEPT-type spectra can, in principle, be also processed with procarray but, of course, no DEPT editing occurs.

editing occurs.

See also: NMR Spectroscopy User Guide

Related: deptproc Process arrayed dept type spectra (M)

plarray Plot arrayed 1D spectra (M)

proc1d Processing macro for simple (non-arrayed) 1D spectra (M)

procplot Automatically process FIDs (M)
stack Set stacking control parameter (M)

stackmode Stack control for processing arrayed 1D spectra (P)

### process Generic automatic processing (M)

Description: Processes a wide range of data types. If the apptype parameter is set, it runs the

execprocess macro if it exists. If the apptype parameter is not set it selects a macro depending on the type of data. For simple 1D spectra, process looks for a macro of form {tn}p with the observe nucleus in lower case (e.g., h1p, c13p, f19p). If no such macro is found, process calls proc1d, a generic processing macro for 1D spectra. For DEPT type data, deptproc is called. For other arrays of 1D spectra, procarray is called. For 2D spectra, proc2d is called. process by itself is called within the procplot macro.

See also: NMR Spectroscopy User Guide

Related: apptype Application type (P)

c13p Processing of 1D carbon spectra (M)
deptproc Process array of DEPT spectra (M)
execpars Set up the exec parameters (M)
execprocess Execute processing macro (P)
f19p Processing of 1D fluorine spectra (M)
h1p Processing of 1D proton spectra (M)

procld Automatically process non-arrayed 1D fids (M)

proc2d Process 2D spectra (M)

procarray Process arrayed 1D spectra (M)
procplot Automatically process FIDs (M)
tn Nucleus for observe transmitter (P)

# procplot Automatically process FIDs (M)

Syntax: procplot<(pltmod\_value)>

Description: Universal FID processing macro called usually with wexp='procplot' by

automatic acquisition macros such as h1, c13, hcapt, and hcosy. The purpose of procplot is not the data processing itself, but rather the selection

of the appropriate processing macro for a given data set.

First, procplot calls a macro process that calculates spectra; that macro by itself then selects an appropriate processing macro, like procld for non-arrayed 1D spectra. Depending whether the parameter pltmod is set to 'none' or not, procplot then calls plot, a universal plotting macro. The setting of the parameter pltmod can be temporarily overridden by specifying an alternative value as argument to procplot.

One of the concepts behind procplot is that the user should never have to modify any processing macro for customizing the processing or the output of automatic experiments or processing; this outcome can happen by selecting a parameter in the calling macro or before calling procplot.

Arguments: pltmod\_value is an alternate value for the parameter pltmod that is only

used for the current call. The values 'none' and 'off' suppress plotting. The range of possible (active) values for pltmod\_value depends on the plotting macros. Often, the parameter pltmod has no effect other than turning on or off plotting. Note that if only the calculation of a spectrum is desired, it is usually

easier to call the process macro.

Examples: procplot

procplot('none')

See also: NMR Spectroscopy User Guide

Related: deptproc Process arrayed dept type spectra (M)

plot Automatically plot spectra (M)
pltmod Determine plot mode (P)

proc1d Processing macro for simple (non-arrayed) 1D spectra (M)

proc2d Process 2D spectra (M)
procarray Process arrayed 1D spectra (M)
process Automatically calculate spectra (M)

profile Set up pulse sequence for gradient calibration (M)

Applicability: Systems with the pulsed field gradients (PFG) module.

Description: Performs an rf and gradient echo sequence that gives a high quality profile of

the sample. This sequence is used with the macro setgcal to provide gradient

strength calibration.

See also: Performa I Pulsed Field Gradient Module Installation; Pulsed Field Gradient

Modules Installation; User Programming

Related: gcal Gradient calibration constant (P)

setgcal Calibrate gradient strength from measured data (M)

proj Project 2D data (C)

Syntax: proj(exp\_number<,'sum'><,start<,width>>)

Description: Projects 2D data onto the axis parallel to the screen x-axis, which can be  $f_1$  or

 $f_2$ , depending upon the parameter trace. Two projections are available:

- *Summing projection*. The data at each frequency are summed and the result becomes the projection.
- *Skyline projection*. The data are searched and the maximum intensity at any given frequency becomes the intensity in the projection (similar to looking

at the skyline of a city where only the largest building along any given line of sight is visible).

Phase-sensitive data can be projected, but the resulting projection can only be displayed in an absolute-value mode

Arguments: exp number is the number of the experiment, from 1 through 9, in which the

resulting spectrum is stored.

'sum' is a keyword to use the summing projection. The default is skyline. start defines the starting trace, in Hz. The default is to project all data.

width defines the width of the traces, in Hz, to be projected. The default is to project all data. If width is supplied as zero, a single trace corresponding to the

start frequency will be stored.

Examples: proj(3)

proj(5,'sum')

proj(4,3\*sfrq,6\*sfrq)

See also: NMR Spectroscopy User Guide

Related: Select mode for 2D data display (P)

#### Set up parameters for <sup>1</sup>H experiment (M) Proton

Set up parameters for <sup>1</sup>H experiment. Description:

#### Macro to start ProTune (M) protune

Applicability: Liquids, Walkup, Automation

Syntax: protune(freq1 <, match1 <, freq2 <, match2>>>)

protune('argument',<\$nucleus,<\$target>>) protune('exec', command1 <, command2, ...>)

Description: Tunes to frequency freq1 MHz if the first argument is the frequency in MHz.

> Executes a sequence of arbitrary tuning commands if the first argument is the keyword exec. Any command that can be typed into the command line box in

the ProTune GUI display is allowed.

Arguments: First case:

freq1 MHz — first tuning frequency in MHz

match1 — % of optimum for the first frequency, 5% is the default

freq2 MHz — optional second tuning frequency in MHz

match2 — % of optimum for the second frequency, 5% is the default.

Second case:

'argument' may have the following values:

no argument opens Tune Probe dialog for probe tuning. Select the or 'popup'

nucleus to tune and how coarse to tune using the

buttons and menus in the dialog box.

'calibrate' open ProTune calibration interface.

'nucleus' tune using specified nucleus - \$nucleus must be

specified.

\$nucleus — Nucleus to tune to, 'H1', 'C13' ...

\$target — Tune target level, 0.1(finest) to 100 (coarsest), defaults to 5 if no value is specified.

Third case:

exec — keyword that precedes a command or string of commands.

Examples: protune('exec', 'setTuneFrequency 0 599.96e6')

Tunes the probe to 599.96 MHz.

See also: User Guide Liquids and VnmrJ Walkup

Related: atune ProTune present (P)
protunegui Macro to start ProTune in graphical user interface (M)

plockport Port number to use to lock out multiple ProTune processes (P)
probeConnect Specify which nucleus can be tuned on each RF channel (P)

settune set up tune parameters for automation showprotunegui show the graphical interface while tuning (P) tchan RF channel number used for tuning (P) tugain Receiver gain used in tuning (P)

tunehf Tune both H1 and F19 on an HFX probe (M)

tunesw Width of the tuning sweep in Hz (P)

tunematch Default match target, in percent of optimum (P)

Transmitter power used in tuning (P)

tuneResult Message indicating how well the tuning succeeded (P)

tunemethod Method to use for tuning (P)
wtune Specify when to tune (P)

wtunedone What to do after tuning is done (P)

xmtune Check tune parameter during automation (M)

# protune Shell script for start ProTune operation (U)

Applicability: Automation

Description: Starts and stops ProTune. Usually called from Protune macros.

See also: NMR Spectroscopy User Guide and VnmrJ Walkup

Related: protune (M Macro to start ProTune (M)

# protunegui Macro to start ProTune in graphical user interface (M)

Applicability: Liquids, VnmrJ Walkup, Automation

Syntax: protune('argument',<\$nucleus,<\$target>>)

Description: Starts ProTune in graphical mode.

Arguments: see protune (M)

See also: NMR Spectroscopy User Guide and VnmrJ Walkup

Related: protune Macro to start ProTune (M)

### prune Prune extra parameters from current tree (C)

Syntax: prune(file)

Description: Destroys parameters in the current parameter tree that are not also defined in the

supplied parameter file. prune is used to remove leftover parameters from previous experimental setups. Recalling a new parameter set into an experiment

has a similar effect and, in general, prune is not required.

Arguments: file is the path of a parameter file.

Examples: prune(systemdir+'/parlib/cosyps.par/procpar')

prune('/vnmr/par400/stdpar/H1.par/procpar')

prune (userdir+'/exp3/curpar')

See also: User Programming

Related: create Create new parameter in a parameter tree (C)

destroy Destroy a parameter (C)

display Display parameters and their attributes (C)

**fread** Read parameters from file and load them into a tree (C)

**fsave** Save parameters from a tree to a file (C)

# pscale Plot scale below spectrum or FID (C)

Syntax: pscale<(<rev><,axis><,label><,vp0><,sp0><,color><,pen>)>

Description: Plots a scale under a spectrum or FID.

Arguments: rev – reverses the direction of the scale. That is, the smaller numbers will be at

the left side of the scale. If used, 'rev' must be the first argument.

axis – If the letter p, h, k, etc. is supplied, it will be used instead of the current value of the parameter axis. For an FID scale, if the letter s, m, or u is supplied, it will be used instead of the current value of the parameter axisf.

label – If a string of 2 or more characters is supplied, it will be used as the axis label.

vp0 – This is supplied as the first real number. It defines the vertical position where the scale is drawn. The default is 5 mm below the current value of the parameter vp.

sp0 – This is supplied as the second real number. It is a modified start of plot. If, for example, the display is from 347 to 447 hz, but the scale is desired to read 0 to 100 hz., sp0 would be input as 0.

wp0 – This is supplied as the third real number. It is a modified width of plot. If, for example, the display is from 347 to 447 hz, but the scale is desired to read 0 to 550 Units. sp0 would be input as 0, wp0 would be 550, and the label would be 'Units'.

An optional color or pen number can be supplied to dscale or pscale. The available colors and pens are: 'red', 'green', 'blue', 'cyan',

'magenta','yellow','black','white'
'pen1','pen2','pen3',...,'pen8'

Examples: pscale

pscale(20)

pscale('h',0,'pen2')
pscale('fid','m')
pscale('h',vp-10,0)

See also: NMR Spectroscopy User Guide

Related: axis Axis label for displays and plots (P)

axisf Axis label for FID displays and plots (P)
dscale Display scale below spectrum or FID (C)
vp Vertical position of spectrum (P)

pseudo Set default parameters for pseudo-echo weighting (M)

Syntax: pseudo<(C1,C2,C3,C4)>

Description: Generates an initial guess at good weighting parameters for absolute-value 2D

experiments. To generate modified guesses, four coefficients are allowed to set

the values of the weighting functions.

Arguments: C1 sets  $\frac{1b}{-0.318}$  (C1\*at). The default value of C1 is 0.0625.

C2 sets gf=C2\*at. The default value of C2 is 0.25.

C3 sets lb1=-0.318/(C3\*(ni/sw1)) but is used with 2D experiments

only. The default value of C3 is 0.0625.

C4 sets gf1=C4\* (ni/sw1) but is used with 2D experiments only. The default

value of C4 is 0.25.

Examples: pseudo

pseudo(.1,.4,.2,.5)

See also: NMR Spectroscopy User Guide

Related: sinebell Select default parameters for sinebell weighting (M)

### psg Display pulse sequence generation errors (M)

Description: Helps identify the problem if, after entering go or su, etc., the message is

returned that pulse sequence generation (PSG) aborted abnormally. Any parameters that are not found are listed. This information is stored in the user's directory (vnmrsys) in a text file named psg.error. If the message "Maximum communication retries exceeded, Experiment unable to be sent" is displayed, a program communications problem is indicated. Consult the system

operator for assistance.

See also: User Programming

# psggen Compile a user PSG object library (M,U)

Description: A user PSG (pulse sequence generation) kit is supplied that allows editing low-

level pulse sequence code. psggen compiles these edits so that subsequent pulse sequence generation with the seqgen command uses the customized

pulse sequence source.

See also: *User Programming* 

# psgset Set up parameters for various pulse sequences (M)

Syntax: psgset(file,par1,par2,...,parN)

Description: Sets up parameters for various pulse sequences using information in a parlib

file. Rather than returning the entire parameter file, psgset returns the parameters listed. psgset, in general, is never entered from the keyboard but

is used as part of experiment setup macros.

Arguments: file is the file from the user or system parlib that provides information on

setting up the parameters listed. The parameters  ${\tt seqfil}$  and  ${\tt pslabel}$  are set

to the supplied file name.

par1, par2, ..., pN are 1 to 11 parameters to be returned from parlib.

Examples: psgset('cosy','dg','ap','ss','d1','axis','phase')

See also: *User Programming* 

Related: pslabel Pulse sequence label (P)

seqfil Pulse sequence name (P)

### psgupdateon Enable update of acquisition parameters (C)

Description: Permits the interactive updating of acquisition parameters.

See also: SpinCAD

Related: psgupdateoff Prevent update of acquisition parameters (C)

updtparam Update specified acquisition parameters (C)

psgupdateoff Prevent update of acquisition parameters (C)

Description: Prevents the interactive updating of acquisition parameters.

See also: SpinCAD

Related: psgupdateon Enable update of acquisition parameters (C)

updtparam Update specified acquisition parameters (C)

pshape Plot pulse shape or modulation pattern (M)

Syntax: pshape<(pattern.ext)>

Description: Plots the real (X) and imaginary (Y) components of a shaped pulse. Any type of

waveform (.RF, .DEC or ,GRD) can be plotted.

Arguments: pattern is the name of a shape or pattern file specified by an absolute file

name, relative file name, or a simple pattern file name. ext is a file name extension that specifies the file type. In the case of a simple file name, dshape searches for the file in the local directory, then in the user's shapelib, and finally in the directory /vnmr/shapelib. If pattern.ext is not given, pshape displays the last created waveform stored in the pbox.fid file.

Examples: pshape

pshape('my shape.DEC')

See also: NMR Spectroscopy User Guide

Related: dshape Display the last created pulse shape (M)

Pbox Pulse shaping software (U)

pshapef Plot the last created pulse shape (M)

Description: Plots real (X) and imaginary (Y) components of the last created shaped pulse.

See also: NMR Spectroscopy User Guide

Related: dshape Display the last created pulse shape (M)

Pbox Pulse shaping software (U)

pshr PostScript High Resolution plotting control (P)

Applicability: ALL

Syntax: pshr=<value>

Description: Global parameter that controls whether a 1D spectrum is plotted in hi-resolution

mode or not. A hi-resolution plot is one in which every data point is represented in the plot. The standard resolution plot determines maximum and minimum values over small regions and plots those. The parameter pshr can have the

values 1 for hi-res and 0 for standard plot.

Values: 0 for standard resolution

1 for high resolution.

Related: pl Plot spectra (C)

PostScript Line Width control (P)

pslabel Pulse sequence label (P)

Description: Contains the text to be displayed in the Seq: field on the top line of the screen.

This string may be different from the pulse sequence name selected with seqfil. However, the string in seqfil is the name of the pulse sequence

searched for when an experiment is started. Generally seqfil=pslabel, and when seqfil is set, the system sets pslabel to the same string.

See also: NMR Spectroscopy User Guide

Related: seqfil Pulse sequence name (P)

### pslw PostScript Line Width control (P)

Applicability: ALL

Syntax: pslw=<value>

Description: Global parameter that adjusts the line width of PostScript plots.

Values: 0 (narrowest) to 100 (widest) line width.

Related: pl Plot spectra (C)

PostScript High Resolution plotting control (P)

### pssl Plot Arrayed Numbers (C)

Syntax: pssl(<options>)

Description: Plots a label for each element in a set of stacked spectra. The label is an integer

value from 1 up to the number of spectra in the display.

Arguments: options can be any of the following:

• 'all' is a keyword to display all of the spectra.

- 'int' is a keyword to display only the integral, independently of the value of the parameter intmod
- 'top' or 'side' are keywords that cause the spectrum to be displayed either above or at the left edge, respectively, of a contour plot. This assumes that the parameters sc, wc, sc2, and wc2 are those used to position the contour plot.
- 'dodc' is a keyword for all spectra to be drift corrected independently.
- 'red', 'green', 'blue', 'cyan', 'magenta', 'yellow', 'black', and 'white' are keywords that select a color.
- 'pen1', 'pen2', 'pen2' ... are keywords that pens.
- 'nopars' prevents the display commands from drawing the parameters at the bottom of the graphics screen.
- 'custom' uses the parameters shownumx (x position) and shownumy (y position), counting from bottom left of every spectrum.
- 'reverse' rotate the text by 90° useful if the arrayed parameter values are long with respect to the width of the individual sub-spectra.
- 'value' The values of up to two simultaneous arrays are displayed. Diagonal arrays are allowed. The second parameter is shown in different color). The name of the arrayed parameter(s) is also shown. If used on a one-dimensional array representation of a 2D spectrum, ni and phase (in case of phase sensitive 2Ds) parameters are shown.
- 'list=xxx' produces a display of the values contained in the arrayed parameter xxx.
- 'format=yyy' uses the format yyy to control the plot of each label. See the write command for information about formats.

Examples: pssl

Related:

pssl('top','left')

pssl('value','format=%3.1f')

See also: NMR Spectroscopy User Guide

Related: dssl Label a display of stacked spectra (M)

write Write formatted text to a device (C)

### ptext Print out a text file (M)

Syntax: ptext(file)

Description: Prints out a text file.

Arguments: file is the name of the text file.

Examples: ptext('/vnmr/maclib/ptext')

ptext(curexp+'/dept.out')

See also: NMR Spectroscopy User Guide

curexp Current experiment directory (P)

dtext Display a text file in the graphics window (C)

Look up words and lines from a text file (C)

pltext Plot a text file (C)

Display text or set new text for current experiment (C)

textvi Edit text file of current experiment (M)
vi Edit text file with vi text editor (C)

# ptspec3d Region-selective 3D processing (P)

Description: Sets whether region-selective 3D processing occurs. If ptspec3d does not

exist, it is created by the macro <code>par3d</code>. <code>ptspec3d</code> is functional at this time only for the f<sub>3</sub> dimension. If <code>ptspec3d='ynn'</code>, only the currently displayed region of f<sub>3</sub> is retained as non-zero values after the f<sub>3</sub> transform in the 3D FT. A larger f<sub>3</sub> region may be kept to ensure that the number of hypercomplex f<sub>3</sub> points is a power of 2; but that portion of the f<sub>3</sub> spectrum that is retained outside of the currently displayed region contains only zeroes. This 3D utility can reduce the fully transformed 3D data size by factors of 2 to 4, especially in some of the

triple resonance experiments.

Values: A three-character string such as 'nnn', 'nny', 'nyn', etc. The default is

'nnn'. The first character refers to the  $f_3$  dimension (sw, np, fn); the second character, to the  $f_1$  dimension (sw1, ni, fn1); and the third character, to the  $f_2$  dimension (sw2, ni2, fn2). Each character may take one of two values: 'n' for no region-selective processing in the relevant dimension, or 'y' for region-

selective processing in the relevant dimension.

See also: NMR Spectroscopy User Guide

Related: fiddc3d 3D time-domain dc correction (P)

fn Fourier number in directly detected dimension (P)
fn1 Fourier number in 1st indirectly detected dimension (P)
fn2 Fourier number in 2nd indirectly detected dimension (P)

Perform a 3D Fourier transform (M)

Number of increments in 1st indirectly detected dimension (P)

Number of increments in 2nd indirectly detected dimension (P)

Number of data points (P)

ntype3d N-type peak selection in  $f_1$  or  $f_2$  (P)

par3d Create 3D acquisition, processing, display parameters (C)

specdc3d 3D spectral dc correction (P)

Sw Spectral width in directly detected dimension (P)
Sw1 Spectral width in 1st indirectly detected dimension (P)
Sw2 Spectral width in 2nd indirectly detected dimension (P)

ptsval PTS frequency synthesizer value (P)

Description: Configuration parameter for the frequency of the PTS synthesizer on each

channel. Every broadband system is equipped with a PTS frequency synthesizer as part of broadband frequency generation. The frequency of the unit is marked on its front panel. The value is set for each channel using the Synthesizer label

in the Spectrometer Configuration window.

Values: 0 (Not Present choice in Spectrometer Configuration window); 160, 200, 250,

320, 500, 620, 1000 (PTS 160, PTS 200, PTS 250, PTS 320, PTS 500, PTS 620, PTS 1000 choices in Spectrometer Configuration window, respectively).

See also: *VnmrJ Installation and Administration*.

Related:

config Display current configuration and possibly change it (M)

latch Frequency synthesizer latching (P)
overrange Frequency synthesizer overrange (P)

pulseinfo Shaped pulse information for calibration (M)

Syntax: pulseinfo<(shape,pulse\_width<,reference\_power>)>

:width,power

Description: Returns or prints a table with the bandwidth and predicted pulse power settings

for a given pulse shape. No parameter settings are changed. The necessary data is contained in the file shapeinfo in the system shapelib subdirectory.

Arguments: shape is the name of the pulse shape. The default is the system interactively

prompts the operator for the name of the shape and the duration of the pulse and then prints a table containing the bandwidth of that pulse and the predicted pulse

power settings.

pulse width is the duration of the pulse, in µs.

reference\_power is a value, in dB, for power calculations. The default is 55. This value replaces the assumption used for power calculation that pw90 is

set for a towr of 55.

width returns the bandwidth of that pulse, in Hz.

power returns the predicted 90° pulse power settings.

Examples: pulseinfo('gauss',1000):bw,pwr

See also: User Programming

Related: bandinfo Shaped pulse information for calibration (M)

pw90 90° pulse width (P)

Observe transmitter power level with linear amplifiers (P)

pulsetool RF pulse shape analysis (U)

Syntax: pulsetool <-shape filepath>

Description: Enables examination of shaped rf pulses. It is started from a UNIX window.

Arguments: The optional -shape filepath specifies the name of an rf pulse template

file that is displayed when pulsetool is started.

Examples: pulsetool

pulsetool -shape /vnmr/shapelib/sinc.RF

purge Remove macro from memory (C)

Syntax: purge<(file)>

Description: Removes one or more macros from memory, freeing extra memory space.

Arguments: file is the name of a macro file to be removed from memory. The default is to

remove all macros that have been loaded into memory.

CAUTION: The purge command with no arguments should never be called from

a macro. The purge command with an argument should never be

called by the macro being purged.

Examples: purge

purge(' sw')

See also: User Programming

Related: macrold Load a macro into memory (C)

# puttxt Put text file into a data file (C)

Syntax: puttxt(file)

Description: Copies text from current experiment into a data file.

Arguments: file is the name of a data file (i.e., a directory with a .fid or .par suffix).

Do not include the suffix in the name provided to file.

Examples: puttxt('mydata')

See also: NMR Spectroscopy User Guide

Related: gettxt Get text file from another file (C)

### putwave Write a wave into Pbox.inp file (M)

Syntax: putwave(sh,bw,pw,ofs,st,ph,fla,trev,d1,d2,d0)

Description: Sets up a single excitation band in the Pbox.inp file. An unlimited number of

waves can be combined by reapplying putwave.

Arguments: 1 to 11 wave parameters in the following predefined order:

sh is the name of a shape file. bw is the bandwidth, in Hz. pw is the pulsewidth, in sec. of s is the offset, in Hz.

st is a number specifying the spin status: 0 for Mz, or 1 for Mxy.

ph is the phase (or phase cycle, see wavelib/supercycles).

fla is the flip angle. Note that fla can override the default flip angle.

trev concerns time reversal. It can be used to cancel time reversal if spin status

(st) is set to 1 for Mxy.

d1 is the delay, in sec, prior the pulse.

d2 is the delay, in sec, after the pulse.

d0 is a delay or command prior to d1. If d0=a, the wave is appended to the

previous wave.

Examples: putwave('eburp1')

putwave('GARP',12000.0)

putwave('esnob',600,-1248.2,1,90.0,'n','n',0.001)

Related: Pbox Pulse shaping software (U)

write a wave definition string into the Pbox.inp file (M)

pw Enter pulse width pw in degrees (C)

Syntax: pw(flip angle,<90 pulse width>)

Description: Calculates the flip tim, in µs, given a desired flip angle and 90° pulse. The value

is entered into the parameter pw.

Arguments: flip angle is the desired flip angle, in degrees.

90 pulse width is the 90° pulse length, in µs. The default is the value of

parameter pw90, if it exists.

Examples: pw(30)

pw(90,12.8)

See also: NMR Spectroscopy User Guide

Related: ernst Calculate the Ernst angle pulse (C)

pw Pulse width (P) pw90 90° pulse width (P)

pw Pulse width (P)

Description: Length of the final pulse in the standard two-pulse sequence. In "normal" 1D

experiments with a single pulse per transient, this length is the observe pulse

width.

Values: 0, 0.1 μs to 8190 sec, smallest value possible is 0.1 μs, finest increment possible

is 12.5 ns.

See also: NMR Spectroscopy User Guide

Related: p1 First pulse width (P)

pw Enter pulse width parameter pw in degrees (C)

pw90 90° pulse width (P)

Description: Length of the 90° pulse. pw90 is not used by pulse sequences directly, but is

used by a number of commands to assist in setting up special experiments. pw90 is also used by certain output programs to be able to print the value of the pulse width in degrees instead of microseconds. Note that this parameter must be updated by the user and is not automatically determined or magically correct

under all circumstances.

Values: 0, 0.1 μs to 8190 sec, smallest value possible is 0.1 μs, finest increment possible

is 12.5 ns.

See also: NMR Spectroscopy User Guide

 $Related: \quad {\tt AC1S-AC11S} \quad \quad Autocalibration \ macros \ (M)$ 

pw Enter pulse width parameter pw in degrees (C)

pwd Display current working directory (C)

Syntax: pwd<:directory>

Description: Displays the path of the current working directory.

Arguments: directory is a string variable with the path of the current directory.

Examples: pwd:\$name

Related: cd Change working directory (C)

dir List files in current directory (C)

1f List files in current directory (C)

List files in current directory (C)

# pwpat Shape of refocusing pulse (P)

Applicability: Systems with imaging capabilities.

Description: Specifies the shape of the refocusing pulse pw in imaging experiments

Values: 'hard', 'sinc', 'qauss', 'sech', 'sine', or any shape resident in the

system pulse shape library or libraries.

See also: VnmrJ Imaging NMR

Related: plpat Shape of an excitation pulse (P)

Pulse width (P)

# pwr Set power mode in directly detected dimension (C)

Description: Selects the power spectra display mode by setting dmg='pwr'. In the power

*mode*, each real point in the displayed spectrum is calculated as the sum of the squares of the real and imaginary points comprising each respective complex data point. All information, including noise, is positive and the relationship

between signal and noise is non-linear.

For multidimensional data, pwr has no effect on data prior to the second Fourier transform. If pmode='full', pwr acts in concert with the commands ph1,

avl or pwrl to yield the resultant contour display for the 2D data.

See also: NMR Spectroscopy User Guide

Related: av Set abs. value mode in directly detected dimension (C)

av1 Set abs. value mode in 1st indirectly detected dimension (C)

dmg Data display mode in directly detected dimension (P)

ft Fourier transform 1D data (C)

ft1d Fourier transform along f<sub>2</sub> dimension (C)

ft2d Fourier transform 2D data (C)

pa Set phase angle mode in directly detected dimension (C)

Set phase angle mode in 1st indirectly detected dimension (C)

ph Set phased mode in directly detected dimension (C)
ph1 Set phased mode in 1st indirectly detected dimension (C)

pmode Processing mode for 2D data (P)

pwr1 Set power mode in 1st indirectly detected dimension (C)
pwr2 Set power mode in 2nd indirectly detected dimension (C)

wft Weight and Fourier transform 1D data (C)
wftld Weight and Fourier transform f<sub>2</sub> of 2D data (M)
wftld Weight and Fourier transform 2D data (M)

# pwr1 Set power mode in 1st indirectly detected dimension (C)

Description: Selects the power spectra display mode along the first indirectly detected

dimension by setting <code>dmg1='pwr1'</code>. If the parameter <code>dmg1</code> does not exist, <code>pwr1</code> creates it and sets it to 'pwr1'. In the *power mode*, each real point in the displayed trace is calculated as the sum of the squares of the real and imaginary points comprising each respective complex data point. For hypercomplex data, the real-real and imaginary-real points from each respective hypercomplex data

point are used in the summation. In this mode, all information, including noise, is positive and the relationship between signal and noise is non-linear.

The pwrl command is only needed if mixed-mode display is desired. If the parameter dmgl does not exist or is set to the null string, the display mode along the first indirectly detected dimension defaults to the display mode of the directly detected dimension (characterized by the parameter dmg). For the contour display of multidimensional data, the result of pwrl is the same as for traces, provided that pmode='partial' or pmode=''.

See also: NMR Spectroscopy User Guide

Related: dmg1 Data display mode in 1st indirectly detected dimension (P)

pa Set phase angle mode in directly detected dimension (C)
pal Set phase angle mode in 1st indirectly detected dimension (C)

pmode Processing mode for 2D data (P)

pwr Set power mode in directly detected dimension (C)pwr2 Set power mode in 2nd indirectly detected dimension (C)

### pwr2 Set power mode in 2nd indirectly detected dimension (C)

Description: Selects the power spectra display mode along the second indirectly detected

dimension by setting dmg2='pwr2'. If dmg2 does not exist or is set to the null string, pwr2 will create dmg2 and set it equal to 'pwr2'. In the *power mode*, all information, including noise, is positive and the relationship between signal and noise is non-linear. Each real point in the displayed trace is calculated as the sum of the squares of the real and imaginary points comprising each respective complex data point. For hypercomplex data, the real-real and imaginary-real points from each respective hypercomplex data point are used in

the summation.

The pwr2 command is only needed if mixed-mode display is desired. If the parameter dmg2 does not exist or is set to the null string, the display mode along the second indirectly detected dimension defaults to the display mode of the directly detected dimension (characterized by the parameter dmg). For the contour display of multidimensional data, the result of pwr2 is the same as for traces, provided that pmode='partial' or pmode=''.

See also: NMR Spectroscopy User Guide

Related: av2 Set abs. value mode in 2nd indirectly detected dimension (C)

dmg2 Data display mode in 2nd indirectly detected dimension (P)

Fourier transform along f<sub>2</sub> dimension (C)

ft2d Fourier transform 2D data (C)

Set phased mode in 2nd indirectly detected dimension (C)

pmode Processing mode for 2D data (P)

Set power mode in directly detected dimension (C)

### pwsadj Adjust pulse interval time (M)

Applicability: Systems with waveform generators.

Syntax: pwsadj(shape file,pulse parameter)

Description: Adjusts the pulse interval time so that the pulse interval for the specified shape

is an integral multiple of 100 ns. This ensures there is no time truncation error

in executing the shaped pulse by waveform generators.

Arguments: shape file is a file name of a shaped pulse file. The name can be specified

with or without the .RF file extension. pwsadj first looks for the file name specified by shape file in the user's shapelib directory. If the file

specified is not found there, pwsadj then looks in the system shapelib

directory.

pulse parameter is a string containing the adjusted pulse interval time.

Examples: pwsadj('pulse12','pulseparam')

See also: User Programming

Related: dmfadj Adjust decoupler tip-angle resolution time (M)

dmf2adj Adjust second decoupler tip-angle resolution time (M)

# pwxcal Decoupler pulse calibration (M)

Description: Provides an interactive method of selecting the decoupler (first, second, or

third) and the nucleus (<sup>13</sup>C, <sup>15</sup>N, or <sup>31</sup>P) to calibrate. The pwxcal pulse sequence determines the pulse width characteristics of the probe's decoupler channel(s) in indirect detection or triple resonance experiments. pwxcal can

also be used to determine the rf field homogeneity of the decoupler.

The parameter pwx1 is arrayed to calibrate the 90° pulse width on the first decoupler. If a second decoupler is present, the parameter pwx2 is arrayed to calibrate the 90° pulse width on that decoupler. If a third decoupler is present, the parameter pwx3 is arrayed to calibrate the 90° pulse width on that decoupler. Other parameters include: jC13 is the  $^{13}C^{-1}H$  coupling, constant, jN15 is the  $^{15}N^{-1}H$  coupling constant, jP31 is the  $^{31}P^{-1}H$  coupling constant,

and iname is a selected calibration nucleus.

See also: System Administration

### pxbss Bloch-Siegert shift correction during Pbox pulse generation (P)

Description: A flag to enable or disable Bloch-Siegert shift correction during the creation of

Pbox pulses.

Values: 'y' enable Bloch-Siegert shift correction

'n' disable Bloch-Siegert shift correction

Default value is 'y'.

See also: NMR Spectroscopy User Guide

Related: <a href="https://https://html.nim.nim.nim.nim.ni">https://htt

# pxrep Flag to set the level of Pbox reports (P)

Description: A flag to set the level of Pbox debug messages displayed at the start of

acquisition.

Values: 'y' shows all Pbox reports.

'h' shows the Hadamard matrix.

'n' shows no reports. Default value is 'nnn'.

See also: NMR Spectroscopy User Guide

Related: htfrq1 Hadamard frequency list in ni (P)

### pxset Assign Pbox calibration data to experimental parameters (M)

Syntax: pxset<(file.ext)>

Description: Retrieves experimental settings from a file and assigns them to corresponding

experimental parameters using a dialog form. If no file name is provided, pxset extracts data from the Pbox.cal file that contains the output data of

the last created waveform

Arguments: file.ext is the name of a shape or pattern file.

Examples: pxset

pxset('Pbox.RF')

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

pboxget Extract Pbox calibration data (M)

### pxshape Generates a single-band shape file (M)

Syntax: pxshape('sh bw/pw ofs st ph fla trev \

d1 d2 d0',name,disp)

Description: Generates a single-band waveform based on wave definition provided as a

single string of wave parameters.

Arguments: A single string of 1 to 12 wave parameters in predefined order. Note that a single

quote is required at the start and the end of the entire string, but no single quotes

are required surrounding characters and strings inside the entire string.

sh is the name of a shape file.

bw/pw is either the bandwidth, in Hz, or the pulsewidth, in sec.

ofs is the offset, in Hz.

st is a number specifying the spin status: 0 for Mz, or 1 for Mxy.

ph is the phase (or phase cycle, see wavelib/supercycles).

fla is the flip angle. Note that fla can override the default flip angle.

trev is a time reversal. This can be used to cancel time reversal if spin status

(st) is set to 1 for Mxy.

d1 is the delay, in sec, prior the pulse.

d2 is the delay, in sec, after the pulse.

d0 is a delay or command prior to d1. If d0=a, the wave is appended to the

previous wave.

name is the output file name. An extension is optional and can be used to

override an internally defined shape type.

disp is the shape is displayed by default in the graphics window. If disp is

set to 'n', the shape is not displayed.

Examples: pxshape('eburp1','myshape.RF')

pxshape('GARP 12000.0','shape2','y')

pxshape('esnob 600.0 -1248.2 n 180.0 n n 0.001','xxx')

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

### Pxsim Simulate Bloch profile for a shaped pulse (U)

Syntax: Pxsim file <simtime <num steps <add/sub>>>

Description: Used by the dprofile macro to simulate a Bloch profile for a shaped pulse.

Pxsim extracts the information necessary for simulation from the shape header.

Only shape files containing this information can be processed.

Arguments: file is the name of a shape or pattern file including an .RF or .DEC extension.

Pxsim searches for the file in the user's shapelib (~/vnmrsys/shapelib), and if not found there, it searches in the system shapelib

(vnmr/shapelib).

simtime is the maximum simulation time (in sec) that can be provided.

num steps is the number of steps in the profile.

add/sub is add (a) or subtract (s) from the previous simulation.

Examples: Pxsim myshape.RF

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

Pxspy Create shape definition using Fourier coefficients (U)

Syntax: Pxspy file

Description: An interactive program that converts shaped pulse files into a Fourier series and

produces an output file pbox.cf in the user's shapelib (~/vnmrsys/shapelib), which can be used to create a wave definition file in the

wavelib directory. Pxspy can also be used to convert hard pulse decoupling sequences into soft ("cool") decoupling waveforms. The resulting Fourier

coefficients can depend on the number of points in the waveform.

Arguments: file is the name of a shape or pattern file, including an .RF, .DEC, or .GRD

extension. The name can be given as a relative name, absolute name, or as a simple name (i.e., with a path). If given as a simple name, Pxspy searches for the file in the user's shapelib (~/vnmrsys/shapelib), and then if not found there, it searches in the system shapelib (vnmr/shapelib).

Examples: Pxspy myshape.RF

Pxspy /vnmr/shapelib/myshape.RF
Pxspy ~vnmrsys/shapelib/myshape.RF

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

P

# Q

**qcomp** Longer dead time for longer ring down (P)

QKexp Set up quick experiment (M)

Tune probe using swept-tune graphical tool (C)

Poisplay the value of an individual parameter)

qcomp Longer dead time for longer ring down (P)

Applicability: Systems with Varian, Inc. Cold Probes

Description: Global parameter to handle longer ring down times following the rf pulse. This

is only active if dsp='i' or if dsp='r' and fsq='y'. The dead time is calculated by the software and the DSP parameters are appropriately adjusted for flat baseline and good phase properties. If it is necessary to use a user specified delay, create the prealfa parameter. qcomp is not effective in

explicit acquisition experiments. Not compatible with srof2.

Values: qcomp='y' triggers a longer dead time before the receiver is gated on for the

acquisition.

Related: prealfa Specify a delay for longer ring down (P)

dsp Type of DSP for data acquisition (P)

QKexp Set up quick experiment (M)

Syntax: QKexp(arguments)

Description: Set up parameters for quick experiment for a chained acquisition. Multiple

arguments can be given to define the chain. Default parameter values are used

by the macro and or the probe file is used.

Examples: QKexp('PROTON','COSY','HMQC')

QKexp('PROTON','CARBON','HETCOR','gCOSY')

qtune Tune probe using swept-tune graphical tool (C)

Syntax: qtune<(gain<,power>)>

Description: Displays a real-time graph showing reflected power versus frequency for tuning

probes. If the acquisition system has been recently rebooted, enter su before running qtune. Refer to the manual *NMR Spectroscopy User Guide* for a

detailed description of this tool.

Arguments: gain specifies the gain value, typically 20 to 50. The default is 50.

power specifies the power value, typically 60 to 70. The default is 60.

Examples: qtune

qtune(20) qtune(38,65)

See also: NMR Spectroscopy User Guide

Related: tugain Amount of receiver gain used by qtune (P)

su Submit a setup experiment to acquisition (M)

tune Assign frequencies (C)

Q

?

### Display the value of an individual parameter (C)

Syntax: parameter name<[index] >?

Description: The question mark displays the current numerical or string value of a parameter

when the parameter name is followed by a question mark. No change is made to the value of the parameter. To display an individual element of an parameter array, provide the index in square brackets (e.g., nt [3]? might display

"nt[3]=2")

Certain parameters can be "turned off" by setting the parameter to 'n'. The display of a parameter that is turned off will be the phrase "Not Used" followed by the actual value in parentheses. For example, if 1b is set to 1.5 and then set to 'n', entering 1b? will display 1b= Not Used (1.5). Such a parameter can be "turned on" by setting it to 'y'. It will then have its prior value.

To show a parameter's array of values or learn about its attributes, use the

display command.

Arguments: index is the integer for a selected member of an arrayed parameter.

Examples: 1b?

sw? pw[2]?

See also: NMR Spectroscopy User Guide

Related: display Display parameters and their attributes (C)

getvalue Get value of a parameter in a tree (C)

# R

Recall display parameter set (M) Recall some display parameters (C) r(n)r1-r7 Real-value storage for macros (P) ra Resume acquisition stopped with sa command (C) Weighting for different receivers (P) rcvrwt react Recover from error conditions during werr processing (M) readallshims Read all shims from hardware (M) readbrutape Read Bruker data files from 9-track tape (U) readfile Read the contents of a text file into two parameters (C) readhw Read current values of acquisition hardware (C) readlk Read current lock level (C) Read one of more parameters from a file (C) readparam Read shim coil setting for Ultra•nmr shim system (M) readultra real Create a real variable without a value (C) recon all Reconstruct images from 2D MRI fid data (C) Record keyboard entries as a macro (M) record redor1 Set up parameters for REDOR1 pulse sequence (M) Restore 2D DOSY display from sub experiment (M) redosy reff1 Reference f2 Indirect Dimension from Observe Dimension (M) Reference f2 Indirect Dimension from Observe Dimension (M) reff2 reffrq Reference frequency of reference line (P) reffra1 Reference freq. of reference line in 1st indirect dimension (P) Reference freq. of reference line in 2nd indirect dimension (P) reffrq2 Position of reference frequency (P) refpos refpos1 Position of reference frequency in 1st indirect dimension (P) Position of reference frequency in 2nd indirect dimension (P) refpos2 refsource1 Center frequency in 1st indirect dimension (P) refsource2 Center frequency in 2nd indirect dimension (P) region Divide spectrum into regions (C) relayh Set up parameters for RELAYH pulse sequence (M) rename Move and/or rename a file (C) reqparcheck Flag which enables/disables required parameters (P) regparclear Clears the parameters in required parameter list (M) List of required parameters (P) reqparlist Tests whether required parameters are set (M) regpartest resetf3 Reset parameters after a partial 3D Fourier transform (M) resetplotter Reset plotter to system plotter (M) resolv Set resolution enhancement parameters (M) restorenuctable Calculate and (Re-)store accurate nuctable (M) Resume paused acquisition queue (C) resume return Terminate execution of a macro (C) System software revision level (P) rev System software preparation date (P) revdate

RF band in use (P) rfband rfblk Reverse FID block (C) Independent control of rf channel selection (P) rfchannel Type of rf channel (P) rfchtype Reverse FID data (C) rfdata rfl Reference peak position in directly detected dimension (P) rfl1 Reference peak position in 1st indirectly detected dimension (P) rf12 Reference peak position in 2nd indirectly detected dimension (P) rfp Reference peak frequency in directly detected dimension (P) Reference peak freq. in 1st indirectly detected dimension (P) rfp1 Reference peak freq. in 2nd indirectly detected dimension (P) rfp2 rftrace Reverse FID trace (C) rftype Type of rf generation (P) RF waveform generator (P) rfwg right Set display limits to right half of screen (C) Determine an operator's specified right (C) rights rinput Input data for a regression analysis (M) rl Set reference line in directly detected dimension (M) rl1 Set reference line in 1st indirectly detected dimension (M) r12 Set reference line in 2nd indirectly detected dimension (M) Delete file (C) rm rmdir Remove directory (C) rmsAddData Add transformed data files with weighting (U) Convert the parameter to a ROESY experiment (M) Roesy Convert the parameter set to a Roesy1d experiment (M) Roesy1d rof1 Receiver gating time preceding pulse (P) Receiver gating time following pulse (P) rof2 Receiver gating time following T/R switch (P) rof3 Rotate 2D data (C) rotate Rotor synchronization (P) rotorsync Zero-order phase in directly detected dimension (P) rp Zero-order phase in 1st indirectly detected dimension (P) rp1 Zero-order phase in 2nd indirectly detected dimension (P) rp2 Retrieve FIDs (M) rt Return Spinsight data into current experiment (C) rtcmx Retrieve parameters (M) rtp Retrieve shim coil settings (C) rts rttmp Retrieve experiment data from experiment subfile (M) Retrieve individual parameters (C) rtv Retrieve parameters based on rtx rules (C) rtx

### r Recall display parameter set (M)

Syntax: (1) rset\_number

(2) r(set number)

Description: Recalls the parameters sp, wp, sp1, wp1, sp2, wp2, sc, wc, sc2, wc2, ho,

vo, vs, and ai/nm of a selected display parameter set. Not recalled are phase

parameters, drift correction parameters, integral reset parameters, and reference parameters. This allows, for example, saving a set of display parameters, adjusting the phase or drift correction, and later recalling the display parameters

without und0oing the new phase or drift correction.

Arguments: set number is the number, from 1 to 9, of a display parameter set.

Examples: r2 r(3)

See also: NMR Spectroscopy User Guide

Related: аi Select absolute intensity mode (C)

Full recall of a display parameter set (M) fr

ho Horizontal offset (P)

Select normalized intensity mode (C) nm S Save display parameters as a set (M)

Start of chart (P) SC

Start of chart in second direction (P) sc2

Start of plot in directly detected dimension (P) sp Start of plot in 1st indirectly detected dimension (P) sp1 Start of plot in 2nd indirectly detected dimension (P) sp2

Vertical offset (P) vo Vertical scale (P) VS WC Width of chart (P)

Width of chart in second direction (P) wc2

Width of plot in directly detected dimension (P) พท wp1 Width of plot in 1st indirectly detected dimension (P) Width of plot in 2nd indirectly detected dimension (P) wp2

#### r(n)Recall some display parameters (C)

Applicability: All

Syntax: r(n<,noupdate>)

Description: r(n) recalls only the following parameters: sp, wp, sp1, wp1, sp2, wp2, sc,

wc, sc2, wc2, ho, vo, vs, and ai/nm.

noupdate — as a second argument prevents the automatic update of

interactive programs.

Arguments: n=1 to 9

See also: User Programming

Related: fr(n) Recall all the parameters of the specified display parameter set (C)

> s(n)Save a copy of the current values of all display parameters (C)

#### Real-value storage for macros (P) r1-r7

Description: The seven parameters r1, r2, r3, r4, r5, r6, and r7 are available in each

experiment for macros to store a real value.

See also: User Programming

Related: dqs Display group of special/automation parameters (M)

> Name storage for macros (P) n1, n2, n3

#### Resume acquisition stopped with sa command (C) ra

Description: Resumes an experiment acquisition that was stopped with the sa command. ra

is not permitted after any parameters have been brought into the stopped

experiment with the rt or rtp macros. The parameters dp and np may not be altered.

ra applies to the experiment that you are joined to at the time the command is entered. If experiment 1 has been previously stopped with sa, you must be joined to experiment 1 for ra to resume that acquisition. If you are in experiment 2, entering ra has no effect on experiment 1.

If an experiment has been stopped with sa, you can increase the number of transients nt and resume the acquisition with ra. You cannot, however, increase nt and enter ra if the experiment had completed in a normal fashion (i.e., it was not stopped with sa).

Note that the completion time and remaining time shown in the Acquisition Status window are not accurate after ra is executed.

See also: NMR Spectroscopy User Guide

Related: dp Double precision (P)

np Number of data points (P)
nt Number of transients (P)
rt Retrieve FID (M)

rtp Retrieve parameters (M)
sa Stop acquisition (C)

# rcvrwt Weighting for different receivers (P)

Applicability: Systems with multiple receivers.

Description: An array of real numbers giving weighting factors to use when combining

multiple receiver data. The i'th array element is used to weight data from the i'th receiver. Applying a weight factor is like increasing the gain of the receiver by the same factor (but the weights are specified as numerical factors rather than in

dB).

Examples: rcvrwt=10,12,8

#### react Recover from error conditions during werr processing (M)

Syntax: react<('wait')>

Description: When an acquisition error occurs, any action specified by the werr parameter

is executed. The react macro is a prototype for handling these errors. This macro can be invoked for error handling by setting werr='react'. The acqstatus parameter is provided so that react can determine which

specific error has occurred.

Arguments: 'wait' is a keyword for a special type of error handling during an automation

run. The react macro always uses the 'next' option when it calls the command au. Under certain conditions, it is also appropriate to use the 'wait' option. react checks to see if an argument was passed to it; that is, werr='werr(\'wait\')' to determine whether to use the 'wait'

option of au.

See also: NMR Spectroscopy User Guide

Related: acqstatus Acquisition status (P)

au Submit experiment to acquisition and process data (C)

werr Specify action when error occurs (C)

werr When error (P)

#### readallshims Read all shims from hardware (M)

Description: Reads all shims from the hardware and sets the values into the shim parameters

in the current parameter tree. The shims used depend on the shimset

configuration. For the shim set on the Ultra•nmr shim system,

readallshims is active only if hardware-to-software shim communication

is enabled.

See also: NMR Spectroscopy User Guide

Related: load Load status of displayed shims (P)

readhw Read current values of acquisition hardware (C)

setallshims Set all shims into hardware (M)

sethw Set values for hardware in acquisition system (C)

shimset Type of shim set (P)

Submit a setup experiment to acquisition (M)

# readbrutape Read Bruker data files from 9-track tape (U)

Syntax: (From UNIX) readbrutape file <number\_skipped>

Description: A shell script that reads one file from a Bruker tape into a UNIX file with the

name specified. Bruker tapes are likely to be made at 1600 bpi, although 1600

bpi is not a requirement.

Arguments: file is the name of the file read into UNIX. For identification, the .bru

extension is added to the file name.

number\_skipped is the number of files skipped and *includes* the header file (which is assumed to be the first file on the tape). The default is the script reads the first file after the header file. If number\_skipped equals 0, there is no

rewinding and the first file (or the next file) on the tape is read.

See also: NMR Spectroscopy User Guide

Related: convertbru Convert Bruker data (M,U)

# readfile Read the contents of a text file into two parameters (C)

Examples: readfile (path, par1, par2, <,cmpstr <,tree> >):num

Description: readfile reads the contents of a file and puts the contents into two supplied

parameters. The first word on each line in the file is placed in the first parameter. The remainder of the line is placed in the second parameter. An optional fourth argument specifies a string which is used to match the first word of the line. For

example, if the file contained:

H1pw 10 H1pwr 55 C13pw 14 C13pwr 50

and the comparison string was set to H1, only the lines starting with H1 would

be put into the parameters. Namely,  ${\tt H1pw}$  and  ${\tt H1pwr}$ .

Arguments: path is the path name of the file to read.

parl is the name of the parameter to hold the first word of the line.

par2 is the name of the parameter to hold the remainder of each line.

cmpstr is the optional comparison string for matching the first word.

tree is an optional parameter to select the tree for parl and par2. The possibilities are current, global, and local. Current is the default. Local is used if the parameters are \$macro parameters. If tree is used, the

cmpstr must also be supplied. If cmpstr is '', then it is ignored.

The parl and par2 parameters must already exist. If parl or par2 are defined as a real parameter, as opposed to a string parameter, then if the value does not have a number as the first word, a zero will be assigned.

num will be set to the number of items in the arrayed parameters parl and par2.

Lines that only contain white space are not added to the parameters. Lines that start with a # are not added to the parameters. Lines which start with a # can be used as comment lines. If a line only contains a single word, that word is put into the first parameter. The corresponding array element of the second parameter will be set to an empty string. The readfile will return the number of lines added to the parameters.

Examples: Examples using a prototype file containing the following:

```
# A readfile test case
# Proton values
H1pw 10
H1pwr 55
# Carbon values
C13pw 14
C13pwr 50
H1macro ft f full aph vsadj
End
readfile(systemdir+'/probes/testcase','attr','vals')
This sets the attr and vals parameters to arrays of six strings.
attr='H1pw','H1pwr','C13pw','C13pwr','H1macro','End'
vals='10','55','14','50','ft f full aph vsadj',''
readfile(systemdir+'/probes/
testcase','attr','vals','H1')
```

This sets the attr and vals parameters to arrays of three strings.

```
attr='H1pw','H1pwr','H1macro'
vals='10','55','ft f full aph vsadj'
```

The readfile command might be used in conjunction with the teststr command. The teststr command can be used to search an arrayed parameter to determine the index of a specified element.

For example,

```
teststr(attr,'H1pwr'):$e
vals[$e] will be the value of 'H1Pwr'
```

#### readhw Read current values of acquisition hardware (C)

```
Syntax: readhw("param1","param2",...)<:r1,r2,...>
    readhw("keword"):$res1,...
```

Description: Returns or displays the current values of the lock system parameters

lockpower, lockgain, lockphase, lock, temp, loc, and z0.

The values of the shims can also be obtained. The particular shims that can be read depends upon the type of shim hardware present in the system. See the description of shimset for a list of the shim names for each type of shim hardware.

Shim DACs read by readhw:

- Axial shim: z1, z2, z3, z4, z1c, z2c
- Non-axial shims: x1, y1, xz, yz, xy, x2y2, x3, y3
- Special Oxford magnets shims: z5, xz2, yz2, zx2y2, zxy

Arguments: param1, param2, ... parameter to read — maximum of 10 parameters.

r1, r2, . . . Vnmr variables hold the returned results no variables supplied — results are displayed in the text panel Keywords:

loc —sample changer location.

temp — returns the sample temperature, controller status, and set point. Results are displayed in the text panel if no variables are supplied

Returned value	Status
0	Regulation off
1	Regulated
2	Not regulated
3	No controller

status — returns the systems status as an integer. The returned values are:

Returned value	Status
10	IDLE
15	PARSE
16	PREP
17	SYNCED
20	ACQUIRE
25	PAD
30	VTWAIT
40	SPINWAIT
50	AGAIN
60	ALOCK
61	AFINDRES
62	APOWER
63	APHASE
70	SHIMMING
80	SMPCHANGE
81	RETRIEVSMP
82	LOADSMP
90	INTERACTIVE
100	TUNING
0	INACTIVE
Error messages	
-1	Available on spectrometer only (i.e. system = 'datastation')
-2	acquisition not active (acquisition communication programs are
	not running try running su acqproc).
-7	console powered down or not connected

Results are displayed in the text panel if no variables are supplied.

readhw cannot be used when an acquisition is in progress or when acqi is connected to the acquisition system.

Arguments:

param1, param2,... are the names of the parameters to be read. value1, value2,... are return variables to store the settings of the parameters specified. The default is to display the setting in the status window.

Examples: readhw('z1c','z2c','z1','z2')

readhw('z1c','z2c','z1','z2'):r1,r2,r3,r4

readhw('temp'):\$t sets \$t

See also: NMR Spectroscopy User Guide

Related: lockgain Lock gain (P)

lockphaseLock phase (P)lockpowerLock power (P)

readallshims Read all shims from hardware (M)

sethw Set values for hardware in the acquisition system (C)

**shimset** Type of shim set (P)

# readlk Read current lock level (C)

Syntax: readlk<:lock\_level>

Description: Returns the same information as would be displayed on the digital lock display

using the manual shimming window. readlk can be used in developing automatic shimming methods such as shimming via grid searching. It *cannot* be

used during acquisition or manual shimming.

Arguments: lock level returns the current lock level.

Examples: readlk

readlk: \$levell

See also: User Programming

Related: alock Automatic lock status (P)

# readparam Read one of more parameters from a file (C)

Syntax: readparam(file,parlist[,tree[,type]]) -

Description: The readparam command will read one or more parameters from a specified

file. The first argument is the name of the file. The second argument is a list of the names of the parameters to be read. It is a string parameter and the names can be separated either by a space or a comma. If a parameter in the list is not present in the file being read, no error is generated. The optional third argument is the tree into which the parameters are read. The variable trees are 'current', 'global', 'processed' and 'systemglobal'. The optional fourth argument controls the behavior of the readparam command. The options are

'read', 'replace', and 'add'. The default type is 'read'.

Examples: In order to specify the type, the tree must also be specified. The behaviors are

best illustrated with specific examples. Lets say that there is a temporary file containing only the parameters a and b. We are going to use the readparam command to read parameters into a current tree which contains the parameters a and c but does not contain the parameters b and d. This can be summarized as:

Parameters in mypar: a=1 b=2

Initial parameters in current tree: a=4 c=8 (b and d do not exist)

readparam(curexp+'/mypar','a b c d','current','read')

Parameter in a current tree is replaced with parameter from mypar. Parameter b in current tree is read in from mypar Parameter c in current tree is unaltered Parameter d in current tree still does not exist. Final parameters in current tree:

a=1 b=2 c=8 (d does not exist).

readparam(curexp+'/mypar','a b c d','current','replace')

Parameter in a current tree is replaced with parameter from mypar. Parameter b in current tree still does not exist. Parameter c in current tree is deleted.

Parameter d in current tree still does not exist. Final parameters in current tree: a=1 (b c and d do not exist).

readparam(curexp+'/mypar','a b c d','current','add')

Parameter in a current tree is unaltered. Parameter b in current tree is read in from mypar Parameter c in current tree is unaltered. Parameter d in current tree still does not exist. Final parameters in current tree: a=4 b=2 c=8 (d does not exist).

This command may be used to read temporary values which have been saved with the writeparam command.

# More Examples:

readparam(curexp+'/mypar','in')

reads the parameter in from the file mypar in the current experiment directory.

readparam(curexp+'/mypar','sw ct np','processed')

reads the parameters sw, ct, and np into the processed tree from the file mypar in the current experiment directory.

# readultra Read shim coil setting for Ultra•nmr shim system (M)

Applicability: Systems with the Ultra•nmr shim system.

Syntax: readultra<(file number)>

Description: Reads shim set files for a Ultra•nmr shim system from a Sun floppy disk into

VnmrJ. The floppy disk for Ultra•nmr contains up to 63 shim sets named

file1.dac to file63.dac.

Arguments: file number is the number of the shim set file, from 1 to 63. The default is

to read all of the shim set files.

Examples: readultra

readultra(6)

See also: NMR Spectroscopy User Guide

Related: **shimset** Type of shim set (P)

Save shim coil settings (C)

#### real Create a real variable without a value (C)

Syntax: real(variable)

Description: Creates a real variable without a value.

Arguments: variable is the name of the variable to be created.

Examples: real('realval1')

See also: User Programming

Related: create Create a new parameter in a parameter tree (C)

string Create a string variable (C)

# recon all Reconstruct images from 2D MRI fid data (C)

Applicability: Imaging Systems

Syntax: recon all(acqstring,<pc option>)

or

recon\_all(acqstring,<image directory>,<pc option>)

or

recon all

Description: Produces 2D images (in fdf format) from FID data acquired with most 2D

imaging sequence, including sems, gems, fsems, and epi.

Arguments:

acqstring: Set to 'acq' to indicate concurrent reconstruction; performs no initialization. Any other value can be used for retrospective reconstruction or the first pass through concurrent reconstruction (initialization is performed). pc option: Optional argument to specify phase correction method (see description of phase correction below).

image directory: Optional argument to specify the directory which will contain produced fdf files.

NB: for control of some features (see below), recon\_all accesses parameters in the PROCESSED tree. It is in the PROCESSED tree that variables should be created and/or modified for effectiveness with recon\_all.

Input/Output: recon\_all reads the FID file in the acqfil subdirectory of the current experiment, and creates fdf files that are written to the recon subdirectory of the current experiment when run in standalone mode, or to the study tree when run in study mode. If raw data output is selected (see option below), the resulting fdf files are written to the raw subdirectory of the current experiment.

#### Supported features include:

- Capable of running concurrently with acquisition (set acqstring to acq after first wnt; empty or dummy string initially).
- Disable image display (relevant parameter: recondisplay. Create in processed tree as a real variable and set it to 0)
- Display every N images (relevant parameter: recondisplay. Create in processed tree as a real variable and set it to N)
- DC removal (relevant parameter: dcrmv)
- Multi-shot/sorting (relevant parameters: petable, etl, and/or nseg)
- Multiple receiver data (magnitude sum) (relevant parameter: rcvrs)
- Weighting (through VnmrJ panel selections) (relevant parameter: ftproc)
- Zero filling (through VnmrJ panel selections) (relevant parameters: fn and/or fn1)
- Output magnitude and/or phase raw data components. (relevant (optional) parameter: raw. Create in processed tree as a string which can be set to 'm' (magnitude), 'p' (phase), or 'b' (both))
- Partial k-space conjugation. Relevant parameters are fract\_kx and fract\_ky, which denote the number of points/echoes acquired beyond the intended N/2. Example: nv=80, fract\_ky=16 results in the central 32 echoes used as a correction map prior to conjugate synthesis. Resulting image has 128 (2\*(80-16)) lines in the phase encoded direction.
- Phase correction. (relevant parameters: image, epi\_pc) Implemented for epi sequences. Phase of transformed imaging data (image=1) is corrected by phase of transformed reference data (image=0). Accepted values for pc option in command string or for the optional parameter epi\_pc are:
  - 1.POINTWISE (the default; direct use of the phase of profile)
  - 2.LINEAR (1st order fit of phase of profile)
  - 3.QUADRATIC (2nd order fit of phase of profile)
  - 4.CENTER\_PAIR (even/odd pair at center of echo train used for all even/odd echoes)

- 5.PAIRWISE (even/odd pair phase differences along echo train used) 6.FIRST PAIR (1st and 2nd echoes used for even/odd correction)
- Navigator Echo correction. Requires acquisition of echo train data (fsems, epi), some of which are not phase encoded. Adjusts phase of encoded echoes according to the phase of navigator echoes of the same echo train, relative to the first such navigator echo. Relevant parameters
- navigator (can be string set to 'y' or 'n', or array of integers giving navigator echo positions within the echo train (i.e., navigator=1, 2).)
- nav type (optional; string, set to 'off' to disable correction or 'POINTWISE' (default)).

# Order of operation:

# per echo in block

1.DC removal

2.echo reversal if necessary

3.raw data output if requested

4. windowing if necessary

5.read direction Fourier transform

6.phase correction if necessary

7.sorting if necessary

# per slice

1.navigator correction if necessary

2.windowing in phase direction if necessary

3.partial Fourier correction if necessary

4. phase direction Fourier transform

5.accumulation of multi-receiver data

6.write fdf output file

```
Examples: recon all('','/usr/home/myimages')
          recon all('','/usr/home/myimages','CENTERPAIR')
          recon all('ignorethis','LINEAR')
          recon all('acq')
```

#### record Record keyboard entries as a macro (M)

Syntax: record<(file|'off')>

Description:

Records keyboard entries and stores the entries as a MAGICAL macro in the user's maclib directory. To start recording keyboard entries, enter record. You are prompted for a macro name (you can also give the name as an argument to record). The command line prompt then becomes "Command?" to indicate that the record macro is active. Type the MAGICAL commands to be recorded on the keyboard. Function keys can be included by entering F1 to F8 for function keys 1 to 8, respectively. Enter off or record ('off') to finish the recording.

file is the name of the macro file in which the entries are saved. The default is that the user is prompted for a file name. If the macro file name already exists, the user is asked if the file should be overwritten.

'off' is a keyword to stop recording the entries.

Examples: record

record('mymacro') record('off')

See also: User Programming

redor1 Set up parameters for REDOR1 pulse seguence (M)

Applicability: Three-channel systems with a triple-tuned MAS solids probe.

Description: Sets up a parameter set, obtained with XPOLAR1, for REDOR (rotational echo

double-resonance) experiment.

See also: User Guide: Solid-State NMR

Related: xpolar1 Set up parameters for XPOLAR1 pulse sequence (M)

#### Restore 2D DOSY display from sub experiment (M) redosy

Restores the previous 2D DOSY display (if one exists) by recalling the data Description:

> stored by the dosy macro in the file subexp/dosy2Ddisplay in the current experiment. undosy and redosy enable easy switching between the 1D DOSY data (spectra as a function of gzlvl) and the 2D DOSY display

(signal as a function of frequency and diffusion coefficient).

NMR Spectroscopy User Guide See also:

Related: dosv Process DOSY experiments (M)

> Restore original 1D NMR data from subexperiment (M) undosy

#### reff1 Reference f1 Indirect Dimension from Observe Dimension (M)

Syntax: reff1<(refsource1)>

Description: Macros uses the ratio of the  $\Xi$  values for the relevant nuclei from refsource1

or the reference source specified to determine the reference frequency in the f1 indirect dimension directly from the reference frequency in the observe

dimension using the formula:

reffrq1 = (reffrq /  $\Xi$ [tn]) \*  $\Xi$ [nucf1] rfp1=0

rfl1 = sw1/2 - (frq[f1] - reffrq1)\*1e6

Ξ is the normalized frequency such that the <sup>1</sup>H signal from TMS is 100.00 MHz.

Referencing in the observe dimension using setref and this method is same as using setref1 (apart from minor round-off errors).

Referencing the observe dimension to an internal reference standard as proposed by IUPAC references all dimensions to that single reference signal and not the lock as with setref, setref1, and setref2.

Limitations: the macro works with data recalled from an archive or acquired on an other system provided the data was acquired using VNMR6.1C or newer.

Referencing is based on nuctables/nuctabrefBio if bioref='y' (global or local). Setting bioref='n' (global or local) or if the flag does not exist the standard IUPAC / organic chemistry referencing (nuctables / nuctabref) is used.

See /vnmr/nuctables/nuctabref.

Arguments: No argument — reference source is determined from refsource1. If the

relevant parameter is missing, the macro tries to determine the (indirect)

reference source from the axis parameter.

'sfrq', 'dfrq', 'dfrq2', 'dfrq3', or 'dfrq4' as a reference source

Examples: reff1 reff1('sfrq')

Related: reff2 Reference f2 Indirect Dimension from Observe Dimension (M)

setref Set Frequency Referencing for Proton Spectra (M)

setref1Set Frequency Referencing for f1 Evolution Dimension (M)setref2Set Frequency Referencing for f2 Evolution Dimension (M)mrefSet Referencing Based on Spectrum from the same sample (M)

bioref Flag for Bio-NMR Referencing (P)

# reff2 Reference f2 Indirect Dimension from Observe Dimension (M)

Syntax: reff2<(refsource2)>

Description: Macros uses the ratio of the  $\Xi$  values for the relevant nuclei from refsource1

or the reference source specified to determine the reference frequency in the f1 indirect dimension directly from the reference frequency in the observe dimension using the formula:

reffrq1 = (reffrq /  $\Xi$  [tn]) \*  $\Xi$  [nucf1] rfp1=0 rfl1 = sw1/2 - (frq[f1] - reffrq1)\*1e6

Ξ is the normalized frequency such that the <sup>1</sup>H signal from TMS is 100.00 MHz.

Referencing in the observe dimension using setref and this method is same as using setref1 (apart from minor round-off errors).

Referencing the observe dimension to an internal reference standard as proposed by IUPAC references all dimensions to that single reference signal and not the lock as with setref, setref, and setref2.

Limitations: the macro works with data recalled from an archive or acquired on an other system provided the data was acquired using VNMR6.1C or newer.

Referencing is based on nuctables/nuctabrefBio if bioref='y' (global or local). Setting bioref='n' (global or local) or if the flag does not exist the standard IUPAC / organic chemistry referencing (nuctables/nuctabref) is used.

See /vnmr/nuctables/nuctabref.

Arguments: No argument — reference source is determined from refsource2. If the

relevant parameter is missing, the macro tries to determine the (indirect)

reference source from the axis parameter.

'sfrq', 'dfrq', 'dfrq2', 'dfrq3', or 'dfrq4' as a reference source

Examples: reff2('dfrq3')

Related: reff1 Reference f2 Indirect Dimension from Observe Dimension (M)

setref Set Frequency Referencing for Proton Spectra (M)

setref1Set Frequency Referencing for f1 Evolution Dimension (M)setref2Set Frequency Referencing for f2 Evolution Dimension (M)mrefSet Referencing Based on Spectrum from the same sample (M)

bioref Flag for Bio-NMR Referencing (P)

# reffrq Reference frequency of reference line (P)

Description: Reference frequency, in MHz, of the reference line. This parameter is set by the

rl macro. By defining reffrq as the conversion factor between Hz and ppm

using the unit command, ppm calculations can be made.

If referencing is on (i.e., refpos is not set to 'n'), the go, ga, and au macros

calculate values of rfl and rfp based on reffrq and refpos. If

referencing is off, go, ga, and au set reffreq to sfrq.

See also: NMR Spectroscopy User Guide

Related: au Submit experiment to acquisition and process data (M)

Crl Clear reference line in directly detected dimension (M)

Submit experiment to acquisition and FT the result (M)

go Submit experiment to acquisition (M)

reffrq1 Ref. frequency of reference line in 1st indirect dimension (P) reffrq2 Ref. frequency of reference line in 2nd indirect dimension (P)

refpos Position of reference frequency (P)

Reference peak position in directly detected dimension (P)
Reference peak frequency in directly detected dimension (P)
rl Set reference line in directly detected dimension (M)

sfrq Transmitter frequency of observe nucleus (P)

unit Define conversion units (C)

#### reffrq1 Reference freq. of reference line in 1st indirect dimension (P)

Description: Reference frequency, in MHz, of the reference line in the first indirect

dimension of a nD experiment. This parameter should be used as the conversion

factor between hertz and ppm in the first indirect dimension.

See also: NMR Spectroscopy User Guide

Related: crl1 Clear reference line in 1st indirectly detected dimension (M)

reffrq Reference frequency of reference line (P)

refpos1 Position of reference frequency in 1st indirect dimension (P)

# reffrq2 Reference freq. of reference line in 2nd indirect dimension (P)

Description: Reference frequency, in MHz, of the reference line in the second indirect

dimension of a 2D experiment. This parameter should be used as the conversion

factor between hertz and ppm in the second indirect dimension.

See also: NMR Spectroscopy User Guide

Related: cr12 Clear reference line in 2nd indirectly detected dimension (M)

reffrg Reference frequency of reference line (P)

refpos2 Position of reference frequency in 2nd indirect dimension (P)

#### refpos Position of reference frequency (P)

Description: Position of reference frequency, set by the setref and rl macros. Setting

refpos='n' indicates that referencing has been turned off. The crl macro

turns referencing off.

Values: Because all spectra are (by definition) referenced to a frequency at 0 ppm,

refpos is either 0 or "not used".

See also: NMR Spectroscopy User Guide

Related:  $\mbox{crl}$  Clear reference line in directly detected dimension (M)

reffrg Reference frequency of reference line (P)

refpos1 Position of reference frequency in 1st indirect dimension (P)
refpos2 Position of reference frequency in 2nd indirect dimension (P)

Set reference line indirectly detected dimension (M)

setref Set frequency referencing (M)

# refpos1 Position of reference frequency in 1st indirect dimension (P)

Description: Position of reference frequency in the first indirect dimension of a nD

experiment, set by setref1 and rl1 macros. Setting refpos1='n' indicates that f1 referencing has been turned off. The crl1 macro turns f1

referencing off.

Values: Because all spectra are (by definition) referenced to a frequency at 0 ppm,

refpos1 is either 0 or "not used".

See also: NMR Spectroscopy User Guide

Related: crl1 Clear reference line in 1st indirectly detected dimension (M)

reffrq1 Ref. frequency of reference line in 1st indirect dimension (P)

refpos Position of reference frequency (P)

Set reference line in 1st indirect dimension (M)

setref1 Set frequency referencing for 1st indirectly detected dimension (M)

# refpos2 Position of reference frequency in 2nd indirect dimension (P)

Description: Position of reference frequency in the second indirect dimension of a 3D

experiment, set by setref2 and rl2 macros. Setting refpos2='n' indicates that f2 referencing has been turned off in 3D spectra. The crl2 macro

turns f2 referencing off.

Values: Because all spectra are (by definition) referenced to a frequency at 0 ppm,

refpos2 is either 0 or "not used".

See also: NMR Spectroscopy User Guide

Related: crl2 Clear reference line in 2nd indirectly detected dimension (M)

reffrq2 Ref. frequency of reference line in 2nd indirect dimension (P)

refpos Position of reference frequency (P)

Set reference line in 2nd indirect dimension (M)

setref2 Set frequency referencing for 2nd indirectly detected dimension (M)

# refsource1 Center frequency in 1st indirect dimension (P)

Description: Holds a parameter name to be used as the center frequency in the first indirect

dimension of 2D experiments. If refsource1 does not exist, the default is

'sfrq'.

For 2D experiments, the second dimension may be related to sfrq if it is a homonuclear experiment. The second dimension may also be related to dfrq if it is a heteronuclear experiment. refsourcel would then be set as refsourcel='sfrq' and refsourcel='dfrq', respectively.

See also: NMR Spectroscopy User Guide

Related: dfrq Transmitter frequency of first decoupler (P)

refsource2 Center frequency in 2nd indirect frequency (P)
sfrq Transmitter frequency of observe nucleus (P)

# refsource2 Center frequency in 2nd indirect dimension (P)

Description: Holds a parameter name to be used as the center frequency in the second indirect

dimension. refsource2 is analogous to refsource1

See also: NMR Spectroscopy User Guide

Related: refsource1 Center frequency in 1st indirect dimension (P)

# region Divide spectrum into regions (C)

Description: Breaks a spectrum up into regions containing peaks.

Arguments: tail length is the length from 0.0 to sw, in Hz, that is added to the start and

end of each calculated peak region; default value is sw/10. The default value is used if a negative number is entered for this argument. If the addition of these wings would cause overlap between adjacent regions, the wings are reduced until the regions no longer overlap.

relative\_number is a number that, in combination with other factors, governs the relative number of regions to be found. The default is 12, which is used if 0 is entered for this argument. relative\_number is used as part of a test to determine whether two spectral areas containing peaks are close enough together to be represented as a single region. There are no strict rules that associate the value of relative\_number to the total number of regions that will be found. In general, increasing this number decreases the number of regions that will be found and increases the size of an individual region. A value of 1 would give more regions; a value of 100 would give fewer regions.

threshold is a sensitivity factor used to decide if a data point is large enough, relative to the noise level, to qualify it as part of a peak. The default value is 0.6, which is used if 0 is entered for this argument. Smaller values of threshold make peak selection more sensitive; larger values make peak selection less sensitive.

number\_points governs the number of successive data points, normally from 7 to 40, that must qualify as part of a peak (see the description of threshold above) in order for that spectral area to be considered a real peak. The default value is a function of fn, sw, weighting functions, and other values. The default is used if 0 is entered for this argument. For carbon spectra with large spectral windows, experimental peaks often contain only one or two data points. Adjust number points to 1 or 2 in those cases.

tail\_size is a number that, in combination with relative\_number and other factors, governs whether two spectral areas that contain peaks are close enough together to be represented as a single region. The default value is used if 0 is entered for this argument.

number\_regions is the total number of regions determined by region.

Examples: region

region:\$1 region(50,0,1)

region(-1,0,0,2):r1

See also: NMR Spectroscopy User Guide

Related: fn Fourier number in directly detection dimension (P)

Spectral width in directly detected dimension (P)

# relayh Set up parameters for RELAYH pulse sequence (M)

Description: Sets up parameters for absolute-value COSY, or a single or double RELAY-

COSY pulse sequence.

See also: NMR Spectroscopy User Guide

Related: Cosy Set up parameters for COSY pulse sequence (M)

cosyps Set up parameters for phase-sensitive COSY (M)

Dqcosy Set up parameters for double quantum filtered COSY (M)

#### Move and/or rename a file (C) rename

Syntax: rename (from file, to file)

Description: Renames and/or moves a file or directory, rename is identical in function to

the command mv.

Arguments: from file is the name of the file to be moved to renamed.

to file is the name of the file after moving or renaming it. If the

from file argument has an extension such as .fid or .par, be sure the

to file argument has the same extension.

Examples: rename('/home/vnmr1/vnmrsys/seqlib/d2pul',

'/vnmr/seqlib/d2pul')

See also: NMR Spectroscopy User Guide

Related: Copy a file (C) сору

Copy a file (C) ср

Delete a file, parameter directory, or FID directory (C) delete

Move and/or rename a file (C)

Delete file (C)

#### Flag which enables/disables required parameters (P) regparcheck

Syntax: regparcheck= 'y' or 'n'

Description:

Description: The parameter regparcheck is a flag with the possible values of 'y' or 'n'. Only if

it is set to 'y' are actual parameters compared to the file. If it is set to 'n',

regpartest will always return 0.

'y' or 'n', indicating whether required parameters are to be checked. Values:

Related: callacq Utility macro to call Acq command (M)

> regparlist List of required parameters (P)

regparclear Clears the parameters in required parameter list (M) regpartest Tests whether required parameters are set (M)

#### reqparclear Clears the parameters in required parameter list (M)

Syntax: reqparclear

Description: Clears the parameters listed in regparlist. If for some reason regparlist has been

> destroyed, then this macro exits without a message. The parameter is cleared on the current tree, if it exists there, or on the global tree, if it exists there. If it exists in neither place, a message is printed and the routine moves on to the next

parameter in reqparlist.

The definition of "clear" is that real parameters are turned "off" and string

parameters are set to the empty string ".

There is a known issue with this macro, which due to its obscurity will remain as "user beware." The issue is that if a parameter of the same name exists in both the 'global' and 'current' trees, and if that parameter is part of regparlist, then it will be cleared in the 'current' tree but not in the global tree. Users should just

not be doing this.

Also note that while this macro checks for regparlist=", if it is an array and any element in the array is "then it assumes " is a parameter and reports a "does not exist" message.

Related: callacq Utility macro to call Acq command (M)

regparcheck Flag which enables/disables required parameters (P)

requarlist List of required parameters (P)

regpartest Tests whether required parameters are set (M)

# reqparlist List of required parameters (P)

Description: The parameter requarlist holds the parameter names. It is an array of strings. It

will not array the experiment.

Related: callacq Utility macro to call Acq command (M)

gettoken Utility macro to separate a string into tokens (M)

reqparcheck Flag which enables/disables required parameters (P)

reqparclear Clears the parameters in required parameter list (M)

reqpartest Tests whether required parameters are set (M)

# regpartest Tests whether required parameters are set (M)

Syntax: reqpartest<('showtext'|'showgui'<,callback\_string>)>

Description: If the parameter requarcheck='y', then this macro examines the list of

parameter names in requarlist and if all of them exist and are properly set, returns 0. Properly set is defined as a non-empty string for string parameters, or the active bit set (parameter is 'on') for real parameters.

the active off set (parameter is "off") for real parameters.

This macro also checks the string which is the concatenation of autoname + globalauto + sqname for any parameters in that string. Parameters in this string are delimited by \$.

For convenience, this macro will return different values depending on the specific non-true condition, as defined in the following table (X is "don't care").

All parameters exist	T	X	F	T	F
All parameters set	T	X	T	F	F
reqparcheck='y	Τ'	F	T	T	T
return value	0	-1	1	2	3

Also note that the non-existence of either reqparcheck or reqparlist is equivalent to reqparcheck not set to 'y'.

Parameters are checked in the current tree first for existence, and if that parameter exists there, then that tree is checked for whether it is set. If it does not exist in the current tree, then the global tree is checked. If and only if it exists in neither tree is it considered to not exist.

If the argument to this macro is 'showtext' then if one or more parameters do not exist or are not properly set, then they are listed on the alphanumeric (text) screen.

If the argument to this macro is 'showgui', then an entry popup is displayed for both creation (of non-existing parameters) and value entry. The return value is not affected by the fact that the values are now being entered - in other words, the return value is to be interpreted as 'did not exist' or 'was not set' prior to running the macro.

The comprehensive list to check is

reqparlist+autoname+globalauto+sqname. Some duplicates may occur, and this macro checks and eliminates duplicates.

The argument callback\_string is an optional argument that gets passed onto VnmrJ, and then gets passed back to vnmrbg when the required parameters entry popup closes. VnmrJ and vnmrbg are not otherwise synchronized, so this allows for re-entrance.

Arguments: 'showgui' | showtext'

'showgui' displays an entry popup in the required parameter is not set;

'show text' displays information about the required parameters in the text

window

callback\_string — optional callback to vnmrbg from VnmrJ (ignored

in 'showtext' option)

See also: VnmrJ User Programming

Related: callacq Utility macro to call Acq command (M)

reqparcheck Flag which enables/disables required parameters (P)
reqparclear Clears the parameters in required parameter list (M)

# resetf3 Reset parameters after a partial 3D Fourier transform (M)

Description: Restores the acquisition parameter sw, the processing parameter fn, and the

display parameters sp, wp, rfl, and rfp in the 3D parameter set, which are read into VnmrJ by either the select command or the dplane or dproj macros. These parameters were modified due to the selection of regional  $f_3$  processing (ptspec3d = 'ynn'). The original value for each of these parameters is stored in the parameter sv, where responsible represents sv, responsible wp,

rfl, or rfp (e.g., swsv).

If a 2D plane into VnmrJ is retrieved from a 3D transformed data set that was processed with regional  $f_3$  processing, resetf3 must be run before executing ft3d in that particular VnmrJ environment.

See also: NMR Spectroscopy User Guide

Related: dplane Display a 3D plane (M)

dproj Display a 3D plane projection (M)

fn Fourier number in directly detected dimension (P)

ft3d Perform a 3D Fourier transform (M)
ptspec3d Region-selective 3D processing (P)

rfl Ref. peak position in directly detected dimension (P)

Ref. peak frequency in directly detected dimension (P)

select Select a spectrum or 2D plane without displaying it (C)

sp Start of plot (P)

Spectral width in directly detected dimension (P)

vnmrjcmd() Commands to invoke the GUI popup(C)

Width of plot (P)

# resetplotter Reset plotter to system plotter (M)

Description: Command to reset a (temporarily chosen) plotter back to the system plotter

sysplotter. Command is called by all plotfile/plotpreview and plot/autoplot

buttons on plot panels.

# resolv Set resolution enhancement parameters (M)

Syntax: resolv<(a,b)>

Description: Calculates a default resolution enhancement function, setting up 1b and qf

based on the acquisition time at. "Zero-filling" is also accomplished, if

possible, by making  $fn \ge =2*np$ .

Arguments: a sets a value of  $\frac{1b}{b}$  using  $\frac{1b}{b} = -0.318/(a*sw)$ . The default for a is 0.1.

b sets a value of qf using qf=b\*sw. The default for b is 0.3.

Examples: resolv

resolv(.2,.4)

See also: NMR Spectroscopy User Guide

Related: at Acquisition time (P)

fn Fourier number in directly detected dimension (P)
gf Gaussian function in directly detected dimension (P)
Line broadening in directly detected dimension (P)

np Number of data points (P)

Spectral width in directly detected dimension (P)

# restorenuctable Calculate & store accurate nuctable for current system (M)

Syntax: restorenuctable

Description: The setref contribution is a generic nucleus table, /vnmr/nuctables/

nuctable, based on a standard proton frequency of  $1000.0\,\mathrm{MHz}$ . All standard nucleus tables in the /vnmr/nuctables are symbolic links pointing to a

generic table.

The restorenuctable is used to replace the standard links with specific links that to files containing proper and accurately calculated nucleus tables. Problems arising with custom macros and third party software that are not aware of the symbolic links pointing to a generic table can be fixed using this

macro.

Commands and utilities that do not scale nuctable entries to the actual proton frequency (as they should) will work better than with the standard tables.

Limitations: restorenuctable is not compatible with qtune and certain

commands in current software.

Examples: restorenuctable

Related: nuctable Display nucleus table for a given H1 frequency (M)

resume Resume paused acquisition queue (C)

Description: Enables continuing submitting experiments to the acquisition system. For

experiments initiated with the command <code>au('wait')</code>, the acquisition is paused during the time of data processing in order to prevent the acquisition from submitting new experiments that might be queued. resume then allows the data processing macro to initiate another acquisition with <code>au('next')</code>, which is then performed immediately instead of at the end of the queue.

See also: NMR Spectroscopy User Guide

Related: au Submit experiment to acquisition and process data (C)

return Terminate execution of a macro (C)

Syntax: return<(expression1,expression2,...)>

Description: Terminates the execution of a macro and optionally returns values to another

calling macro. This is usually used after testing some condition. return is

used only in macros and not entered from the keyboard.

Arguments: expression1, expression2, . . . are return values to another calling

macro.

See also: User Programming

Related: abort Terminate action of calling macro and all higher macros (C)

rev System software revision level (P)

Description: Stores a string identifying the VnmrJ software version for the system. This

parameter is not be entered by the user, but can be examined by entering rev?.

See also: VnmrJ Installation and Administration

Related: revdate System software preparation date (P)

revdate System software preparation date (P)

Description: Stores a string identifying the date the current VnmrJ software version was

prepared. This parameter is not be entered by the user, but can be examined by

entering revdate?.

See also: VnmrJ Installation and Administration

Related: rev System software revision level (P)

rfband RF band in use (P)

Description: Indicates which rf band of the amplifier is in use for each channel.

Values: A string, such as 'hlc', in which the first channel is determined by the first

character, the second channel is determined by the second character, and so

forth. The following values are available for each channel:

'h' indicates the high rf band is in use on the channel.

'l' indicates the low rf band is in use on the channel.

'c' indicates the system software will calculate whether to use the high band

or the low band for the channel.

See also: NMR Spectroscopy User Guide

rfblk Reverse FID block (C)

Syntax: rfblk(<src expno>,src blk no,dest expno,dest blk no)

Description: Reverses and copies data from a source FID block specified by src\_blk\_no

to a destination FID block specified by dest\_expno and dest\_blk\_no, using memory-mapped input and output. The file header determines the size

and type of data to reverse.

rfblk searches for the source and destination FID file in the directory \$vnmruser/expN/acqfil; N is the requested experiment number or the current experiment number. If the FID file is not open, rfblk opens the file, copies the data, and closes the file. If a number of blocks need to be copied, explicitly opening and closing the files with the commands mfopen and mfclose can significantly speed up the data reformatting process.

rfblk can also be used to append blocks of data to a FID file by specifying that the dest blk no is greater than the number of blocks in a file.

Be aware that rfblk can modify data returned to an experiment with the rt command. To avoid modification, enter the following sequence of commands before running rfblk:

```
cp(curexp+'/acqfil/fid',curexp+'/acqfil/fidtmp')
rm(curexp+'/acqfil/fid')
mv(curexp+'/acqfil/fidtmp',curexp+'/acqfil/fid')
```

Arguments:

src\_expno specifies the experiment number of the source FID file. The

default is the FID file of the current experiment.

src\_blk\_no specifies the source block of data to be copied. Block numbers run from 1 to the number of blocks in a file.

dest\_expno specifies the experiment number of the destination FID file.

dest\_blk\_no specifies the destination block to send the copied data.

Examples: rfblk(1,2,1) reverses and copies block 1 from the current experiment to

block 1 of experiment 2.

See also: User Programming

Related: mfblk Move FID block (C)

mfclose Memory map close FID file (C)

mfdata Move FID data (C)

mfopen Memory map open FID file (C)

mftrace Move FID trace (C)
rfdata Reverse FID data (C)
rftrace Reverse FID trace (C)

# rfchannel Independent control of rf channel selection (P)

Description:

Gives override capability over the selection of rf channels. rfchannel does not normally exist but can be created by a user with the command create('rfchannel', 'flag').

The control of each rf channel is built around a collection of parameters and pulse sequence statements. The frequency of channel 1 is set by sfrq and tof, its power by tpwr and tpwrf. The first decoupler uses the corresponding parameters dfrq, dof, dpwr, and dpwrf, respectively. Furthermore, the decoupler can have modulation modes specified with the parameters dmf, dm, dmm, dres, and dseq. The second decoupler has the same set of parameters as the first decoupler and they are distinguished by appending a 2 to each name. That is, the names aer dfrq2, dof2, dpwr2, dpwrf2, dmf2, dm2, dmm2, dres2, and dseq2. The third decoupler would use parameters with a 3 appended: dfrq3, dof3, dpwr3, dpwrf3, dmf3, dm3, dmm3, dres3, and dseq3. The rfchannel parameter provides a mechanism to override the default parameter usage.

Values: A string of one to four characters in which the position of each character identifies the rf channel controlled.

- The first character selects which rf channel (1 to 4) the parameters sfrq, tof, tpwr, etc. control. The first character also identifies the rf channel used as the receiver.
- The second character selects which rf channel (1 to 4) the parameters dfrq, dof, dpwr, etc. control.
- The third character maps the parameter set dfrq2, dof2, dpwr2, etc. to an rf channel (1 to 4).
- The fourth character maps tdfrq3, dof3, dpwr3, etc. to an rf channel (1 to 4).

For example, rfchannel='132' would exchange control of the second and third rf channels from the default parameter usage.

The number of characters in the rfchannel parameter must match the number of real rf channels (defined by the parameter numrfch) and each rf channel must be selected by the parameter.

Besides remapping the parameters to different rf channels, pulse sequence statements are also remapped. For example, if rfchannel='132', then statements decpulse, decshaped\_pulse, decoffset, decpower, decspinlock, and so on are applied on rf channel 3 and dec2pulse, dec2shaped pulse, and so on are applied on rf channel 2.

An obvious use for this remapping is on systems with the decoupler set to U+ H1 Only in the Spectrometer Configuration window. On these systems, if multinuclear pulses are needed and  $^1H$  needs to be observed, the parameter sets that assume a dual-broadband system can be used and the parameters remapped by setting rfchannel='21'. However, internal logic checks if the first decoupler is set to U+ H1 Only, tn is set to 'H1', and dn is not set to 'H1'. If these settings are the case, the parameter mapping for rf channels 1 and 2 is exchanged automatically.

See also: NMR Spectroscopy User Guide; User Programming

Related:  $\mbox{create}$  Create new parameter in parameter tree (C)

dfrq Transmitter frequency for first decoupler (P)

Decoupler mode for first decoupler (P)

dmf Decoupler modulation frequency for first decoupler (P)
dmm Decoupler modulation mode for first decoupler (P)

dn Nucleus for first decoupler (P)

dof Frequency offset for first decoupler (P)

dpwr Power level for first decoupler with linear amplifier (P)

dpwrf First decoupler fine power (P)

dres Tip-angle resolution for first decoupler (P)
dseq Decoupler sequence for first decoupler (P)

numrfch Number of rf channels (P)

Transmitter frequency for observe nucleus (P)

Nucleus for observe transmitter (P)

Frequency offset for observe transmitter (P)

Observe transmitter power level with linear amplifiers (P)

bserve transmitter fine power (P)

# rfchtype Type of rf channel (P)

Description: Configurati

Configuration parameter for type of rf on each channel. The value for a channel is set using the Type of RF label in the Spectrometer Configuration window. Pulse sequence programs check rfchtype to determine if indirect detection should be used for some experiments. Indirect detection occurs automatically if the decoupler is set to U+ H1 Only in the Spectrometer Configuration window, tn is set to 'H1', and dn is not set to 'H1'.

Values:

The values of rfchtype parallel the rftype values. The only distinction is that the setting for rftype is 'd' on the U+ Direct Synthesis and U+ H1 Only entries.

'U+ Direct Synthesis' is the setting for a system with direct synthesis (U+ Direct Synthesis in the Spectrometer Configuration window).

'U+ H1 Only' is a fixed-frequency proton system (U+ H1 Only in Spectrometer Configuration window).

'Deuterium Decoupler' is the setting for a system deuterium decoupler channel. 'Direct Synthesis' is the setting for direct synthesis (Direct Synthesis in the Spectrometer Configuration window).

'Broadband' is the setting for broadband (Broadband in the Spectrometer Configuration window).

'Fixed Frequency' is the setting for fixed frequency (Fixed Frequency in the Spectrometer Configuration window).

'SIS Modulator' is the setting for imaging modulator (SIS Modulator in the Spectrometer Configuration window).

VnmrJ Installation and Administration See also:

Related: Display current configuration and possibly change it (M) config

> Nucleus for first decoupler (P) dn Type of rf generation (P) rftype

Nucleus for observe transmitter (P) tn

#### rfdata Reverse FID data (C)

Syntax: rfdata(<src expno,>src blk no,src start loc, dest expno, dest blk no, dest start loc, num points)

Reverses and copies data specified by src start loc from a FID block Description:

specified by src blk no to a destination location specified by

dest expno, dest blk no, and dest start loc, using memorymapped input and output. The data point locations and the num points to be reversed are specified by data points corresponding to the np parameter, not bytes or complex points; however, when reversing the data, rfdata looks at the file header to determine the size and type of data to reverse.

rfdata searches for the source and destination FID file in the directory \$vnmruser/expN/acqfil; N is the requested experiment number or the current experiment number. If the FID file is not open, rfdata opens the file, copies the data, and closes the file. If a number of blocks need to be copied, explicitly opening and closing the files with the commands mfopen and mfclose can significantly speed up the data reformatting process.

Be aware that rfdata can modify data returned to an experiment with the rt command. To avoid modification, enter the following sequence of commands before running rfdata:

```
cp (curexp+'/acgfil/fid', curexp+'/acgfil/fidtmp')
rm(curexp+'/acgfil/fid')
mv (curexp+'/acqfil/fidtmp', curexp+'/acqfil/fid')
```

Arguments:

src expno specifies the experiment number of the source FID file. The default is the FID file of the current experiment.

src blk no specifies the source block of data to be copied. Block numbers run from 1 to the number of blocks in a file.

src start loc specifies the starting data location within the specified block to copy the data. Data locations start from 0 and are specified as data points corresponding to the np parameter.

dest expno specifies the experiment number of the destination FID file.

dest blk no specifies the destination block to send the copied data.

dest start loc specifies the starting data destination location within the

specified block to send the copied data.

Examples: rfdata(1,0,2,1,(nv-1)\*np,np) copies and reverses np points of

data from the starting location 0 of block 1 of the current experiment to the data

location (nv-1) \*np of block 1 of experiment 2.

See also: User Programming

Related: Move FID block (C) mfblk

> mfclose Memory map close FID file (C)

mfdata Move FID data (C)

mfopen Memory map open FID file (C)

mftrace Move FID trace (C)
rfblk Reverse FID block (C)
rftrace Reverse FID trace (C)

### rf1 Reference peak position in directly detected dimension (P)

Description: Actual position of the reference line in the spectrum (i.e., the distance from the

right edge of the spectrum to the reference line). If there is no reference line in the spectrum, rfl can be used to enter the frequency where the reference line

would appear if the line were present in the spectrum.

Values: Number, in Hz.

See also: NMR Spectroscopy User Guide

Related: rfl1 Reference peak position in 1st indirectly detected dimension (P)

Reference peak position in 2nd indirectly detected dimension (P)

Reference peak frequency in directly detected dimension (P)

# rf11 Reference peak position in 1st indirectly detected dimension (P)

Description: Analogous to the rfl parameter except that rfl1 applies to the first indirectly

detected dimension of a multidimensional data set. rfl1 can either be set manually or be adjusted automatically when the macro rl1 is used to assign a

reference line.

Values: Number, in Hz.

See also: NMR Spectroscopy User Guide

Related: rfl Reference peak position in directly detected dimension (P)

Reference peak position in 2nd indirectly detected dimension (P)

Reference peak frequency in 1st indirectly detected dimension (P)

### rf12 Reference peak position in 2nd indirectly detected dimension (P)

Description: Analogous to the rfl parameter except that rfl2 applies to the second

indirectly detected dimension of a multidimensional data set. rfl2 can either be set manually or be adjusted automatically when the macro  $\tt rl2$  is used to

assign a reference line.

Values: Number, in Hz.

See also: NMR Spectroscopy User Guide

Related: rfl Reference peak position in directly detected position (P)

rfl1 Reference peak position in 1st indirectly detected dimension (P)
rfp2 Reference peak frequency in 2nd indirectly detected dimension (P)

# rfp Reference peak frequency in directly detected dimension (P)

Description: Sets the frequency to be assigned to the reference line in the spectrum. rfp is

always stored in Hz, but can be entered in ppm by using the p suffix (e.g.,

rfp=2.1p).

Values: Number, in Hz.

See also: NMR Spectroscopy User Guide

Related: rfl Reference peak position in directly detected dimension (P)

rfp1 Ref. peak frequency in 1st indirectly detected dimension (P)

rfp2 Ref. peak frequency in 2nd indirectly detected dimension (P) rl Set reference line in directly detected dimension (M)

#### rfp1 Reference peak freq. in 1st indirectly detected dimension (P)

Description: Analogous to the rfp parameter except that rfp1 applies to the first indirectly

detected dimension of a multidimensional data set. rfp1 can either be set manually or be assigned a value when rll is called with an argument (e.g.,

rl1 (7.2p) assigns the value of 7.2 ppm to rfp1).

Values: Number, in Hz.

See also: NMR Spectroscopy User Guide

Related: Ref. peak position in 1st indirectly detected dimension (P)

> Ref. peak frequency in directly detected dimension (P) rfp rfp2 Ref. peak frequency in 2nd indirectly detected dimension (P) rl1 Set reference line in 1st indirectly detected dimension (M)

#### Reference peak freq. in 2nd indirectly detected dimension (P) rfp2

Description: Analogous to the rfp parameter except that rfp2 applies to the second

> indirectly detected dimension of a multidimensional data set. rfp2 can be set manually or be assigned a value when rl2 is called with an argument. For example, entering r12 (7.2p) assigns the value of 7.2 ppm to rfp2.

Values: Number, in Hz.

See also: NMR Spectroscopy User Guide

Related: Reference peak position in 2nd indirectly detected dimension (P)

> Reference peak frequency in directly detected dimension (P) rfp rfp1 Reference peak frequency in 1st indirectly detected dimension (P) Set reference line in 2nd indirectly detected dimension (C) r12

#### rftrace Reverse FID trace (C)

```
Syntax: rftrace(<src expno,src blk no,src trace no,
         dest expno, <dest blk no, dest trace no)
```

Description: Reverses and copies FID traces specified by src trace no from a FID block specified by src blk no to a destination location specified by dest expno, dest blk no, and dest trace no, using memorymapped input and output. The file header determines the size and type of data to be reversed.

> rftrace searches for the source and destination FID file in the directory \$vnmruser/expN/acqfil; N is the requested experiment number or the current experiment number. If the FID file is not open, rftrace opens the file, copies the data, and closes the file. If a number of blocks need to be copied, explicitly opening and closing the files with the commands mfopen and mfclose can significantly speed up the data reformatting process.

> You cannot use rftrace to append data to a FID file. Its purpose is for moving around data.

> Be aware that rftrace can modify data returned to an experiment with the rt command. To avoid modification, enter the following sequence of commands before running rftrace:

```
cp(curexp+'/acqfil/fid',curexp+'/acqfil/fidtmp')
rm(curexp+'/acqfil/fid')
mv (curexp+'/acqfil/fidtmp', curexp+'/acqfil/fid')
```

Arguments: src\_expno specifies the experiment number of the source FID file. The default

is the FID file of the current experiment.

src\_blk\_no specifies the source block of data to be copied. Block numbers run from 1 to the number of blocks in a file.

src\_trace\_no specifies the source trace of data within the specified block
to be copied. Trace numbers run from 1 to number of traces in a file.

dest\_expno specifies the experiment number of the destination FID file. dest\_blk no specifies the destination block to send the copied data.

src\_trace\_no specifies the destination trace of data within the specified block to be copied. Trace numbers run from 1 to the number of traces in a file.

Examples: rftrace(1,1,2,1,nv) copies and reverses trace 1 from block 1 of the

current experiment to trace nv of block 1 of experiment 2.

See also: User Programming

Related: mfblk Move FID block (C)

mfclose Memory map close FID file (C)

mfdata Move FID data (C)

mfopen Memory map open FID file (C)

mftrace Move FID trace (C)
rfblk Reverse FID block (C)
rfdata Reverse FID data (C)

# rftype Type of rf generation (P)

Description: Configuration parameter for type of rf generation on each rf channel. On other

systems, the value is set using the Type of RF label in the Spectrometer

Configuration window.

Values: The values of rftype parallel the rfchtype values. The setting for rftype

is 'd' on the entries U+ Direct Synthesis and U+ H1 Only.
'd' is the setting for a system with direct synthesis (U+ Direct Synthesis in the Spectrometer Configuration window) or a fixed-frequency proton system (U+

H1 Only in Spectrometer Configuration window).

'1' is the setting for a deuterium decoupler channel.

'c' is the setting for direct synthesis (Direct Synthesis in the Spectrometer Configuration window).

'b' is the setting for broadband (Broadband in the Spectrometer Configuration window).

'a' is the setting for fixed frequency (Fixed Frequency in the Spectrometer Configuration window).

'm' is the setting for imaging modulator (SIS Modulator in the Spectrometer Configuration window).

See also: VnmrJ Installation and Administration

Related: config Display current configuration and possibly change it (M)

rfchtype Type of rf channel (P)

# rfwg RF waveform generator (P)

Description: Configuration parameter for whether a waveform generator board is present or

not on the current rf channel. The value for each channel is set using the Waveform Generator label in the Spectrometer Configuration window.

Values: 'n' is setting for no waveform generator board on the channel (Not Present

choice in Spectrometer Configuration window).

'y' is setting for a waveform generation board on the channel (Present choice

in Spectrometer Configuration window).

See also: VnmrJ Installation and Administration

Related: config Display current configuration and possibly change it (M)

# right Set display limits to right half of screen (C)

Description: Sets the horizontal control parameters, sc and wc, to produce a display (and

subsequent plot) in the right portion of the screen (and page). For 2D data, space

is left for the scales.

See also: NMR Spectroscopy User Guide

Related: center Set display limits for center of screen (C)

full Set display limits for a full screen (C)

fullt Set display limits for full screen with room for traces (C)

left Set display limits for left half of screen (C)

Sc Start of chart (P)
WC Width of chart (P)

# rights Determine an operator's specified right (C)

Applicability: Walkup

Syntax: rights('right'<, 'errval'>)<:\$ret>

Description: The rights program queries the rights database to determine if the current

operator has the specified right. This command is used by the interface designer to determine if and how certain options are presented. An operator does not typically use this command. The system administrator sets (restricts) the rights for an operator using VnmrJ administrator interface. By default, the rights command grants any requested right. Rights requested that are not in the rights database are granted. Granting a right means that the rights program returns a 1

to the calling macro.

Arguments: right — a specific operator right, not case sensitive.

• 1 is returned by the command if the specified right is granted or the right is not in the rights data base

• 0 is default value returned by the command if the right is both in the database and the operator does not have the specified right.

errval — optional argument specifying return value if a right is both in the database and the operator does not have the specified right.

\$ret — variable holding the return value from the right command.

Examples: rights('prioritySample',-1):\$ok

Sets \$ok to -1 if the prioritySample right is not granted. A value of 1 is returned if the prioritySample is granted. Returning either a 0 or -1 if a right is not granted lets the interface designer choose to show or gray out a

control.

See also: VnmrJ Installation and Administration and VnmrJ Walkup manuals.

# rinput Input data for a regression analysis (M)

Description: Formats data for regression analysis and places the data into the file

regression.inp. The program is interactive. If a regression.inp already exists, rinput starts by asking if you want to overwrite the file. Type y and press the Return key. It then asks for an x-axis title and a y-axis title. Enter

the titles as asked (for no title, simply press Return). Next, rinput asks you to input the data in pairs. Separate each pair of values with a blank and press Return after the second value. At the end of the data set, press Return in response to the request for data. If you have another data set, type y and press Return to the question and then type in the data when it is asked for.

See also: NMR Spectroscopy User Guide; User Programming

Related: expl Display exponential or polynomial curves (C)

poly0 Find mean of data in the file regression.inp (C)

# rl Set reference line in directly detected dimension (M)

Syntax: rl<(frequency)>

Description: Sets the direct dimension reference line, taking into account any frequency

scaling with the scalesw parameter.

Arguments: frequency is a value, in Hz, to assign to the reference line. The default is the

cursor position cr. To enter the value in ppm, add a p suffix.

Examples: rl

rl(0) rl(7.2p)

See also: NMR Spectroscopy User Guide

Related: cr Current cursor position in directly detected dimension (P)

crl Clear ref. line in directly detected dimension (C)
reffrq Reference frequency of the reference line (P)
rl1 Set ref. line in 1st indirectly detected dimension (M)
rl2 Set ref. line in 2nd indirectly detected dimension (M)
scalesw Scale spectral width in directly detected dimension (P)

# rll Set reference line in 1st indirectly detected dimension (M)

Syntax: rl1<(frequency)>

Description: Sets the first indirect dimension reference line, taking into account any

frequency scaling with the scalesw1 parameter.

Arguments: frequency is a value, in Hz, to assign to the reference line. The default is the

cursor position cr1. You can enter the suffixes p, d, or k to mean ppm, decoupler ppm, and kilo, respectively. These suffixes are exactly equivalent to using \*sfrq, \*dfrq, and \*1000. Thus, if you are doing a 2D experiment in which the indirect axis is determined by the decoupler channel, you might enter,

for example, rl1(10d), which is equivalent to rl1(10\*dfrq).

Examples: rl1

rl1(0) rl1(7.2p)

See also: NMR Spectroscopy User Guide

Related: cr1 Cursor position in 1st indirectly detected dimension (P)

Clear ref. line in 1st indirectly detected dimension (M)

dfrq Transmitter frequency of first decoupler (P)

refpos2 Position of reference frequency in 2nd indirect dimension (P)

rl Set ref. line in directly detected dimension (M)
rl2 Set ref. line in 2nd indirectly detected dimension (M)
scalesw1 Scale spectral width in 1st indirectly detected dimension (P)

**sfrq** Transmitter frequency of observe nucleus (P)

# r12 Set reference line in 2nd indirectly detected dimension (M)

Syntax: rl2<(frequency)>

Description: Sets the second indirect dimension reference line, taking into account any

frequency scaling with the scalesw2 parameter.

Arguments: frequency is a value, in Hz, to assign to the reference line. The default is the

cursor position cr2. You can enter the suffixes p, d, or k to mean ppm, decoupler ppm, and kilo, respectively. These suffixes are exactly equivalent to using \*sfrq, \*dfrq, and \*1000. Because there is no suffix for the second decoupler (i.e., the third channel), to reference the third axis using rl2 you

might enter (e.g., rl2(45\*dfrq2)).

Examples: rl2

rl2(0) rl2(7.2p)

See also: NMR Spectroscopy User Guide

Related: cr2 Cursor position in 2nd indirectly detected dimension (P)

cr1Clear ref. line in directly detected dimension (C)cr11Clear ref. line in 1st indirectly detected dimension (C)cr12Clear ref. line in 2nd indirectly detected dimension (C)

dfrq Transmitter frequency of first decoupler (P)
dfrq2 Transmitter frequency of second decoupler (P)
rl Set ref. line in directly detected dimension (M)
rl1 Set ref. line in 1st indirectly detected dimension (M)

scalesw2 Scale spectral width in 2nd indirectly detected dimension (P)

sfrq Transmitter frequency of observe nucleus (P)

# rm Delete file (C)

Syntax: rm(file1<, file2,...>)

Description: Removes one or more files from the file system, functioning like the UNIX

command of the same name. Because it allows wildcard characters (\* and ?) in the command argument and recursive file deletion with the -r option, rm is very powerful. But it can be quite dangerous—without warning important files can be inadvertently deleted, even by experienced users. **Using rm to delete files in VnmrJ is not recommended**. The delete command is provided as

a safer alternative.

Arguments: file1, file2, ... are names of files to delete.

See also: NMR Spectroscopy User Guide

Related: delete Delete a file, parameter directory, or FID directory (C)

delexp Delete an experiment (C)

exists Determine if a parameter, file, or macro exists (C)

mv Move and/or rename a file (C)
rename Move and/or rename a file (C)

# rmdir Remove directory (C)

Syntax: rmdir(directory)

Description: Removes one or more empty directories (i.e., directories without files).

Arguments: directory is the name of the directory to be removed.

Examples: rmdir('/home/dan/temp')

See also: NMR Spectroscopy User Guide

Related: delete Delete a file, parameter directory, or FID directory (C)

dir List files in current directory (C)

lf List files in current directory (C)

List files in current directory (C)

mkdir Create new directory (C)

rmsAddData Add transformed data files with weighting (U)

Applicability: Systems with multiple receivers.

Description: This command is not normally executed directly by the user.

Roesy Convert the parameter to a ROESY experiment (M)

Description: Convert the parameter to a rotating frame Overhauser effect spectroscopy

(ROESY) experiment.

Roesy1d Convert the parameter set to a Roesy1d experiment (M)

Description: Convert the parameter set to a 1D rotating frame Overhauser effect

spectroscopy (Roesy1D) experiment.

See also: NMR Spectroscopy User Guide

Related: Proton Set up parameters for <sup>1</sup>H experiment (M).

selld Selective 1D protocols to set up (M).

rof1 Receiver gating time preceding pulse (P)

Description: Sets the period of time in most pulse sequences when the receiver is gated off

before each pulse. This allows the amplifier to fully turn on before the start of the pulse. Systems are configured with linear amplifiers that are normally "blanked" to give the best possible signal-to-noise (i.e., the amplifiers are turned off when the receiver is turned on). The  $^1H/^{19}F$  amplifiers have a short turn-on time, usually 1 to 5  $\mu$ s following the removal of blanking by turning the receiver off. The low-frequency amplifier modules have a longer turn-on time,

about 40 to 60 μs.

Values: Typically 2-5 microseconds. See also: *NMR Spectroscopy User Guide* 

Related: rof2 Receiver gating time following pulse (P)

rof2 Receiver gating time following pulse (P)

Description: Sets the time after the final pulse in each pulse sequence that the receiver is

gated off before acquisition begins. If "pulse breakthrough" effects are seen (a spike in the beginning of the FID), increasing rof2 can reduce or eliminate the

problem, particularly for low-frequency nuclei.

Values: Typically 10 microseconds.

See also: NMR Spectroscopy User Guide

Related: rof1 Receiver gating time preceding pulse (P)

setlp0 Set parameters for zero linear phase (M)

rof3 Receiver gating time following T/R switch (P)

Description: Sets the time when the receiver is gated on following the T/R switch during the

pulse. This allows for the elimination of pulse artifacts during the acquisition

period.

rotate Rotate 2D data (C)

Syntax: rotate<(number\_degrees)>

Description: Rotates a 2D spectrum. Both complex and hypercomplex 2D data will work.

Arguments: number degrees is the amount of counter-clockwise rotation, in degrees.

The default is 45.

See also: NMR Spectroscopy User Guide

Related: foldcc Fold INADEQUATE data about 2-quantum axis (C)

foldj Fold J-resolved 2D spectrum about fl=0 axis (C) foldt Fold COSY-like spectrum along diagonal axis (C)

rotorsync Rotor synchronization (P)

Applicability: Systems with the solids rotor synchronization module.

Description: Configuration parameter that identifies if the system has the optional solids

rotor synchronization module. The value of rotorsync is set using the Rotor Synchronization label in the Spectrometer Configuration window. Rotor synchronization requires either the Acquisition Controller board (Part No. 969204) or the Pulse Sequence Controller board (Part No. 992560) in the

system.

Values: 1 is setting that system has solids rotor synchronization (Present choice in the

Spectrometer Configuration window).

0 is setting that system does not have solid rotor synchronization (Not Present

choice in the Spectrometer Configuration window).

See also: VnmrJ Installation and Administration

Related: config Display current configuration and possibly change it (M)

rp Zero-order phase in directly detected dimension (P)

Description: Specifies the right phase-correction angles along the directly detected

dimension according to

absorption spectrum( $\omega$ ) =

real channel( $\omega$ ) \* sin  $\theta$  + imaginary channel( $\omega$ ) \* cos  $\theta$ 

where the phase angle  $\theta$  is a function of frequency:

 $\theta = rp + (\omega - \omega_0) * lp$ 

 $\omega_0$  is defined as the right end of the spectrum. This dimension is referred to as the  $f_2$  dimension in 2D data sets,  $f_3$  dimension in 3D data sets, and so on.

Values: -360 to +360, in degrees.

See also: NMR Spectroscopy User Guide

Related: aph Automatic phase adjustment of spectra (C)

aph0 Automatic phase of zero-order term (C)

First-order phase in directly detected dimension (P)

Zero-order phase in 1st indirectly detected dimension (P)

Zero-order phase in 2nd indirectly detected dimension (P)

setlp0 Set parameters for zero linear phase (M)

#### rp1 Zero-order phase in 1st indirectly detected dimension (P)

Description: Specifies the right phase parameter along the first indirectly detected

dimension, in degrees, for the f<sub>1</sub> dimension of a multidimensional data set

during the process of phase-sensitive 2D transformation.

See also: NMR Spectroscopy User Guide

Related: lp1 First-order phase in 1st indirectly detected dimension (P)

rp Zero-order phase in directly detected dimension (P)
rp2 Zero-order phase in 2nd indirectly detected dimension (P)

#### rp2 Zero-order phase in 2nd indirectly detected dimension (P)

Description: Controls the zero-order phase constant along the second indirectly detected

dimension during a ds, dconi, or equivalent display operation on the 2D data or a 1D trace therein. This dimension is often referred to as the f<sub>2</sub> dimension.

See also: NMR Spectroscopy User Guide

Related: dconi Interactive 2D contour display (C)

ds Display a spectrum (C)

First-order phase in 2nd indirectly detected dimension (P)

Zero order phase in directly detected dimension (P)

# rt Retrieve FIDs (M)

Syntax: rt<(file<,'nolog'>)>

Description: Retrieves FIDs from a file into the current experiment.

The rt macro does not copy the FID into the experiment. Instead, it links access to the original FID from the experiment. Most of the time, this behavior is desired, because the FID file is seldom changed. By making a link, disk space is also conserved. However, if the FID file in the experiment is written to, the data in the original file is also written to. It is best to make a copy of a FID file before altering it. The makefid command alters the FID file. The manual entry for makefid gives details on how to make a copy of the FID.

As another somewhat subtle point, because the FID in the experiment is a link to another .fid file, if that .fid file is removed, the link from the experiment may be gone. If you expect the FID in the experiment to be there, even if you delete the .fid file from where it was retrieved using rt, you should explicitly copy the file into the experiment.

Arguments: file is the name of the file that, with the suffix. fid added, contains the FIDs

to be retrieved. The default is that the system prompts for the name (in that case, the name can be given without single quotes). If file.fid does not exist and

file.par does, rt retrieves the parameters from file.par.

'nolog' is a keyword specifying that the log file is not to be retrieved.

Examples: rt

rt('/vnmr/fidlib/fid1d')

See also: NMR Spectroscopy User Guide

Related: fixpar Correct parameter characteristics in experiment (M)

makefid Make a FID element using numeric text input (C)

rtp Retrieve parameters (M)

rtv Retrieve individual parameters (C)
svf Save FIDs in current experiment (M)

Return Spinsight data into current experiment (C) rtcmx

Syntax: rtcmx<(file)>

Description: Retrieves Spinsight data into the current experiment.

Arguments: file is the name of the file. The default is that the macro prompts for the file

name.

Alternate: Load button in the files program.

Examples: rtcmx

rtcmx('redor.data')

See also: NMR Spectroscopy User Guide

Related: files Interactively handle files (C)

#### rtp Retrieve parameters (M)

Syntax: rtp<(file)>

Description: Retrieves parameters from a file into the current experiment.

Arguments: file is the name of the file that, with the suffix .par added, contains the

> parameters to be retrieved;. The default is that the system prompts for the name (in that case, the name can be given without single quotes). If file.par does not exist and file.fid does, rtp retrieves the parameters only from

file.fid.

Examples: rtp

rtp('/vnmr/stdpar/P31')

See also: NMR Spectroscopy User Guide

Related: fixpar Correct parameter characteristics in experiment (M)

> Retrieve FIDs (M) rt

rtv Retrieve individual parameters (C)

Save parameters from current experiment (M) svp

#### Retrieve shim coil settings (C) rts

Syntax: rts(file)<:status>

Description: Locates a preexisting file of shim settings and copies the settings into the current

parameter set of the current experiment and sets load='y' to facilitate subsequent loading of shims with su (or related commands or macros). If the shim file is not found, rts displays the file names it tried.

The rts command returns shims from a .fid file or a .par file, selecting the

shim parameters from the parameters stored there.

Arguments: file — the name of a file containing the shim coil settings to be retrieved. If

the file name is an absolute path, rts uses it with no modifications. Otherwise,

rts searches the applications directories.

status — the return variable with one of the following values after rts finishes searching for the shim coil settings file:

- 0 indicates that rts failed to find requested file.
- 1 indicates that rts found the requested file, either as an absolute path or in the shims directory of the first application directory.
- >=2 indicates that rts found the requested file in shims subdirectory of the second, third, or later application directory.

```
Examples: rts('acetone')
         rts('bb10mm'):r1
```

See also: NMR Spectroscopy User Guide

Related: load Load status of displayed shims (P)

Submit a setup experiment to acquisition (M)

Save shim coil settings (C)

#### rttmp Retrieve experiment data from experiment subfile (M)

Syntax: rttmp(file)

Description: Retrieves experiment data—parameters, FID, and transformed spectrum—from

the file specified in a subdirectory inside curexp+'/subexp'.

Arguments: file is the name of the subfile from which to retrieve the experiment data.

Examples: rttmp('H1')

rttmp('cosy')

See also: NMR Spectroscopy User Guide

Related: captain Copy experiment data into experiment subfile (M)

curexp Current experiment directory (P)

svtmp Move experiment data into experiment subfile (M)

# rtv Retrieve individual parameters (C)

Syntax: rtv<(file,par1<,index1<,par2,index2...>>)><:val>

rtv('parmaster','noabort','parameter'):\$pm

Description: Retrieves one or more parameters from a parameter file. The file might have

been made with svf or svp or sd commands, or it might be from another experiment. If no return argument is added, the parameters are copied into the experiment's current tree. If the parameter does not already exist in the current tree, it is created. If the returned parameter is an array, the entire array is

returned.

rtv returns values into the macro if a return argument is added. This form of rtv command, in which values are passed only to macro variables, avoids the

creation of additional parameters in the experiment's current tree.

Arguments: file — name of the directory or a parameter file. If the supplied value for

file is a directory (with or without the .fid or .par extension), the

parameters are retrieved from the procpar file in that directory. If the supplied value does not correspond to a directory but rather is a parameter file, that file is used. The default is that rtv prompts for a file name. In that case, the file

name can be given without single quotes.

par1, index1, par2, index2, . . . — name and array index of one or more parameters to be retrieved. The default for each array index argument is the first index. Including the array index for a parameter is only useful when

returning values to the macro through a return argument.

val — return argument for values to return to the macro. If the requested

parameter do not exist in the parameter file, rtv will abort.

noabort — keyword option must follow the 'parmaster' keyword and precede the parameter argument. This option applies to a single parameter. Command does not abort if the requested parameter does not exist.

parmaster — filename of the parameter set.

parameter — the parameter name.

Executing rtv without macro return values causes the fixpar macro run. The macro fixpar is not executed if return values are requested. rtv will prompt

for a file name if the command is executed without an argument. The filename

given in response to the prompt does not require single quotes.

In LC-NMR, rt will retrieve the lcdata (and drunlog) files if these files were saved along with the NMR data by using svf.

Examples: rtv

rtv('/vnmr/parlib/cosy.par','phase')

rtv('/vnmr/parlib/cosy.par','noabort','phase')

See also: NMR Spectroscopy User Guide and User Programming manuals

Related: rt Retrieve FIDs (M)

rtp Retrieve parameters (M)

Set first decoupler frequency to cursor position (M)

Save FIDs in current experiment (M)

Save parameters from current experiment (M)

#### rtx Retrieve parameters based on rtx rules (C)

Syntax: rtx(filename <,tree <, keyword1 <, keyword2 >>>)

Description: The rtx command retrieves parameters from filename, based on the setting of

the P\_LOCK protection bit and using the rules below.

Arguments: tree is 'current', 'processed', 'global', or 'systemglobal'.

keyword1 may be 'keep' or 'rt'. The default is 'keep'.

keyword2 may be 'clear' or 'noclear'. The default is 'clear'.

keyword2 determines if the P\_LOCK bit is cleared after rtx is executed.

Truth table for rtx.

Status of P_LOCK bit in current exp	Status of P_LOCK bit in filename	keyword1	result
on	on	keep or rt	do not rt
on	off	keep or rt	do not rt
off	on	keep or rt	do rt
off	off	keep	do not rt
off	off	rt	do rt
<no parameter=""></no>	on	keep or rt	do rt
<no parameter=""></no>	off	keep	do not rt
<no parameter=""></no>	off	rt	do rt

See also: NMR Spectroscopy User Guide

Related: execpars Set up the exec parameters (M)

rtp Retrieve parameters (M)

# S

s	Save display parameters as a set (M)
s(n)	Save display parameters (C)
s2pul	Set up parameters for standard two-pulse sequence (M)
sa	Stop acquisition (C)
sample	Submit change sample, Autoshim experiment to acquisition (M)
samplename	Sample name (P)
save	Save data (M)
savefile	Base file name for saving files (P)
saveglobal	Save selected parameters from global tree (P)
sb	Sinebell constant in directly detected dimension (P)
sb1	Sinebell constant in 1st indirectly detected dimension (P)
sb2	Sinebell constant in 2nd indirectly detected dimension (P)
sbs	Sinebell shift in directly detected dimension (P)
sbs1	Sinebell shift in 1st indirectly detected dimension (P)
sbs2	Sinebell shift in 2nd indirectly detected dimension (P)
sc	Start of chart (P)
sc2	Start of chart in second direction (P)
scalelimits	Set limits for scales in regression (M)
scalesw	Set scaling factor for multipulse experiments (M)
scalesw	Scale spectral width in directly detected dimension (P)
scalesw1	Set f <sub>1</sub> scaling factor for 2D multipulse experiments (M)
scalesw1	Scale spectral width in 1st indirectly detected dimension (P)
scalesw2	Scale spectral width in 2nd indirectly detected dimension (P)
sd	Set first decoupler frequency to cursor position (M)
sd2	Set second decoupler frequency to cursor position (M)
sd3	Set third decoupler frequency to cursor position (M)
sda	Set first decoupler frequency array (M)
sd2a	Set second decoupler frequency array (M)
sd3a	Set third decoupler frequency array (M)
sdp	Show diffusion projection (M)
sel1d	Apptype macro for Selective 1D experiments
select	Select spectrum, FID, trace, or 2D plane without display (C)
selex	Defines excitation band (M)
selexcit	Set up PFG selective excitation pulse sequence (M)
selexHT	Set up a selective Hadamard experiment (M)
send2vnmr	Send a command to VnmrJ (U)
seqfil	Pulse sequence name (P)
seqgen	Initiate compilation of user's pulse sequence (M,U)
serverport	Returns the VnmrJ network listening port value (C)
set2D	General setup for 2D experiments (M)
set2d	General setup for 2D experiments (M)
set3dproc	Set 3D processing (C)
setallshims	Set all shims into hardware (M)

setcolorSet colors for graphics window and for plotters (C)setdecparsSet decoupler parameter values from probe file (M)setdec2parsSet decoupler 2 parameter values from probe file (M)

setdgroupSet the Dgroup of a parameter in a tree (C)setenumeralSet values of a string parameter in a tree (C)setetherConnect or reconnect host computer to Ethernet (U)

**setexport** Set parameter bits for use with protocols (M)

setfrq Set frequency of rf channels (C)

setgaussSet a Gaussian fraction for lineshape (M)setgcalSet the gradient calibration constant (M)setgcoilAssign sysgcoil configuration parameter (M)setgridDivide graphics window into rows and columns (C)

setgroup Set group of a parameter in a tree (C)

sethtfrq1Set a Hadamard frequency list from a line list ((M)sethwSet values for hardware in acquisition system (C)

setint Set value of an integral (M)

setlimit Set limits of a parameter in a tree (C)

setlkSet up lock parameters (M)setlockfreqSet lock frequency (M)

**Set up linear prediction in the direct dimension (M)** 

setLP1Set F1 linear prediction parameters (M)setlp0Set parameters for zero linear phase (M)setnoetherDisconnect host computer from Ethernet (U)

setoffset Calculate offset frequency for given nucleus and ppm (M)

setparamsWrite parameter to current probe file (M)setpenSet maximum number of HP plotter pens (M)setplotdevReturn characteristics of a named plotter (C)

setpower Set power and pulsewidth for a given γB1 value (M)

setprotect Set protection mode of a parameter (C)

setrcSet receiver constants (M)setrefSet frequency referencing (M)

setref1Set freq. referencing for 1st indirectly detected dimension (M)setref2Set freq. referencing for 2nd indirect detected dimension (M)

 ${\color{red} \textbf{Set up a scout run (M)}}$ 

**setssfilter** Set sslsfrq to the frequencies of each suppressed solvents (M)

setsw Set spectral width (M)

setsw1Set spectral width in evolution dimension (M)setsw2Set spectral width in 2nd evolution dimension (M)

 $\begin{tabular}{ll} \textbf{Set selective frequency and width } (M) \end{tabular}$ 

setselinv Set up selective inversion (M)

settcldefault Select default display templates for pulse sequence (M)

Settune Opens the Auto Tune Setup dialog (M)

settype Change type of a parameter (C)

setupSet up parameters for basic experiments (M)setup\_dosySet up gradient levels for DOSY experiments (M)

**setvalue** Set value of any parameter in a tree (C)

write a wave definition string into Pbox.inp file (M)

setwin Activate selected window (C)

sf Start of FID (P)

Start of interferogram in 1st indirectly detected dimension (P)

Start of interferogram in 2nd indirectly detected dimension (P)

Transmitter frequency of observe nucleus (P)

sh2pu1Set up for a shaped observe excitation sequence (M)shdecSet up for shaped observe excitation sequence (M)

shell Start a UNIX shell (C)

shelli Start an interactive UNIX shell (C)

**shim** Submit an Autoshim experiment to acquisition (C)

**shimset** Type of shim set (P)

showconfigShow system configuration settings (M)showconsoleShow console configuration parameters (U)showfitDisplay numerical results of deconvolution (M)

showloginbox Shows operator login dialog (M)

shownumx x position counting from bottom left of every spectrum (P)
shownumy y position counting from bottom left of every spectrum (P)

showoriginalRestore first 2D spectrum in 3D DOSY experiment (M)showplotterShow list of currently defined plotters and printers (M)

showplotqDisplay plot jobs in plot queue (M)showprintqDisplay print jobs in print queue (M)

showprotuneguiShow the graphical interface while tuning (P)showrfmonShow RF Monitor Button in Hardware Bar (P)showstatDisplay information about status of acquisition (M,U)

Find sine value of an angle (C)

Find values for a sine window function (M)

sinebellSelect default parameters for sinebell weighting (M)sinesqFind values for a sine-squared window function (M)

Returns the number of elements in an arrayed parameter (O)

slfreqMeasured line frequencies (P)slwSpin simulation linewidth (P)

smaxfMaximum frequency of any transition (P)sminfMinimum frequency of any transition (P)

Sample Management System serial port connection (P)

sn Signal-to-noise ratio (P)

solppm Return ppm and peak width of solvent resonances (M)

solvent Lock solvent (P)

solvinfo Retrieve information from solvent table (C)

Sort real values of a parameter (M)

spStart of plot in directly detected dimension (P)sp1Start of plot in 1st indirectly detected dimension (P)sp2Start of plot in 2nd indirectly detected dimension (P)spaddAdd current spectrum to add/subtract experiment (C)

**spcfrq** Display frequencies of rf channels (M)

specdc3d 3D spectral dc correction (P)

Submit a spin setup experiment to acquisition (C)

spin Sample spin rate (P)

Run SpinCAD program (C) spincad

spingen Compile SpinCAD pulse sequence \*C)

spinll Set up a slfreq array (M)

spinner Open the Spinner Control window (C) spins Perform spin simulation calculation (C) Split difference between two cursors (M) split

spintype Spinner Type ((P)

spmax Take the maximum of two spectra (C)

spmin Take minimum of two spectra in add/subtract experiment (C)

Enter spin system (M) spsm

Subtract current spectrum from add/subtract experiment (C) spsub sqcosine Set up unshifted cosine-squared window function (M)

sqdir Study queue directory (P) End a study queue (M) sgend

sqexp Load experiment from protocol (M) Study queue file menu commands (M) sqfilemenu

Study queue mode (P) sqmode

Study queue parameter template (P) sqname

sqpars Create study queue parameters for imaging (M)

sqprotocol Macro to create protocols (M)

Reset study queue parameters for imaging (M) sqreset Return square root of a real number (O) sqrt

sqsavestudy Macro to save study parameters for imaging (M) Set up unshifted sinebell-squared window function (M) sqsinebell

srate Spinning rate for magic angle spinning (P) sread Read converted data into VnmrJ (C)

Calculate exact rof2 value for Cold Probes (M) srof2

Steady-state transients (P) SS

Set up solid-state echo pulse sequence (M) ssecho

ssecho1 Set up parameters for SSECHO1 pulse sequence (M) Full bandwidth of digital filter to yield a filtered FID (P) ssfilter sslsfrq Center of solvent-suppressed region of spectrum (P)

Number of coefficients in digital filter (P) ssntaps

ssorder Order of polynomial to fit digitally filtered FID (P)

stack Stacking mode for processing and plotting arrayed spectra (M)

stackmode Stacking control for processing arrayed 1D spectra (P)

Start a chained study queue (M) startq

status Display status of sample changer (C,U)

Apptype macro for Standard 1D experiments (M) std1d

Interactively create a method string for autoshimming (M) stdshm

Minimum intensity threshold (P) sth string Create a string variable (C)

strtext Starting point for LP data extension in np dimension (P) Starting point for LP data extension in ni dimension (P) strtext1

strtext2 Starting point for LP data extension in ni2 dimension (P) strtlp Starting point for LP calculation in np dimension (P)

Starting point for LP calculation in ni dimension (P) strtlp1

strtlp2 Starting point for LP calculation in ni2 dimension (P)

studyidStudy identification (P)studyparStudy parameters (P)studystatusStudy status (P)studytimeStudy time (P)

Submit a setup experiment to acquisition (M)

sub Subtract current FID from add/subtract experiment (C)

substr Select a substring from a string (C)

suselfrq Select peak, continue selective excitation experiment (M)

svdat Save data (C)

svfSave FIDs in current experiment (M)svfdfSave FID data in FDF format (M)svfdirDirectory for non-study data (P)

**svfname** Filename parameter template for non-study data ((P)

Svfname Create path for data storage (C)

Save parameters from current experiment (M)

Save shim coil settings (C)
Spin simulation vertical scale (P)

svtmp Move experiment data into experiment subfile (M)

sw Spectral width in directly detected dimension (P)

sw1 Spectral width in 1st indirectly detected dimension (P)

sw2 Spectral width in 2nd indirectly detected dimension (P)

sw3 Spectral width in 3rd indirectly detected dimension (P)

sysgcoil System gradient coil (P)

system type (P)

systemdir VnmrJ system directory (P)

## s Save display parameters as a set (M)

Syntax: (1) sset number

(2) s (set\_number)

Description: Saves a copy of the current values of all display parameters. The set is data-

independent because the parameters that govern a display (sp, wp, vs, etc.) are

saved but no data is saved.

Arguments: set number is number of the display parameter set to be saved.

Examples: s2

s(3)

See also: NMR Spectroscopy User Guide

Related: fr Full recall of display parameter set (M)

r Recall display parameter set (M)

#### s (n) Save display parameters (C)

Applicability: All

Syntax: s(n<,noupdate>)

Description: Saves a copy of the current values of all display parameters as display parameter

set n in the current experiment

noupdate as second argument prevents the automatic update of interactive

programs.

Arguments: n=1 to 9

Related: fr (n) Recall all the parameters of the specified display parameter set (C)

r (n) Recalls limited number of display parameters)

### s2pul Set up parameters for standard two-pulse sequence (M)

Description: Converts the current experiment to an experiment suitable for the standard two-

pulse sequence (S2PUL).

See also: NMR Spectroscopy User Guide

## sa Stop acquisition (C)

Syntax: sa<(option|number)>

Description: Stops an experiment that has been submitted to acquisition. If experiment is

active, it is stopped. Data is retained. sa applies to the experiment that you are joined to at the time the sa command is entered. Thus, if experiment 1 is active, you must be joined to experiment 1 for sa to stop that acquisition. If you are in

experiment 2, entering sa has no effect on experiment 1.

When experiments are queued, the behavior of sa is more complex. If an experiment is active in exp1 and queued in exp2, entering sa from exp1 stops that experiment and immediately begins acquisition on exp2. Entering sa from exp2, on the other hand, removes exp2 from the queue, without affecting the active experiment 1.

Entering sa from an experiment that is not active or queued has no effect.

Arguments: option is one of the following:

• 'eos', 'ct', 'scan' are keywords to stop at the next ct.

• 'eob', 'bs' are keywords to stop at the next block size.

• 'eof', 'nt', 'fid' are keywords to stop at the next complete FID.

• 'eoc', 'il' are keywords to stop at next complete il cycle (i.e., the latest block size that has been completed for all FIDs in interleave cycle.

number is an integer number to stop at the next ct, where the value of ct is a multiple of number. This is useful when you want to complete a phasecycle before stopping.

Examples: sa

sa('ct')

sa(4)

See also: NMR Spectroscopy User Guide

Related: bs Block size (P)

Ct Completed transients (P)

il Interleave arrayed and 2D experiments (P)

nt Number of transients (P)

Resume acquisition stopped with sa command (C)

# sample Submit change sample, Autoshim experiment to acquisition (M)

Applicability: Systems with a sample changer.

Description: Performs the combined operations change, spin, lock, and shim, making

it a convenient setup command for a new sample.

See also: NMR Spectroscopy User Guide

Related: au Submit experiment to acquisition and process data (C)

change Submit a change sample experiment to acquisition (M) ga Submit experiment to acquisition and FT the result (C)

go Submit experiment to acquisition (C)

lockSubmit an Autolock experiment to acquisition (C)shimSubmit an Autoshim experiment to acquisition (C)spinSubmit a spin setup experiment to acquisition (C)suSubmit a setup experiment to acquisition (M)

### samplename Sample name (P)

Description: Specifies the name of the sample. It is saved with a liquids study.

See also: NMR Spectroscopy User Guide, VnmrJ Walkup

Related: cqsavestudy Macro to save study queue parameters (M)

notebookNotebook name (P)pageName of page (P)studyparStudy parameters (P)

### save Save data (M)

Description: Macro to save data. In a study, it uses sqdir and autoname to construct the data

filename. If not in a study, it uses svfdir and svfname to construct the data

filename.

See also: NMR Spectroscopy User Guide, VnmrJ Walkup

Related: acquire Acquire data (M)

autoname Create path for data storage (C)
autoname Prefix for automation data file (P)
sqdir Study queue directory (P)
svfdir Directory for non-study data (P)
Svfname Create path for data storage (C)

svfname Filename parameter template for non-study data ((P)

#### savefile Base file name for saving files (P)

Applicability: Systems with LC-NMR accessory.

Description: Contains the base file name using the format savefile.001,

savefile.002, etc., to which a series of FIDs or data sets are saved. If

savefile does not exist, the parlc macro can create it.

See also: NMR Spectroscopy User Guide

Related: parlc Create LC-NMR parameters (M)

#### saveglobal Save selected parameters from global tree (P)

Description: Saves an array of parameter names from the global or systemglobal tree.

Whenever go is executed, the parameters listed are saved in the current tree with an underscore (\_) appended. These parameters are copied back into the global tree (without the underscore) whenever processing by wbs, wnt, wexp,

or werr occurs.

See also: NMR Spectroscopy User Guide

Related: go Submit experiment to acquisition (C)

Location of sample in tray (P)

#### Sinebell constant in directly detected dimension (P) sb

Description: Applies a sinebell constant along the directly detected dimension. This

dimension is often referred to as the f<sub>2</sub> dimension in 2D data sets, the f<sub>3</sub>

dimension in 3D data sets, etc.

A positive value applies a sinebell of the form  $\sin\left(\frac{t \cdot \pi}{2 \cdot sb}\right)$ A negative value applies a squared sinebell function of form  $\sin^2\left(\frac{t \cdot \pi}{2 \cdot sb}\right)$ 

sb is given in seconds. Typical value is sb='n'.

See also: NMR Spectroscopy User Guide

Related: Sinebell constant in 1st indirectly detected dimension (P

> sh2 Sinebell constant in 2nd indirectly detected dimension (P) shs Sinebell shift constant in directly detected dimension (P)

sine Find values for a sine window function (M)

sinebell Select default parameters for sinebell weighting (M) Find values for a sine squared window function (M) sinesa

#### sb1 Sinebell constant in 1st indirectly detected dimension (P)

Description: Applies a sinebell constant along the first indirectly detected dimension. This

> dimension is often referred to as the  $f_1$  dimension in multidimensional data sets. sb1 works analogously to the parameter sb. The "conventional" parameters, such as 1b and gf, operate on the detected FIDs, while this "2D" parameter is

used during processing of the interferograms.

Values:

A positive value applies a sinebell of the form  $\sin\left(\frac{t \cdot \pi}{2 \cdot sb1}\right)$ A negative value applies a squared sinebell function of form  $\sin^2\left(\frac{t \cdot \pi}{2 \cdot sb1}\right)$ 

sb1 is given in seconds. Typical value is sb1='n'.

See also: NMR Spectroscopy User Guide

Related: Sinebell constant in the directly detected dimension (P)

> sb2 Sinebell constant in 2nd indirectly detected dimension (P)

#### sb2 Sinebell constant in 2nd indirectly detected dimension (P)

Description: Applies a sinebell constant along the second indirectly detected dimension. This

> dimension is often referred to as the f<sub>2</sub> dimension in multidimensional data sets. sb2 works analogously to the parameter sb. The value of sb2 can be set with

wti on the 2D interferogram data.

Values: A positive value applies a sinebell of the form  $\sin\left(\frac{t \cdot \pi}{2 \cdot \text{sb2}}\right)$ A negative value applies a squared sinebell function of form  $\sin^2\left(\frac{t \cdot \pi}{2 \cdot \text{sb2}}\right)$ 

sb2 is given in seconds. Typical value is sb2='n'

See also: NMR Spectroscopy User Guide

Related: Sinebell constant in directly detected dimension (P)

> sh1 Sinebell constant in 1st indirectly detected dimension (P)

wti Interactive weighting (C)

#### Sinebell shift in directly detected dimension (P) sbs

Description: Working in combination with the parameter sb, sbs allows shifting the origin

of the sinebell function along the directly detected dimension. This dimension is often referred to as the f<sub>2</sub> dimension in 2D data sets, the f<sub>3</sub> dimension in 3D

The origin is shifted according to the formula  $sin\left(\frac{(t-sbs)\cdot\pi}{2\cdot sb}\right)$ Values:

The square of this function is applied if sb is negative. sbs is given in seconds.

The typical value is sbs='n'.

See also: NMR Spectroscopy User Guide

Related: Sinebell constant in directly detected dimension (P)

> Sinebell shift in 1st indirectly detected dimension (P) sbs1 sbs2 Sinebell shift in 2nd indirectly detected dimension (P)

Find values for a sine window function (M) sine

sinesq Find values for a sine squared window function (M)

#### sbs1 Sinebell shift in 1st indirectly detected dimension (P)

Description: Working in combination with the parameter sb1, sbs1 allows shifting the

> origin of the sinebell function along the first indirectly detected dimension. This dimension is often referred to as the  $f_1$  dimension in multidimensional data sets. sbs1 works analogously to parameter sbs. The "conventional" parameters, such as 1b and qf, operate on the detected FIDs, while this "2D" parameter is

used during processing of the interferograms.

The origin is shifted according to the form  $sin\left(\frac{(t-sbs1)\cdot \pi}{2\cdot sb1}\right)$ Values:

The square of this function is applied if sb1 is negative. sbs1 is given in

seconds. The typical value is sbs1='n'.

See also: NMR Spectroscopy User Guide

Related: sh1 Sinebell constant in 1st indirectly detected dimension (P)

> Sinebell shift constant in directly detected dimension (P) shs sb2 Sinebell constant in 2nd indirectly detected dimension (P)

#### Sinebell shift in 2nd indirectly detected dimension (P) sbs2

Description: Working in combination with the parameter sb2, sbs2 allows shifting the

origin of the sinebell function along the second indirectly detected dimension. This dimension is often referred to as the f<sub>2</sub> dimension in multidimensional data sets. sbs2 works analogously to parameter sbs. sbs2 can be set with wti on

the 2D interferogram data.

The origin is shifted according to the formula  $sin\left(\frac{(t-sbs2) \cdot \pi}{2 \cdot sb2}\right)$ 

The square of this function is applied if sb2 is negative. sbs2 is given in

seconds. The typical value is sbs2='n'.

See also: NMR Spectroscopy User Guide

Related: sbs Sinebell shift constant in directly detected dimension (P)

> sb2 Sinebell constant in 2nd indirectly detected dimension (P)

wti Interactive weighting (C)

#### Start of chart (P) sc

Description: Positions of the start of the plotting position (the "chart") with respect to the

right edge of the plotter.

Values: 0 to wcmax, in mm

See also: NMR Spectroscopy User Guide

Related: Start of chart in second direction (P) sc2

> Width of chart (P) WC

Maximum width of chart (P) wcmax

sc2 Start of chart in second direction (P)

Description: Controls the start of plotting position of the second axis (or y axis) of a 2D

contour plot. The parameter wc2 controls the width of the chart.

Values: 0 to wc2max, in mm.

See also: NMR Spectroscopy User Guide
Related: sc Start of chart (P

wc2 Width of chart in second direction (P)

wc2max Maximum width of chart in second direction (P)

scalelimits Set limits for scales in regression (M)

Syntax: scalelimits(x start,x end,y start,y end)

Description: Causes the command expl, which is used by regression to display data, to use

typed-in scale limits. The limits are retained as long as an expl display is

retained.

Arguments: x start,x end,y start,y end are x-axis and y-axis starting and ending

limits. The default is that scalelimits prompts for the limits.

See also: NMR Spectroscopy User Guide, User Programming

Related: autoscale Resume autoscaling after limits set by scalelimits (M)

expl Display exponential or polynomial curves (C)

scalesw Set scaling factor for multipulse experiments (M)

Description: Sets the spectral width scaling factor for the multipulse sequences set up by

macros br24 and mrev8. The value of the scaling factor is stored in the

parameter scalesw.

See also: User Guide: solid-State NMR

Related: br24 Set up BR24 multiple pulse experiment (M)

mrev8Set up MREV8 multiple pulse experiment (M)scaleswScale spectral width in directly detected dimension (P)scalesw1Set f1 scaling factor for 2D multipulse experiments (M)

scalesw Scale spectral width in directly detected dimension (P)

Description: Adjusts the frequency scale dimension used with the parameter sets in the

sequences set up by the br24, mrev8, ssecho, and xpolar1 macros. If scalesw is active, the labels for the frequency scales includes the letters sc in parentheses. A scaled frequency can be referenced using the rl macro.

Values: 'n', number greater than 0.0 See also: *User Guide: Solid-State NMR* 

Related: br24 Set up BR24 multiple pulse experiment (M)

mrev8 Set up MREV8 multiple pulse experiment (M)

rl Set reference line (M)

scalesw Set scaling factor for multipulse experiments (M)

scalesw1 Scale spectral width in 1st indirectly detected dimension (P)
scalesw2 Scale spectral width in 2nd indirectly detected dimension (P)

Set up solid-state echo pulse sequence (M)

xpolar1 Set up parameters for XPOLAR1 pulse sequence (M)

scalesw1 Set f<sub>1</sub> scaling factor for 2D multipulse experiments (M)

Description: Sets the f<sub>1</sub> spectral width scaling factor for the multipulse sequences set up by

the br24 and mrev8 macros. The value of the scaling factor is stored in the

parameter scalesw1.

See also: User Guide: Solid-State NMR

Related: br24 Set up BR-24 multiple pulse experiment (M)

mrev8 Set up MREV8 multiple pulse experiment (M)

Scale spectral width in 1st indirectly detected dimension (P)

scalesw1 Scale spectral width in 1st indirectly detected dimension (P)

Description: Analogous to the scalesw parameter except that scalesw1 applies to first

indirectly detected dimension of a multidimensional data set. A scaled frequency along this dimension can be referenced using the rll macro.

Values: 'n', number greater than 0.0 See also: *User Guide: Solid-State NMR* 

Related: rl1 Set reference line in 1st indirectly detected dimension (M)

scaleswScale spectral width in directly detected dimension (P)scalesw1Set f1 scaling factor for 2D multipulse experiments (M)scalesw2Scale spectral width in 2nd indirectly detected dimension (P)

scalesw2 Scale spectral width in 2nd indirectly detected dimension (P)

Description: Analogous to the scalesw parameter except scalesw2 applies to second

indirectly detected dimension of a multidimensional data set. A scaled frequency along this dimension can be referenced using the rl2 macro.

Values: 'n', number greater than 0.0 See also: *User Guide: Solid-State NMR* 

Related: rl2 Set reference line in 2nd indirectly detected dimension (M)

sd Set first decoupler frequency to cursor position (M)

Description: Sets the first decoupler frequency offset parameter dof to place the first

decoupler at the cursor position in the spectrum. This works only if the transmitter nucleus and first decoupler nucleus are the same (tn=dn).

See also: NMR Spectroscopy User Guide

Related: dof Frequency offset for first decoupler (P)

dn Nucleus of first decoupler (P)

Set second decoupler frequency to cursor position (M)
Set third decoupler frequency to cursor position (M)

sda Set first decoupler frequency array (M)
tn Nucleus for observe transmitter (P)

sd2 Set second decoupler frequency to cursor position (M)

Applicability: Systems with a second decoupler.

Description: Sets the second decouple frequency offset parameter dof2 to place the second

decoupler at the cursor position in the spectrum. This works only if the transmitter nucleus and second decoupler nucleus are the same (tn=dn2).

See also: NMR Spectroscopy User Guide

Related: dn2 Nucleus for second decoupler (P)

dof2 Frequency offset for second decoupler (P)

Set first decoupler frequency to cursor position (M)

sd2a Set second decoupler frequency array (M)
tn Nucleus for observe transmitter (P)

#### sd3 Set third decoupler frequency to cursor position (M)

Applicability: Systems with a third decoupler.

Description: Sets the third decoupler frequency offset parameter dof3 to place the third

decoupler at the cursor position in the spectrum. This works only if the transmitter nucleus and third decoupler nucleus are the same (tn=dn3).

See also: NMR Spectroscopy User Guide

Related: dn3 Nucleus for third decoupler (P)

dof3 Frequency offset for third decoupler (P)

Set first decoupler frequency to cursor position (M)

sd3a Set third decoupler frequency array (M)
tn Nucleus for observe transmitter (P)

#### sda Set first decoupler frequency array (M)

Description: Sets up an array of offset values for the first decoupler, using sd for the first

decoupler position and sda for subsequent positions. This works only if the transmitter nucleus and first decoupler nucleus are the same (tn=dn).

See also: NMR Spectroscopy User Guide

Related: dn Nucleus for first decoupler (P)

sd Set first decoupler frequency to cursor position (M)
sd2a Set frequency array for second decoupler (M)
sd3a Set frequency array for third decoupler (M)
tn Nucleus for observe transmitter (P)

sd2a Set second decoupler frequency array (M)

Applicability: Systems with a second decoupler.

Description: Sets up an array of offset values for the second decoupler, using sd2 for the first

position and sd2a for subsequent positions. This works only if the transmitter

nucleus and second decoupler nucleus are the same (tn=dn2).

See also: NMR Spectroscopy User Guide

Related: dn2 Nucleus for second decoupler (P)

Set second decoupler frequency to cursor position (M)

sda Set first decoupler frequency array (M)
tn Nucleus for observe transmitter (P)

sd3a Set third decoupler frequency array (M)

Applicability: Systems with a third decoupler.

Description: Sets up an array of offset values for the third decoupler, using sd3 for the first

position and sd3a for subsequent positions. This works only if the transmitter

nucleus and third decoupler nucleus are the same (tn=dn3).

See also: NMR Spectroscopy User Guide

Related: dn2 Nucleus for third decoupler (P)

Set third decoupler frequency to cursor position (M)

Set first decoupler frequency array (M)

Nucleus for observe transmitter (P)

## sdp Show diffusion projection (M)

Description: Displays projection onto diffusion axis using the dsp facility. Use with 2D or

3D DOSY data after DOSY analysis. The unit of the resulting axis is D  $(10^{-10} \text{ m}^2/\text{sec})$ . Because sdp overwrites the parameters in the current experiment, use it in only an experiment in which it is okay for existing data to be overwritten.

See also: NMR Spectroscopy User Guide

Related: dosy Process DOSY experiments (M)

### selld Apptype macro for Selective 1D experiments (M)

Description: Perform the actions for Selective 1D protocols to set up, process, and plot

experiments.

Examples: selld('setup') - execute selld experimental setup

sel1d('process') - execute sel1d processing

sel1d('plot') - execute sel1d plotting

Related: apptype Application type (p)

execpars Set up the exec parameters (M)

#### select Select spectrum, FID, trace, or 2D plane without display (C)

Syntax: (1) select<('next'|'prev'|selection)><:index>

(2) select<(<'f1f3'|'f2f3'|'f1f2'><,'proj'>

<,'next'|'prev'|plane>)><:index>

Description: Directs future actions to apply to a particular spectrum or FID in a 1D array, to

a trace in 2D (syntax 1), or to a particular 2D plane from a 3D data set (syntax 2). If select is called with no arguments, it returns the current index. When VnmrJ is first booted up, select is in 1D mode. select enters the 2D mode if any of the keywords 'f1f3', 'f2f3', 'f1f2', or 'proj' are present in the argument list. Entering the ds and jexp commands set select back in

the 1D mode.

Arguments: For 1D operations (syntax 1):

- 'next' is keyword to increment by 1 the 1D spectrum or trace index.
- 'prev' is keyword to decrement by 1 the 1D spectrum or trace index.
- selection is a number selecting a 1D spectrum, FID, or trace.
- index returns the number of the current 1D spectrum, FID, or trace.

For selecting various 2D planes of a 3D data set (syntax 2):

- 'f1f3', 'f2f3', and 'f1f2' are types of 2D planes. The parameters plane and index2 serve to indicate the exact 2D plane that is currently viewable by VnmrJ. Note that index2 cannot be entered from the keyboard (i.e., you cannot select a new 2D plane by changing the value of index2); you must use the select command instead.
- 'proj' is keyword to use the 2D projection whose plane type is determined by the parameter plane.

• 'next' is keyword to increment the parameter index2 to its next value
and sets up VnmrJ to be ready to display the 2D plane whose number is the
new index2 value.

• 'prev' performs analogously except that index2 is decremented.

• plane is a number selecting the plane.

• index returns the number of the current plane.

Examples: select('next')

select(2):r1
select('f1f3')

See also: NMR Spectroscopy User Guide, User Programming

Related: arraydim Dimension of experiment (P)

ds Display a spectrum (C)

index2 Projection or 3D plane index selected (P)

jexp Join existing experiment (C)

plane Currently displayed 3D plane type (P)

# selex Defines excitation band (M)

Syntax: selex<(sh<,pw<,st<,ph<,fla<,trev>>>>)>

Description: Defines the excitation band from the position of cursors in the graphics window

and reports them to user. It also sets r1 to excitation bandwidth and r2 to offset. selex is part of the Pbox software environment and uses the Pbox macros

pbox\_bw and putwave.

Arguments: sh is the name of a shape file.

pw is the pulsewidth, in sec.

st is the spin status: 0 for excitation, 0.5 for refocusing, or 1 for de-excitation.

ph is the phase (or phase cycle, see wavelib/supercycles).

fla is the flip angle.

trev is the time reversal. This argument can be used to cancel time reversal

introduced by setting the spin status (st) to 1 for de-excitation.

Examples: selex

selex('esnob',0.0,1,90.0)

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

#### selexcit Set up PFG selective excitation pulse sequence (M)

Applicability: Systems with a pulsed field gradient module.

Description: Prepares an experiment for PFG (pulsed field gradient) selective excitation,

with presaturation option.

See also: NMR Spectroscopy User Guide

### selexHT Set up a selective Hadamard experiment (M)

Description: Sets up parameters for a selective shaped pulse Hadamard-encoded test

experiment.

See also: NMR Spectroscopy User Guide

Related: htofs1 Hadamard offset in ni (P)

Fourier number in 1st indirectly detected dimension (P)

Number of increments in 1st indirectly detected dimension (P)

ft2d Fourier transform 2D data (C)

sethtfrq1 Set Hadamard frequency list from a line list (M)

send2vnmr Send a command to VnmrJ (U)

Syntax: send2Vnmr \$vnmruser/.talk command

Description: Sends a command from UNIX to VnmrJ using the port number stored in the

\$vnmruser/.talk file. This file is created when the macro listenon is

entered on the VnmrJ command line.

Arguments: command is any character string (commands, macros, or if statements)

normally typed into the VnmrJ command line.

Examples: send2Vnmr \$vnmruser/.talk dg

See also: *User Programming* 

Related: bootup Macro executed automatically when VnmrJ activated (M)

listenon Enable receipt of messages from send2Vnmr (M)

listenoff Disable receipt of messages from send2Vnmr (M)

segfil Pulse sequence name (P)

Description: Identifies the name of the pulse sequence to be used. The value of seqfil is

displayed on the top line of the screen after the "Seq:" label. Macros used to set up new pulse sequences, such as Dept and Apt, automatically change the

seqfil parameter.

See also: NMR Spectroscopy User Guide

Related: pslabel Pulse sequence label (P)

seggen Initiate compilation of user's pulse sequence (M,U)

Syntax: (From VnmrJ) seqgen(<-static,>file<.c>)

(From UNIX) seggen <-static> file<.c> <file1,...>

Description: Begins compilation of a user pulse sequence. When used from VnmrJ, the

macro seggen calls the UNIX shellscript seggen, which can also be called directly from UNIX, as shown above. The seggen shellscript then calls the compilation makefile seggenmake, located in the directory /vnmr/

acqbin.

The specified pulse sequence can be located in ~/vnmrsys/psglib or in / vnmr/psglib. If two files with the same name exist in these two directories, the local directory (~/vnmrsys/psglib) takes precedence. For sequences in /vnmr/psglib, seqgen first copies the file into the local directory ~/ vnmrsys/psglib and then compiles it there; the resulting executable is then placed in ~/vnmrsys/seqlib. A copy of the pulse sequence is also copied into the seqlib directory along with the executable. As it is running, seqgen reports where it found the specified sequence(s).

seggen uses library files (object modules) found in /vnmr/lib. If setuserpsg and psggen has been run, the library files in the local directory ~/vnmrsys/psg take precedence of those in /vnmr/lib.

Error messages are written into the file file.errors, where file is the name of the pulse sequence in psglib in which compilation is performed.

Note that seggen not only accepts file names with and without extensions, but also accepts files specified with wildcards and complex paths (seggen strips the directory part, and seggen /vnmr/psglib/apt will compile ~/vnmrsys/psglib/atp.c if it exists).

Arguments: -static is a keyword for seggen to use static rather than dynamic binding.

Static binding results in larger executables in seqlib (several hundred Kbytes), but these sequences execute slightly faster (i.e., the go command). While insignificant generally, faster execution is helpful in some special applications such as the Scout Scan<sup>TM</sup> mode of LC-NMR, where the time spent on the go command becomes critical. Static binding results in a fixed-size time gain, regardless of the number of increments; for large multidimensional

file is the file name of a standard two-pulse sequence.

experiments, the speed difference is not noticeable.

.c is the extension on the file name.

file1, file2, ... are the names of files containing more sequences.

Examples: (From VnmrJ) seggen('/vnmr/psglib/\*.c')

(From UNIX) seqgen /vnmr/psglib/\*.c (From UNIX) seqgen apt dept noesy (From UNIX) seqgen -static lc1d

See also: User Programming

## serverport Returns the VnmrJ network listening port value (C)

Applicability: VnmrJ

Syntax: serverport

Description: The serverport command returns the port number when VnmrJ opens a network

port (socket) for other programs to send it network messages. See the write ('net',...) command for an example on how to use this port

number.

Related: write Write formatted text to a device (C)

# set2D General setup for 2D experiments (M)

Syntax: set2D<(F2 dig res<,F1 dig res>)>

Description: Similar to set2d but does not execute par2d and does not make sw1, rf11,

and rfp1 decisions based on tn=dn condition.

Arguments: F2 dig res is the f<sub>2</sub> digital resolution desired, in Hz/pt. Default is 6.

F1 dig res is the f<sub>1</sub> digital resolution desired, in Hz/pt. Default is 12.

Related: rfl1 Reference peak position in 1st indirectly detected dimension (P)

Reference peak frequency in 1st indirectly detected dimension (P)

set2d General setup for 2D experiments (M)

sw1 Spectral width in 1st indirectly detected dimension (P)

# set2d General setup for 2D experiments (M)

Syntax: set2d(experiment<,F2\_dig\_res<,F1\_dig\_res>>)

Description: Runs the macro par2d to create new parameters needed for 2D experiments,

then selects starting values for a number of parameters. The set2d macro is

"internal" and not normally typed directly by the user.

Arguments: experiment is the name of a 2D experiment (e.g., 'noesy').

F2\_dig\_res is the  $f_2$  digital resolution desired, in Hz/pt. F1 dig res is the  $f_1$  digital resolution desired, in Hz/pt. Examples: set2d('cosyps')

set2d('hetcor',16)

set2d('het2dj',16,(2\*sw1)/fn1)

See also: NMR Spectroscopy User Guide

Related: par2d Create 2D acquisition parameters (M)

# set3dproc Set 3D processing (C)

yntax: set3dproc<(<'nocoef'><,directory>)>

Description: Creates the file procdat that contains binary 3D information used by ft3d

in processing the 3D FID data. It also creates the 3D parameter set procpar3d that is used by the select command to display the 2D planes from the 3D transformed data. set3dproc can only create the proper 3D coefficient file if the parameters phase and phase2 are used to generate States-Haberkorn (hypercomplex) or TPPI data along the t<sub>1</sub> and t<sub>2</sub> dimensions.

set3dproc creates the coefficient file for the following five values of array (where SH is States-Haberkorn):

• if array='' (null string), type of 3D data is  $TPPI(t_1) - TPPI(t_2)$ 

• if array='phase', type of 3D data is  $SH(t_1) - TPPI(t_2)$ 

• if array='phase2', type of 3D data is  $SH(t_2) - TPPI(t_1)$ 

• if array='phase2, phase', type of 3D data is  $SH(t_1) - SH(t_2)$ 

If  $\ensuremath{\mathsf{array}}$  is set to some other value,  $\ensuremath{\mathsf{set3dproc}}$  cannot create the 3D

coefficient file and an error is reported within VnmrJ.

Arguments: 'nocoef' is a keyword that the 3D coefficient file coef is not to be created.

directory is the name of the directory for procdat and procpar3d. The

default is the subdirectory info in the directory curexp.

Examples: set3dproc

set3dproc('nocoef','curexp/info3d')

See also: NMR Spectroscopy User Guide

Related: array Parameter order and precedence (P)

Perform a 3D Fourier transform (M,U)

phase Phase selection (P)

phase 2 Phase selection for 3D acquisition (P)

select Select a spectrum or 2D plane without displaying it (C) wftt3 Process f<sub>3</sub> dimension during 3D acquisition (M)

## setallshims Set all shims into hardware (M)

Description: Sets shims from the current parameter tree into hardware. setallshims is

equivalent to entering load='y'su but without setting all the hardware parameters normally set by su (temperature, decoupling, transmitter

initialization, etc.). The shims used depend on the shimset configuration. For the shim set on the Ultra•nmr shim system, setallshims is active only if

hardware-to-software shim communication is enabled.

See also: NMR Spectroscopy User Guide

Related: load Load status of displayed shims (P)

readallshims Read all shims from hardware (M)

readhw Read current values of acquisition hardware (C)
sethw Set values for hardware in acquisition system (C)

shimset Type of shim set (P)

Submit a setup experiment to acquisition (M) S11

#### setcolor Set colors for graphics window and for plotters (C)

Syntax: (1) setcolor('pcl',item index,'color') (2) setcolor('hpgl',item index,'color')

(3) setcolor('pen',pen number,'color')

(4) setcolor('graphics',item index,red,green,blue)

(5) setcolor('ps',item index,red,green,blue) (6) setcolor('plotter', black plane, color planes)

Description: Sets colors used on the graphics window and on plotters. This command is a

utility program used by the color macro and other macros. It is not expected

that setcolor would be entered directly from the input window.

Arguments: 'pcl' is a keyword to set colors on a plotter device that uses the PCL language. PCL plotters are the laser type of plotter.

> 'hpgl' is a keyword to set colors on a plotter device that uses the HPGL language. HPGL plotters are the pen type of plotter.

'pen' is a keyword that next two arguments set the color for a physical pen on a plotter device that uses the HPGL language.

'graphics' is a keyword to set colors on the graphics window.

'ps' is a keyword to set colors on a plotter using the PostScript language.

red, green, blue are three integers between 0 and 255 that set the amount of red, green, and blue color on the graphics window or PostScript plotter.

'plotter' is a keyword that the next two arguments set the black mode and number of colors available for a plotter device.

item index is an index number from the following list that represents a specific drawing item.

8 background of images

9 real channel of an FID

10 imaginary channel of an FID

spectrum 11

12 integral

parameters 13

14 scale

threshold line (graphics device only) 15

second spectrum or FID in addi (graphics device only) 16

result spectrum or FID in addi (graphics device only) 17

1 8 cursors (graphics device only)

19 foreground of images

20 background color of graphics window (graphics device only)

contour 0 to contour 15 of absolute value 2D display 20-35

36-42 contours -7 to -1 of phased 2D display

contours 1 to 7 of phased 2D display 44-50

pen\_number is an integer from 1 to 8 that specifies the physical pen used. color is a string for the color set for the device: 'red', 'green', 'blue', 'cyan', 'magenta', 'yellow', 'white', or 'black'.

black plane is 1 or 0, specifying whether the plotter has a separate black mode. Because all currently supported plotters have this feature, the value is usually 1.

color planes specifies how many colors are available. Use 3 for color

plotters and 0 for black and white plotters.

Examples: setcolor('pcl',11,'green')

setcolor('hpgl',11,'red') setcolor('pen',2,'red')

setcolor('graphics',11,255,0,0) setcolor('ps',11,255,255,0) setcolor('plotter',1,0)

See also: NMR Spectroscopy User Guide

Related: addi Start interactive add/subtract mode (C)

> color Select plotting colors from a graphical interface (M)

Set decoupler parameter values from probe file (M) setdecpars

Syntax: setdecpars

Description: Reads from the probe file pwxlvl, pwx, pplvl, pp, dpwr, dmf, dmm, dres,

and dseq values, if they exist, and updates the current experiment parameters.

Related: setdec2pars Set decoupler 2 parameter values from probe file (M)

Set decoupler 2 parameter values from probe file (M) setdec2pars

> Syntax: setdec2pars

Description: Reads from the probe file pwx2lvl, pwx2, dpwr2, dmf2, dmm2, dres2,

and dseg2 values, if they exist, and updates the current experiment parameters.

Related: setdecpars Set decoupler parameter values from probe file (M)

setdgroup Set the Dgroup of a parameter in a tree (C)

setdgroup(parameter,dgroup<,tree>)

Description: Sets the Dgroup of a parameter in a tree. The application determines the usage

of setdgroup. Only Tcl-dg currently uses this feature.

Arguments: parameter is the name of the parameter.

dgroup is an integer.

tree is 'current', 'global', 'processed', or 'systemglobal'. The default is 'current'. Refer to the description of the create command

for more information on types of trees.

Examples: setdgroup('a',1)

setdgroup('b',3,'global')

See also: User Programming

Related: create Create new parameter in a parameter tree (C)

Set values of a string parameter in a tree (C) setenumeral

> Syntax: setenumeral(parameter, N, enum1, enum2, . . ., enumN<, tree>)

Description: Sets the possible values of a string parameter in a parameter tree. To remove

enumerated values from a parameter, set argument N to 0 (see example below).

Arguments: parameter is the name of the parameter.

N is the number of enumeral values to be assigned to parameter (or removed

from parameter if N is set to 0).

enum1 to enumN are the possible string values of the parameter.

tree is 'current', 'global', 'processed', or 'systemglobal'. The default is 'current'. Refer to the description of the create command

for more information on types of trees.

Examples: setenumeral('size',0)

setenumeral('size',2,'large','small')

setenumeral('user',3,'user','superuser','master',

'global')

See also: User Programming

Related: create Create new parameter in a parameter tree (C)

#### setether Connect or reconnect host computer to Ethernet (U)

Description: Connects or reconnects the host computer to the Ethernet network. Only root

can execute this shellscript properly. If the system is already connected to the

Ethernet network, setether does nothing.

On systems running Solaris, setether undoes the work of setnoether. You cannot use setether unless you previously entered the setnoether

command. setether restores the files hostname.le0,

defaultdomain, and defaultrouter so that Ethernet is activated on the

host computer when UNIX is rebooted.

See also: VnmrJ Installation and Administration

Related: setnoether Disconnect host computer from Ethernet (U)

### setexport Set parameter bits for use with protocols (M)

Description: Set the parameter protection bits for use with the rtx command. Usually called

by other macros, and not used from the command line.

Related: rtx

cqprotocol Create study queue parameters for liquids (M)

## setfrq Set frequency of rf channels (C)

Syntax: setfrq<(channel)><('nucleus')>

Description: Calculates frequencies based on the nucleus (tn, dn, dn2, etc.), referencing

(lockfreq), solvent, and the offset parameter (tof, dof, etc.). The result of the calculation is stored in parameters sfrq, dfrq, dfrq2, etc. The

parameters are rounded to the resolution of the channel—either 0.1 or 100 Hz.

The setfrq command should never need to be entered from the keyboard. It is called automatically when the appropriate parameters are changed or a parameter set is returned. If a parameter is entered that affects a single frequency, setfrq is called from an internal underscore macro (e.g., \_tn, \_tof, \_dn, \_dof) to recalculate the frequency for that channel. Likewise, if a parameter is entered that affects all frequencies, setfrq is called from an internal underscore macro (e.g., \_solvent, \_lockfreq) to recalculate the

frequencies.

Arguments: channel is a single integer specifying the rf channel to be set. The default is

to calculate the frequencies for all rf channels.

nucleus displays or returns the frequency of the supplied nucleus. Channel 1 is assumed for rounding information and an offset (e.g., tof or dof) is not

added to the result.

Examples: setfrq

setfrq(2)

setfrq('P31'):freq

See also: NMR Spectroscopy User Guide

Related: spcfrq Display frequencies of rf channels (M)

#### setgauss Set a Gaussian fraction for lineshape (M)

Syntax: (1) setgauss(fraction)

(2) setgauss (fraction\*)

Description: Modifies the output of a deconvolution using pure Lorentzian lineshape

(fitspec.outpar) and makes it the input for a subsequent analysis (fitspec.inpar), after first modifying the Gaussian fraction. To allow this

fraction to vary, use syntax 1; to fix the fraction, use syntax 2.

Arguments: fraction is the Gaussian fraction of the lineshape, a number from 0 to 1. To

fix the fraction (syntax 2), suffix the value with an asterisk (\*) and enclose the

value in single quotes (see the second example below).

Examples: setgauss(0.4)

setgauss('1.0\*')

See also: NMR Spectroscopy User Guide

Related: fitspec Perform spectrum deconvolution (C)

## setgcal Set the gradient calibration constant (M)

Applicability: Systems with pulsed field gradients (PFG) or imaging capabilities.

Description: Determines the gradient calibration constant gcal by using a proton phantom

of known dimensions. setgcal requests the linear dimension of the phantom in the readout direction. It uses the value entered, together with cursor separation of this dimension from the image profile and the strength of the readout gradient gzlvll if pulsed field gradients, to calculate gcal in units of gauss/cm-DAC units. You are then prompted whether this value should be entered. If you answer yes, it is stored as a system constant in the your global

file.

Note that a particular value of gcal is closely related to the current eddy current compensation settings. If these settings are changed (e.g., reading in a

new curecc file), a different value of gcal should be expected.

Before running setgcal, use the pulse sequence set up by profile to acquire a signal from a known sized object while the gradient is on.

See also: Pulsed Field Gradient Modules Installation; VnmrJ Imaging NMR

Related: gcal Gradient calibration constant (P)

profile Set up pulse sequence for gradient calibration (M)

# setgcoil Assign sysgcoil configuration parameter (M)

Syntax: setgcoil<(file)>

Description: Allows users to change the configured gcoil for the system. setgcoil

updates the systemglobal parameter sysgcoil to the named table and updates the assignment value of the parameter gcoil in the named table. The directory

\$vnmrsystem/imaging/gradtables must have write permission for all users for the macro to be effective. This table now exists in the system local

/var/vnmr/gradtables directory, with a soft link from \$vnmrsystem/imaging/gradtables to that directory.

Arguments: file is the any legal file name defined for the parameter gcoil.

See also: VnmrJ Imaging NMR

Related: config Display current configuration and possible change it (M)

gcoil Read data from gradient calibration tables (P)
sysgcoil System value for gcoil parameter (P)

## setgrid Divide graphics window into rows and columns (C)

Syntax: setgrid(row<,column>)

Description: Divides graphics window into an array of rows and columns (or window panes).

Only one pane is active at a time. An individual pane can be activated by double-clicking in it with the left mouse button or by entering setwin in the

input window.

Arguments: row is the number of rows (maximum is 3) in the graphics window. If 0 is

entered, the number of rows remains the same; e.g., in setgrid(0,2), the number of rows is unchanged and two columns are created in each row.

column is the number of columns (maximum is 3) in the graphics window.

Examples: setgrid(3)

setgrid(3,3)
setgrid(0,2)

See also: NMR Spectroscopy User Guide

Related: curwin Current window (P)

fontselectOpen FontSelect window (C)jwinActivate current window (M)mapwinList of experiment numbers (P)setwinActivate selected window (C)

## setgroup Set group of a parameter in a tree (C)

Syntax: setgroup(parameter,group<,tree>)

Description: Sets the group of a parameter in a tree.

Arguments: parameter is the name of the parameter.

group is one of the following keywords: 'all', 'sample',
'acquisition', 'processing', 'display', or 'spin'.

tree is one of the keywords 'current', 'global', or 'processed'. The default is 'current'. See the  $\tt create$  command for information on the

types of trees.

Examples: setgroup('a','sample')

setgroup('b','all','global')

See also: User Programming

Related: create Create new parameter in a parameter tree (C)

destroy Destroy a parameter (C)

destroygroupDestroy parameters of a group in a tree (C)displayDisplay parameters and their attributes (C)

groupcopy Copy parameters of group from one tree to another (C)

paramvi Edit a parameter and its attributes using vi text editor (M)

setlimitSet limits of a parameter in a tree (C)setprotectSet protection mode of a parameter (C)

# sethtfrq1 Set a Hadamard frequency list from a line list ((M)

Description: A macro to set the Hadamard frequency list <a href="https://https

dll.out'. It assumes that the line list has already been created. The macro also sets ni to the Hadamard matrix size, creates htofsl, and sets fnl from

the minimum frequency difference in htfrq1.

See also: NMR Spectroscopy User Guide

Related: htfrq1 Hadamard frequency list in ni (P)

dll Display listed line frequencies and intensities (C)

htofs1 Hadamard offset in ni (P)

fn1 Fourier number in the 1st indirectly detected dimension (P)

ni Number of increments in the 1st indirectly detected dimension (P)

## sethw Set values for hardware in acquisition system (C)

Applicability: Syntax 1 through 5 apply to all systems. Syntax 6 applies only to systems with

a sample changer. Syntax 7 and 8 apply only to systems with a variable

temperature (VT) controller.

Syntax: The following syntax is used with the sethw command:

```
(1) sethw(<'wait'|'nowait',>par1,val1<,par2,val2,...)</pre>
```

(2) sethw('lock','on'|'off')

(3) sethw('spin', speed)

(4) sethw('spinner','bump')

(5) sethw('eject','on'|'off')

(6) sethw('loc', location)

(7) sethw('vt','reset'|'off')

(8) sethw('temp',temperature)

(9) sethw('lockfreq',lockfreq value)

Description: Sets acquisition system hardware values. sethw cannot be used when an acquisition is in progress or when the acqi program is active.

Syntax 1 can be used to set the lock system parameters lockpower, lockgain, lockphase, and z0. This syntax can also be used to set the values of the shims. The particular shim that can be set depends upon the type of shim hardware present in the system. See the description of shimset for a list of the shim names for each type of shim hardware.

Syntax 2 turns the hardware lock on or off.

Syntax 3 controls spinning speed.

Syntax 4 carries the sample to bump by giving it a short burst of eject air. This is sometimes useful to reseat the sample if it is failing to spin.

Syntax 5 ejects and inserts samples into the probe. Entering the command sethw('eject','on') is equivalent in function to macros eject and e; and sethw('eject','off') is equivalent to macros insert and i.

Syntax 6 sets a location for the sample currently in the magnet on a system with a sample changer. The parameter <u>loc</u> is updated.

Syntax 7 resets the VT controller, useful when changing the probe in a system with VT regulation. By entering sethw('vt','reset') after installing a new probe in the magnet and attaching the VT controller interface to the probe, the VT controller is ready to regulate the temperature. No other parameters can be modified by the command. As an alternate, you can manually turn the VT

controller unit off and then back on. Syntax 7 also turns the VT controller off by entering sethw('vt','off').

Syntax 8 sets the temperature in degrees celsius. The host computer does not wait for the temperature to regulate.

Syntax 9 sets the lock frequency, in MHz.

#### Arguments:

'wait' or 'nowait' keyword must be either the first or last argument.

- 'wait' sends the new values to the acquisition console, verifies these values, and updates the corresponding parameters. This is the default.
- 'nowait' sends the new values to the console without verifying them or changing parameters.

parameter1, value1, parameter2, value2, ... are paris of parameter names and their values (see the first two examples below). At least one parameter name and its value must be specified. A maximum of ten parameters can be set.

- 'lock', 'on' is a keyword pair to turn the hardware lock on.
- 'lock','off' is a keyword pair to turn the hardware lock off.
- 'liqbear' sets the bearing air on level; see liqbear parameter.
- 'pneufault' second argument is 'clear', 'n', 'w', or 'y' to clear or set the pneumatics fault code.
- 'spin' is a keyword that identifies the next argument, speed, as the sample spinning speed, in Hz.
- 'spinner', 'bump' is a keyword pair to bump the sample.
- 'eject', 'on' is a keyword pair to eject the sample from the probe.
- 'eject', 'off' is a keyword pair to insert the sample into the probe.
- 'loc' is a keyword to identify that the next argument, location, is a number for the sample currently in the magnet ('loc' is unrelated to the loc parameter).
- 'vt', 'reset' is a keyword pair to reset the VT controller after the controller has been disconnected from the probe. This is equivalent to turning the VT controller power off and on.
- 'vt', 'off' is a keyword pair to turn the VT controller off.
- 'temp' is a keyword that identifies the next argument, temperature, as the requested sample temperature, in degrees celsius.
- 'lockfreg' is a keyword that the next argument is the lock frequency.

lockfreq value is the lockfreq value, in MHz, for the lock frequency.

'lockrate' is a number <5000 used internally; usually 20 or 2000.

```
Examples: sethw('z1c',30,'z2c',-50)
        sethw('wait','z1',150,'z2',-400)
        sethw('lock','on')
        sethw('spin',20)
        sethw('spinner','bump')
        sethw('eject','on')
        sethw('loc',5)
        sethw('vt','reset')
        sethw('lockfreq',46.042)
```

See also: NMR Spectroscopy User Guide

Related: loc Location of sample in tray (P) lockpower Lock power (P) lockfreq Lock frequency (P) lockgain Lock gain (P)

lockphase Lock phase (P)

readhw Read current values of acquisition hardware (C)

spin Sample spin rate (P) z0 Z0 field position (P)

#### setint Set value of an integral (M)

Syntax: setint(int number<,value>)

Description: Sets the value of an integral.

Arguments: int number is the integral number. It corresponds to the index number

displayed by dli if all integrals are shown (i.e., intmod='full') or the region if alternating integrals are shown (i.e., intmod='partial').

value sets the actual value of the selected integral. The default is ins.

Examples: setint(2)

setint(1,3)

See also: NMR Spectroscopy User Guide

Related: dli Display list of integrals (C)

ins Integral normalization scale (P)
intmod Integral display mode (P)

### setlimit Set limits of a parameter in a tree (C)

Applicability: All

Syntax: setlimit(name, max,min,step [,tree])

setlimit(name, index[,tree])

Description: setlimit sets the limits of a variable in a tree.

The limits are max value, min. value and step size. A variable, such as an index into the table, can look up maximum, minimum, and step sizes in a table. Supplying all three (max, min., and step) arguments sets the parameter's protection bits (see setprotect) so that the table lookup is turned off. The parameter's protection bits are set so that table lookup is turned on if only a single index argument is supplied.

The step value is only used if the parameter is a real number.

Step Value	Parameter setting	
<-1	The parameter is set to the nearest larger value that is a power of 2. The fn parameter uses a step of -2 to select this case.	
>-1 and < 0	The inverse of the parameter is set to the nearest multiple of the absolute value of the step. The sw parameter uses a step of negative of the minimum dwell time to select this mode.	
>0 and <1	The parameter is set to the nearest multiple of the step value. As an equation, value = $n * step$ where $n$ is a positive or negative integer.	
≥1	The parameter is set to nearest value that is a multiple of step relative to the minimum value. For example, setlimit ('var', 3, -3, 2) allows only the following values -3, -1, 1, and 3. As an equation, value = $\min$ + $n*step$ where n is an integer >= 0. In this example, the equation is: value = $(-3)$ + $(n * 2)$ .	

Up to four optional return arguments can be used. The first will return the maximum, the second will return the minimum, and the third will return the step size. The fourth argument will return a 0 if the parameter is not using an indexed table lookup for the maximum, minimum, and step size. If the parameter is

using the table lookup mechanism, the fourth argument will be set to the index

for that table.

The variable trees are 'current', 'global', 'processed' and

'systemglobal'. The default tree is 'current'.

Arguments: name — the name of the variable.

tree — the variable tree: current (the default), global, processed, or

systemglobal.

Examples: setlimit('a',10000,0,.3)

setlimit('b',1e5,-3e2,1,'global')

setlimit('dpwr',9)

See also: User Programming

Related: create Create new parameter in a parameter tree (C

destroy Destroy a parameter (C)

display Display parameters and their attributes (C)

fread Read parameters from file and load them into a tree (C)

fsave Save parameters from a tree to a file (C)

getlimit Get the limits of a variable in a tree (C)

Edit a parameter and its attributes using vi text editor (M)

parmax Parameter maximum values (P)
parmin Parameter minimum values (P)
parstep Parameter step size values (P)

prune Prune extra parameters from current tree (C)
setgroup Set group of a parameter in a tree (C)
setprotect Set protection mode of a parameter (C)
settype Change type of a parameter (C)

setvalue Set value of any parameter in a tree (C)

setlk Set up lock parameters (M)

Syntax: setlk(solvent)

Description: Called from other macros to provide adjustment of locking and shimming as a

function of solvent. Removing quotation marks from around different parts of the text file of the macro places that particular section into effect. If the macro is left unchanged, setting <code>alock='s'</code> is required in the parameter sets where

used.

Arguments: solvent is the solvent to be used.

See also: NMR Spectroscopy User Guide

Related: alock Automatic lock status (P)

setlockfreq Set lock frequency (M)

Description: Calculates and sets the lock frequency parameter lockfreq. Before using

setlockfreq, you must acquire a signal using <sup>1</sup>H as the transmitter nucleus (tn='H1'). To avoid errors in calculating frequencies, set lockfreq='n'

before starting the acquisition.

See also: VnmrJ Installation and Administration

Related: lockfreq Lock frequency (P)

Nucleus for observe transmitter (P)

setLP Set up linear prediction in the direct dimension (M)

Applicability: ALL

Syntax: setLP(n)

Description: Sets up linear prediction in the direct dimension using the number of

coefficients specified.

Examples: setLP(3)

See also: NMR Spectroscopy User Guide

Related: lpext LP data extension in np dimension (P)

lpfiltLP coefficients to calculate in np dimension (P)lpnuptsLP number of data points in np dimension (P)lpoptLP algorithm data extension in np dimension (P)

Type of processing on np FID (P)

Set requency referencing based upon lock signal shift (M)

Starting point for LP data extension in np dimension (P)

Starting point for LP calculation in np dimension (P)

setLP1 Set F1 linear prediction parameters (M)

Syntax: setLP1<(extended\_length<,current\_length>)>

Description: Sets F1 linear prediction parameters. If no arguments are specified, the

interferograms are quadrupled in length.

Arguments: extended length is the number of complex points now existing (ni).

current\_length is the number of points desired after the (forward) linear

prediction.

See also: NMR Spectroscopy User Guide

Related: ni Number of increments in 1st indirectly detected dimension (P)

setlp0 Set parameters for zero linear phase (M)

Syntax: setlp0

Description: A new value of ddrtc is calculated by setlp0 using the current values of

alfa, rof2, and lp to achieve a zero linear phase condition (lp=0). A trial experiment must first be acquired and phased for pure absorption before running setlp0. A value of lp near zero is required for flat base line.

See also: NMR Spectroscopy User Guide

Related alfa Set alfa delay before acquisition (P)

ddrtc Set ddr time constant (P)

lp First-order phase in directly detected dimension (P)
rp Zero-order phase in directly detected dimension (P)
sw Spectral width in directly detected dimension (P)
rof2 Receiver gating time following pulse (P)

setnoether Disconnect host computer from Ethernet (U)

Description: Disconnects the host computer from the Ethernet network. Only root can

execute this shellscript properly. setnoether does nothing if the system is

already disconnected from the Ethernet network.

On systems running Solaris, setnoether renames the hostname.le0, defaultdomain, and defaultrouter files so that Ethernet is not

activated when the system is rebooted.

See also: VnmrJ Installation and Administration

Related: setether Connect or reconnect host computer to Ethernet (U)

Calculate offset frequency for given nucleus and ppm (M) setoffset

setoffset(nucleus,ppm):offsetfreq

Description: Using the setref macro, setoffset calculates the offset frequency for a

given chemical shift and returns the value.

Arguments: nucleus is the given nucleus.

ppm is the chemical shift.

offsetfreq returns the offset frequency for the given chemical shift.

Examples: setoffset(tn,5):tof

setoffset('C13',85):dof

See also: NMR Spectroscopy User Guide

Related: setref Set frequency referencing for proton spectra (M)

setparams Write parameter to current probe file (M)

Syntax: setparams(param, value<, nucleus>)

Description: Writes the value of a parameter to the current probe file. The name of the probe

file is referenced from the parameter probe.

Arguments: param is the name of the parameter to write.

value is a string with the value to be written for the parameter.

nucleus is the nucleus to write in the probe file. The default is the current

value of the parameter tn.

setparams('pw90','10') Examples:

setparams('pplvl','60')

setparams('dpwr',\$strdpwr,'H1')

See also: NMR Spectroscopy User Guide

addnucleus Add new nucleus to existing probe file (M) Related:

addparams Add parameter to current probe file (M) addprobe Create new probe directory and probe file (M) Retrieve parameter from probe file (M)

getparam

Probe type (P) probe

Nucleus for the observe transmitter (P)

updateprobe Update probe file (M)

Set maximum number of HP plotter pens (M) setpen

Syntax: setpen<(maxpen, max number pens)>

Allows the user to interactively define the maximum number of pens when Description:

changing to a Hewlett-Packard plotter.

Arguments: maxpen is the current value of the parameter maxpen.

> maximum number pens is the maximum number of pens to be used. If the value of max number pens is less than or equal to the current value of the

parameter maxpen, this value becomes the new value of maxpen.

See also: NMR Spectroscopy User Guide

Related: color Select plotting colors from a graphical interface (M)

> Maximum number of pens to use (P) maxpen

setplotdev Return characteristics of a named plotter (C)

Syntax: setplotdev<:plotter\_type,plotter\_host,ppmm,raster>

Description: Returns information from the devicenames and devicetable files to

identify the characteristics of a plotter. This command need never be entered directly by a user because it is automatically called whenever the plotter parameter is set. Note that different "types" of plotters (and printers) are characterized in devicetable. The devicenames file associates different

"names" to a given "type."

Arguments: plotter type returns the type of the named plotter.

plotter\_host returns the host associated with the plotter.

ppmm returns the plotter resolution in points per millimeter.

raster returns the value from the devicetable file.

See also: VnmrJ Installation and Administration

Related: plotter Plotter device (P)

setpower Set power and pulsewidth for a given  $\gamma$ B1 value (M)

Syntax: setpower(γB1, nucleus)

Description: Sets power level and pw90 values. For tn, setpower uses ref\_pwr and

ref\_pw90 from the parameter set or from the probe table. For dn, it uses ref\_pwx1vl and ref\_pwx90 from the parameter set or from the probe table. For dn2, it uses ref\_pwx2lvl and ref\_pwx290 from the parameter set or from the probe table. If the reference power levels and pulse width do not exist, setpower uses tpwr (pw90), dpwr (1/dmf) or dpwr2 (1/dmf2) (if the nucleus is tn, setpower uses tpwr; if the nucleus is dn, it uses

dpwr; if the nucleus is dn2, it uses dpwr2).

Arguments:  $\gamma B1$  is a given  $\gamma B1$  value.

nucleus is a given nucleus.

Examples: setpower(sw,tn)

setpower (5000, H1)

Related: dn Nucleus for first decoupler (P)

dn2 Nucleus for second decoupler (P)

dpwr Power level for first decoupler with linear amplifiers (P)

dpwr2 Power level for second decoupler (P)

pw90 90° pulse width (P)

Spectral width in directly detected dimension (P)

Observe transmitter power level with linear amplifiers (P)

setprotect Set protection mode of a parameter (C)

Syntax: setprotect(parameter,'set'|'on'|'off',bit\_vals<,tree>)

Description: Enables changing the protection bits associated with a parameter.

Arguments: parameter is the name of the parameter.

'set' causes the current protection bits for the parameter to be completely

replaced with the bits specified by bit vals.

'on' causes the bits specified in bit vals to be turned on without affecting

any other protection bits.

'off' causes the bits specified in bit vals to be turned off without

affecting any other protection bits.

'list' causes all parameter with the specified bit\_vals to be listed. This list may be returned to the calling macro.

'clear' option clears the specified bit\_vals from all parameters. For both the list and clear options, the names argument can be ''. The return value when setprotect is called with the list option can be used as the 'names' argument for other forms of setprotect. It can also be names for other commands which use lists of parameter names, such as writeparam and readparam.

bit vals is the *sum* of the *values* of bits selected from the following list:

Bit	Value	Description
0	1	Cannot array the parameter
1	2	Cannot change active/not active status
2	4	Cannot change the parameter value
3	8	Causes _parameter macro to be executed (e.g., if parameter is named sw, macro _sw is executed when sw is changed)
4	16	Avoids automatic redisplay
5	32	Cannot delete parameter
6	64	System ID for spectrometer or data station
7	128	Cannot copy parameter from tree to tree
8	256	Will not set array parameter
9	512	Cannot set parameter enumeral values
10	1024	Cannot change the parameter's group
11	2048	Cannot change protection bits
12	4096	Cannot change the display group
13	8192	Look up minimum, maximum, step values in table
14	16384	Parameter marked for locking (P_LOCK; see rtx)
15	32768	Global parameter not shared in multiple VJ viewports
16	65536	Force automatic redisplay in VJ templates

For example, to change the first two protection bits, with values 1 and 2, either enter setprotect twice (once for each value) with the keyword 'on', or enter setprotect once with bit\_vals set to 3 (sum of 1 and 2) with the keyword 'set'.

tree is one of the keywords 'global', 'current', 'processed', or 'systemglobal'. The default is 'current'. Refer to the create command for more information on the types of parameter trees.

Examples: setprotect('syn,'on',2)

setprotect('pslabel','on',8)

See also: User Programming

See also.			
Related:	array	Parameter order and precedence (P)	
	create	Create new parameter in a parameter tree (C)	
	destroy	Destroy a parameter (C)	
	display	Display parameters and their attributes (C)	
	fread	Read parameters from file and load them into a tree (C)	
	fsave	Save parameters from a tree to a file (C)	
	getlimit	Get the limits of a variable in a tree (C)	
	paramvi	Edit a parameter and its attributes using vi text editor (M)	
	prune	Prune extra parameters from current tree (C)	
	setlimit	Set limits of a parameter in a tree (C)	

#### setrc Set receiver constants (M)

Applicability: VNMRS and 400 - MR systems

Syntax: setrc

Description: Sets receiver time constants to optimal values. alfa is set to a minimum value

from the probe file (default is  $10 \, \mu s$ ). rof2 is set to a minimum value from the probe file (default is  $25 \, \mu s$ ). lp is set to zero. ddrtc is set to a value based upon the ddrpm parameter, which is set based upon pulse sequence type (default value ddrpm = 'p'). Linear prediction is turned on in the direct dimension if the ddrtc value is more than a dwell time. setrc is used in the apptype macros for setting up pulse sequences or from the command line to optimize

receiver constants.

Description: sets receiver time constants to optimal values.

See also: NMR Spectroscopy User Guide

Related alfa Set alfa delay before acquisition (P)

rof2 Receiver gating time following pulse (P)

pw Pulse width (P)
probe Probe type (P)

ddrtcSet ddr precession mode (P)ddrpmSet ddr precession mode (P)

Spectral width in directly detected dimension (P)

**Set F1** linear prediction parameters (M)

## setref Set frequency referencing (M)

Syntax: setref<(nucleus)>:\$rfl,\$rfp,\$reffrq,\$refpos

Description: Calc

Calculates the referencing for a given parameter or FID data set, for samples locked on deuterium, and based on the chemical shift of the lock solvent line. setref uses information in /vnmr/solvents (<sup>2</sup>H chemical shift for current solvent) and /vnmr/nuctables/nuctabref (absolute reference frequencies for NMR nuclei) to predict the position of the reference frequency with the current solvent, spectral window, and spectrometer frequency. setref assumes a locked sample.

The macro calculates the (auxiliary) 2H reference frequency (TMS-d1) from the lock frequency (lockf = lockfreq + lkof/le6) as follows:

```
H2 TMSfreq = lockf / (1 + solppm/1e6)
```

then takes the  $\Xi$  values for <sup>2</sup>H and tn and calculates the auxiliary reference frequency (reffrq) for the observe nucleus at the given field strength:

```
reffrq = (H2\_TMSfreq / \Xi(H2)) * \Xi(tn)
```

from this, rfl and rfp are set:

rfp=0 rfl = 
$$sw/2$$
 - (sfrq - reffrq) \*1e6.

Setting the global (or local) flag bioref = 'y' uses Bio-NMR referencing (based on nuctables/nuctabrefBio) rather than standard IUPAC / organic chemistry referencing (based on nuctables/nuctabref)

Ξ is the normalized frequency such that the <sup>1</sup>H signal from TMS is 100.00 MHz.

This estimate of the frequency based upon the chemical shift value of the lock signal and does not account for temperature, pH, or other factors affecting the chemical shift of the lock solvent.

The default tree is 'current'.

Arguments: An argument and return values are beneficial for the use of setref within

other macros such as setref1 and setref2. By default (i.e., without an

argument), setref calculates the referencing for 1D spectra or for the directly detected dimension in nD spectra (f2 in 2D, f3 in 3D).

When only nucleus is used as an argument, setref returns values without setting parameters.

\$rfl, \$rfp, \$reffrq, \$refpos are return values for reference peak
position, reference peak frequency, reference line frequency, and reference line
position, respectively.

Examples: setref

setref('C13'):\$rfl,\$rfp

See also: NMR Spectroscopy User Guide

Related: reffrq Reference frequency of reference line (P)

refpos Position of reference frequency (P)
rfl Reference peak position (P)
rfp Reference peak frequency (P)

Set reference line in directly detected dimension (M)

setref1 Set frequency referencing for 1st indirectly detected dimension (M)
setref2 Set frequency referencing for 2nd indirectly detected dimension (M)

setup Set up parameters for basic experiments (M)

tmsref Reference 1D proton or carbon spectrum to TMS (M)

bioref Use nuctables/nuctabrefBio) rather than standard

IUPAC / organic chemistry

## setref1 Set freq. referencing for 1st indirectly detected dimension (M)

Syntax: setref1(nucleus)

Description: Calculates the referencing for the first indirect dimension (f1) in nD parameters

and FID data sets, for samples locked on deuterium, and for the solvent specified by the solvent parameter. setref1 uses the setref macro to calculate the reference frequency and based on the chemical shift of the lock solvent line and /vnmr/nuctables/nuctabref (absolute reference frequencies for NMR nuclei) to predict the referencing in f1 (reffrq1, rfl1, rfp1) with the current solvent, sw1, and for the frequency of the specified nucleus.

ucicus.

This estimate of the frequency based upon the chemical shift value of the lock signal, as in setref, and does not account for temperature, pH, or other factors affecting the chemical shift of the lock solvent. Using setref, setref1, and setref2, maintains a consistent reference for all dimensions.

 $\Xi$  is the normalized frequency such that the <sup>1</sup>H signal from TMS is 100.00 MHz.

Setting the global (or local) flag bioref ='y' uses bio-NMR referencing (based on nuctables/nuctabrefBio) rather than standard IUPAC / organic chemistry referencing (based on nuctables/nuctabref)

See /vnmr/nuctables/nuctabref.

Arguments: nucleus is the frequency-relevant nucleus in f1.

Examples: setref1(tn)

setref1('C13')

See also: NMR Spectroscopy User Guide

Related: reffrq1 Reference frequency of reference line in 1st indirect dimension (P)

refpos1 Position of reference frequency in 1st indirect dimension (P)

rfl Reference peak position (P)

rfl1 Reference peak position in 1st indirectly detected dimension (P)
rfp1 Reference peak frequency in 1st indirectly detected dimension (P)

setref Set frequency referencing (M)

bioref Use nuctables/nuctabrefBio

#### setref2 Set freq. referencing for 2nd indirect detected dimension (M)

Syntax: setref2(nucleus)

Description: Calculates the referencing for the second indirect dimension (f2) in nD

parameters and FID data sets, for samples locked on deuterium, and for the solvent specified by the solvent parameter. setref2 uses setref to calculate the reference frequency and based on the chemical shift of the lock solvent line and /vnmr/nuctables/nuctabref (absolute reference frequencies for NMR nuclei) to predict the referencing in f2 (reffrq2, rfl2, rfp2) with the current solvent, sw2, and for the frequency of the specified nucleus.

This estimate of the frequency based upon the chemical shift value of the lock signal, as in setref, and does not account for temperature, pH, or other factors affecting the chemical shift of the lock solvent. Using setref, setref1, and setref2, maintains a consistent reference for all dimensions.

Setting the global (or local) flag bioref ='y' uses bio-NMR referencing (based on nuctables/nuctabrefBio) rather than standard IUPAC / organic chemistry referencing (based on nuctables/nuctabref)

See /vnmr/nuctables/nuctabref.

Arguments: nucleus is the frequency-relevant nucleus in f2.

Examples: setref2(tn)

setref2('C13')

See also: NMR Spectroscopy User Guide

Related: reffrq2 Reference frequency of reference line in 2nd indirect dimension (P)

refpos2Position of reference frequency in 2nd indirect dimension (P)rf12Reference peak position in 2nd indirectly detected dimension (P)rfp2Reference peak frequency in 2nd indirectly detected dimension (P)

Set reference line in 2nd indirectly detected dimension (M)

setref Set frequency referencing (M)

bioref Use nuctables/nuctabrefBio

# setscout Set up a scout run (M)

Applicability: Systems with LC-NMR accessory.

Description: Designed to help run simple experiments during the setup phase of LC-NMR or

to be the first of two experiments run on peaks in a stopped-flow or loop-flushing mode. In the latter application, you can set wexp='setwet au' so that the scout run is analyzed, parameters adjusted, and an appropriate solvent-

suppressed experiment run.

If parameters already exist in the current experiment for performing the lc1d
pulse sequence, setscout turns off the solvent suppression portion of the sequence; if they do not exist, they are created and set to default values using

Icld.

See also: NMR Spectroscopy User Guide

Related: lc1d Pulse sequence for LC-NMR (M)

**Set up a solvent-suppressed experiment (M)** 

setssfilter Set sslsfrq to the frequencies of each suppressed solvents (M)

Applicability: Systems with LC-NMR accessory.

Description: Sets sslsfrq to the frequencies of each of the suppressed solvents.

See also: NMR Spectroscopy User Guide

setsw Set spectral width (M)

Syntax: setsw(downfieldppm,upfieldppm)

Description: Sets sw and tof for the given spectral window and also does referencing.

Arguments: downfieldppm is the downfield frequency, in ppm.

upfieldppm is the upfield frequency, in ppm.

Examples: setsw(12,0)

setsw(235,-15)

See also: NMR Spectroscopy User Guide

Related: setsw1 Set spectral width in evolution dimension (M)

setsw2Set spectral width in 2nd evolution dimension (M)swSpectral width in directly detected dimension (P)tofFrequency offset for observe transmitter (P)

setsw1 Set spectral width in evolution dimension (M)

Syntax: setsw1(nucleus,downfieldppm,upfieldppm):offset

Description: Sets sw1 for the given spectral window and also does referencing.

Arguments: nucleus returns the nucleus.

 ${\tt downfield\,ppm\,is\,the\,downfield\,frequency,\,in\,ppm.}$ 

upfieldppm is the upfield frequency, in ppm.

offset returns the appropriate offset.

Examples: setsw1(tn,12,0)

setsw1(dn,235,-15):dof

See also: NMR Spectroscopy User Guide

sw1 Spectral width in 1st indirectly detected dimension (P)

setsw2 Set spectral width in 2nd evolution dimension (M)

Syntax: setsw2(nucleus,downfieldppm,upfieldppm):offset

Description: Sets sw2 for the given spectral window and also does referencing.

Arguments: nucleus returns the nucleus.

downfieldppm is the downfield frequency, in ppm.

upfieldppm is the upfield frequency, in ppm.

offset returns the appropriate offset.

Examples: setsw2(tn,12,0)

setsw2(dn,235,-15):dof

See also: NMR Spectroscopy User Guide

Related: setsw Set spectral width (M)

Spectral width in 2nd indirectly detected dimension (P)

setselfrqc Set selective frequency and width (M)

Description: Sets selective frequency and width of the excitation bandwidth for selective

excitation. Used after TOCSY1D and Noesy1d selection. Selected frequencies

and widths of the excitation bandwidth are used by suselfrg.

Related: Noesyld Change parameters for NOESYlD experiment (M)

Suselfrq Select peak, continue selective excitation experiment (M)

TOCSY1D Change parameters for TOCSY1D experiment (M)

setselinv Set up selective inversion (M)

Description: Sets power, pulsewidth, and shape for selective inversion; used by suselfrq.

By default, setselinv selects a q3 gaussian cascade pulse if a waveform generator or linear modulator is present. Otherwise, setselinv selects a

"rectangular" pulse.

Related: setselfrqc Select selective frequency and width (M)

suselfrq Select peak, continue selective excitation experiment (M)

settcldefault Select default display templates for pulse sequence (M)

Syntax: settcldefault<(<default><,sequence>)>

Description: Selects the display templates to use as the default for a pulse sequence.

Arguments: default is the name of the set of display templates to use for the default

display of the current pulse sequence (defined by the parameter seqfil). If no arguments are given, the user is prompted for the name of the display templates. sequence defines which pulse sequence will use the default displays of the pulse sequence given as the first argument. The default is the pulse sequence

defined by the parameter seqfil.

Examples: settcldefault

settcldefault('cosy')

settcldefault('default2d','HMQC8')

See also: User Programming

Related: segfil Pulse sequence name (P)

settune Opens the Auto Tune Setup dialog (M)

Applicability: VnmrJ Walkup, Automation

Syntax: settune

Description: Opens a dialog for setting when to tune in automation using ProTune.

See also: NMR Spectroscopy User Guide and VnmrJ Walkup

Related: protune Macro to start ProTune (M)

wtune Specify when to tune (P)

settype Change type of a parameter (C)

Syntax: settype(parameter,type<,tree>)

Description: Changes the type of an existing parameter. A string parameter can be changed

into a string or flag type, or a real parameter can be changed into a real, delay, frequency, pulse, or integer type. Note that settype cannot change a string

parameter into a real, or change a real into a string.

Arguments: parameter is the name of an existing parameter.

type is one of the keywords 'string', 'flag', 'real', 'delay',

'frequency', 'pulse', or 'integer'.

tree is one of the keywords 'global', 'current', 'processed', or

'systemglobal'. The default is 'current'. Refer to the create

command for more information on the types of parameter trees.

Examples: settype('in','flag','global')

settype('p12','pulse')

See also: User Programming

Related: create Create new parameter in a parameter tree (C)

displayDisplay parameters and their attributes (C)setgroupSet group of a parameter in a tree (C)setlimitSet limits of a parameter in a tree (C)setprotectSet protection mode of a parameter (C)setvalueSet value of any parameter in a tree (C)

#### setup Set up parameters for basic experiments (M)

Syntax: setup<(nucleus<,solvent>)>

Description: Returns a parameter set to do the experiment requested, complete with

positioning of the transmitter and decoupler. Parameters set by setup are recalled from the /vnmr/stdpar directory or from the user's stdpar directory if the appropriate file exists there. Any changes made to the files in these directories are reflected in setup. The default parameters for carbon and proton survey spectra are in files /vnmr/stdpar/C13.par and /vnmr/stdpar/H1.par, respectively. These files should be modified as desired to

produce spectra under desirable conditions.

Arguments: nucleus is a nucleus chosen from the files in /vnmr/stdpar or in the

user's stdpar directory (e.g., 'H1', 'C13', 'P31').

solvent is a solvent chosen from the file /vnmr/solvents (e.g.,

'CDC13', 'C6D6', 'D20'). The default is 'CDC13'.

Examples: setup

setup('H1')

setup('C13','DMSO')

See also: NMR Spectroscopy User Guide

## setup dosy Set up gradient levels for DOSY experiments (M)

Description: Initiates a dialogue to set up an array of qzlvl1 values for DOSY experiments.

setup\_dosy requests the number of array increments and an initial and a final gzlvl1 value and sets up an array that gives increments in gzlvl1 squared between these limits. setup\_dosy retrieves the gradient strength from the probe calibration file if probe<>'' and stores it in the local experimental parameter DAC\_to\_G. If probe='' (i.e., the probe is not defined), then DAC\_to\_G is set to the current value of the global parameter

gcal.

See also: NMR Spectroscopy User Guide

Related: dosy Process DOSY experiments (M)

DAC\_to\_G Parameter to store gradient calibration value in DOSY sequences (P)

setgcal Set the gradient calibration constant (M)

#### setvalue Set value of any parameter in a tree (C)

Syntax: setvalue(parameter, value<, index><, tree>)

Description: Sets the value of any parameter in a tree. This command bypasses the normal

range checking for parameter entry, as well as bypassing any action that would

be invoked by the parameter's protection mode (see the setprotect

command). If the parameter entry normally causes a parameter macro to be

executed, this action also is bypassed.

Arguments: parameter — name of the parameter.

value —set value for the parameter.

index — number of a single element in an arrayed parameter.

The default is 1. A value of 0 for the index resets an arrayed (or non-arrayed) parameter to the one element supplied as the second argument to setvalue.

tree — keyword 'global', 'current', 'processed', or
'systemglobal'. The default is 'current'. Refer to the create

command for more information on the types of parameter trees.

Examples: setvalue('arraydim',128,'processed')

See also: *User Programming* 

Related: create Create new parameter in a parameter tree (C)

setprotect Set protection mode of a parameter (C)

## setwave Write a wave definition string into Pbox.inp file (M)

Syntax: setwave('sh bw/pw ofs st ph fla trev d1 d2 d0')

Description: Sets up a single excitation band in the Pbox.inp file. An unlimited number of

waves can be combined by reapplying setwave.

Arguments: A single string of 1 to 10 wave parameters in predefined order. Note that a single

quote is required at the start and the end of the entire string, but no single quotes

are required surrounding characters and strings inside the entire string.

sh name of a shape file.

bw/pw either the bandwidth, in Hz, or the pulsewidth, in sec.

ofs offset, in Hz.

number specifying the spin status:

0 for excitation1 for de-excitation0.5 for refocusing.

ph phase (or phase cycle, see wavelib/supercycles).

fla flip angle.

fla can override the default flip angle.

trev time reversal. This can be used to cancel time reversal if spin status

(st) is set to 1 for Mxy.

dl delay, in sec, prior the pulse.

delay, in sec, after the pulse.

d0 delay or command prior to d1.

If d0=a, the wave is appended to the previous wave.

Examples: setwave('eburp1')

setwave('GARP 12000.0')

setwave('esnob 600 -1248.2 1 90.0 n n 0.001')

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

setwin Activate selected window (C)

Syntax: setwin(row<,column>)

Description: Activates a specific pane in the graphics window. Panes are numbered

sequentially from left to right and top to bottom.

Arguments: row is the number of the row containing the pane to be activated.

column is the number of the column containing the pane to be activated.

Examples: setwin(3)

setwin(1,2)

See also: NMR Spectroscopy User Guide

Related: curwin Current window (P)

fontselect Open FontSelect window (C)
jwin Activate current window (M)
mapwin List of experiment numbers (P)
setgrid Activate selected window (M)

sf Start of FID (P)

Description: Sets the start of the FID display. This parameter can be entered in the usual way

or interactively controlled by the sf wf button during a FID display.

Values: 0 to the value of at, in seconds. See also: *NMR Spectroscopy User Guide* 

Related: at Acquisition time (P)

dcon Display noninteractive color intensities map (C)

dconi Interactive 2D data display (C)
df Display a single FID (C)

Start of interferogram in 1st indirectly detected dimension (P)

Start of interferogram in 2nd indirectly detected dimension (P)

vf Vertical scale of FID (P) wf Width of FID (P)

sf1 Start of interferogram in 1st indirectly detected dimension (P)

Description: Sets the start of the interferogram display in the first indirectly detected

dimension.

Values: 0 to  $(2 \times ni)/sw1$ , in seconds. See also: *NMR Spectroscopy User Guide* 

Related: ni Number of increments in 1st indirectly detected dimension (P)

sf Start of FID (P)

Spectral width in 1st indirectly detected dimension (P)

wf1 Width of interferogram in 1st indirectly detected dimension (P)

sf2 Start of interferogram in 2nd indirectly detected dimension (P)

Description: Sets the start of the interferogram display in the second indirectly detected

dimension.

Values: 0 to  $(2 \times ni2)/sw2$ , in seconds. See also: *NMR Spectroscopy User Guide* 

Related: ni2 Number of increments in 2nd indirectly detected dimension (P

sf Start of FID (P)

Spectral width in 2nd indirectly detected dimension (P)

Wf2 Width of interferogram in 2nd indirectly detected dimension (P)

sfrq Transmitter frequency of observe nucleus (P)

Description: Contains the frequency for the observe transmitter. sfrq is automatically set

when tn is changed, and it should not be necessary for the user to manually set

this parameter.

Values: Number, in MHz.

See also: NMR Spectroscopy User Guide

Related: dfrq Transmitter frequency of first decoupler (P)

dfrq2 Transmitter frequency of second decoupler (P)
dfrq3 Transmitter frequency of third decoupler (P)

Nucleus for observe transmitter (P)

Frequency offset for observe transmitter (P)

spcfrq

Display frequencies of rf channels (M)

sh2pu1 Set up for a shaped observe excitation sequence (M)

Applicability: Systems with waveform generators.

Syntax: sh2pul

Description: Behaves like standard two-pulse sequence S2PUL but with the normal hard

pulses changed into shaped pulses from the waveform generator. The name of the shaped pulse associated with pw is pwpat and p1 is p1pat. Information about the specifics of power settings and bandwidths is available from the

macros bandinfo and pulseinfo.

See also: User Programming

Related: bandinfo Shaped pulse information for calibration (M)

plpat Shape of an excitation pulse (P)

pwpat Shape of refocusing pulse (P)

pulseinfo Shaped pulse information for calibration (M)

shdec Set up for shaped observe excitation sequence (M)

Applicability: Systems with waveform generators.

Description: Sets up the SHDEC pulse sequence that generates a shaped pulse on the observe

channel using the waveform generator. It also allows for programmed (e.g.: multiselective) homodecoupling or solvent presaturation using the observe transmitter, and an optional gradient pulse following the excitation

pulse.

See also: NMR Spectroscopy User Guide

Related: Pbox Pulse shaping software (U)

shell Start a UNIX shell (C)

Syntax: shell<(command)>:\$var1,\$var2,...

Description: Brings up a normal UNIX shell for the user. On the Sun, a pop-up window is

created. On the GraphOn terminal, the entire terminal is used.

Arguments: command is a UNIX command line to be executed by shell. The default is

to bring up a UNIX shell. If the last character in the command line is the symbol &, the command is executed in background, which allows commands to be

entered and executed while the shell command is still running. Note that if this background feature is used, any printed output should be redirected to a file. Otherwise, the output may pop up in the text window at random times.

shell calls involving pipes or input redirection (<) require either an extra pair of parentheses or the addition of; cat to the shell command string.

\$var1, \$var2,... are names of variables to hold text lines that are
generated as a result of the UNIX command. The default is to display the text
lines. Each variable receives a single display line. shell always returns a text
line; in many cases, it is a simple carriage return. To prevent this carriage return
from being shown, capture it in a dummy variable, such as

```
shell('command'):$dum
```

#### Examples: shell

shell('ps')

shell('ls -lt'):\$filelist

shell(systemdir+'/acqbin/Acqstat '+hostname+' &')

shell('ls -t|grep May; cat')

or

shell('(ls -t|grep May)')

See also: NMR Spectroscopy User Guide, User Programming

Related: shelli Start an interactive UNIX shell (C)

## shelli Start an interactive UNIX shell (C)

Syntax: shelli(command)

Description: On a terminal, runs interactively the UNIX command line given as the

argument. No return or output variables are allowed.

Arguments: command is a UNIX command line to be executed.

Examples: shelli('vi myfile')

See also: NMR Spectroscopy User Guide, User Programming

Related: shell Start a UNIX shell (C)

#### shim Submit an Autoshim experiment to acquisition (C)

Description: Performs validity checks on the acquisition parameters and then submits an

Autoshim experiment to acquisition.

See also: NMR Spectroscopy User Guide

Related: au Submit experiment to acquisition and process data (C)

change Submit a change sample experiment to acquisition (M) ga Submit experiment to acquisition and FT the result (C)

go Submit experiment to acquisition (C)

lock Submit an Autolock experiment to acquisition (C)

Submit change sample, autoshim experiment to acquisition (M)

Submit a spin setup experiment to acquisition (C)
Su Submit a setup experiment to acquisition (M)

## shimset Type of shim set (P)

Description: Configuration parameter for the type of shims on the system. The value of

shimset is set using the Shimset label in the Spectrometer Configuration

window.

Values: 1 to 14, where the value identifies one of the following shim sets:

- 1 is a shim set in a Varian 13-shim supply with computer-controlled axial shims z1, z1c, z2, z2c, z3, z4, and radial shims x1, y1, xz, yz, xy, x2y2, x3, y3. Shims can be adjusted from -2047 to +2047. This value is used with the Ultra•nmr shim system when operated from the HIM box (Varian 13 Shims choice in Spectrometer Configuration window).
- 2 is a shim set in a Oxford 18-shim supply with computer-controlled axial shims z1, z1c, z2, z2c, z3, z4, z5, and radial shims x1, y1, xz, yz, xy, x2y2, x3, y3, xz2, yz2, zxy, zx2y2. Shims can be adjusted from -2047 to +2047 (Oxford 18 Shims choice in Spectrometer Configuration window).
- 3 is a shim set in a Varian 23-shim supply with computer-controlled axial shims z1, z2, z3, z4, z5, z6, and radial shims x1, y1, xz, yz, xy, x2y2, x3, y3, xz2, yz2, zxy, zx2y2, z3x, z3y, z2x2y2, z2xy. Shims can be adjusted from -32767 to +32767 (Varian 23 Shims choice in Spectrometer Configuration window).
- 4 is a shim set in a Varian 28-shim supply with computer-controlled axial shims z1, z2, z3, z4, z5, z6, z7, and radial shims x1, y1, xz, yz, xy, x2y2, x3, y3, xz2, yz2, zxy, zx2y2, z3x, z3y, z2x2y2, z2xy, zx3, zy3, z4x, z4y. Shims can be adjusted from -32767 to +32767 (Varian 28 Shims choice in Spectrometer Configuration window).
- 5 is a shim set in an Ultra•nmr shim system (39 shim channels) with computer-controlled axial shims z1, z1c, z2, z2c, z3, z3c, z4, z4c, z5, z6, z7, z8, and radial shims x1, y1, xz, yz, xy, x2y2, x3, y3, xz2, yz2, zxy, zx2y2, z3x, z3y, z2x2y2, z2xy, zx3, zy3, z4x, z4y, z3x2y2, z3xy, z2x3, z2y3, z3x3, z3y3, z4x2y2, z4xy, z5x, z5y. Shims can be adjusted from –32767 to +32767 (Ultra Shims choice in Spectrometer Configuration window).
- 6 is a shim set in a Varian 18-shim supply with computer-controlled axial shims z1, z2, z3, z4, z5, and radial shims x1, y1, xz, yz, xy, x2y2, x3, y3, xz2, yz2, zxy, zx2y2. Shims can be adjusted from -32767 to +32767 (Varian 18 Shims choice in Spectrometer Configuration window).
- 7 is a shim set in a Varian 20-shim supply with computer-controlled axial shims z1, z2, z3, z4, z5, and radial shims x1, y1, xz, yz, xy, x2y2, x3, y3, xz2, yz2, zxy, zx2y2, z3x, z3y. Shims can be adjusted from -32767 to +32767 (Varian 20 Shims choice in Spectrometer Configuration window).
- 8 is a shim set in a Oxford 15-shim supply with computer-controlled axial shims z1, z2, z3, z4, and radial shims x1, y1, xz, yz, xy, x2y2, zx2y2, xz2, yz2, zxy. Shims can be adjusted from –2047 to +2047 (Oxford 15 Shims choice in Spectrometer Configuration window).
- 9 is a shim set in a Varian Ultra•nmr shim system II (40 shim channels) with computer-controlled axial shims z1, z1c, z2, z2c, z3, z3c, z4, z4c, z5, z6, z7, z8, and radial shims x1, y1, xz, yz, xy, x2y2, x3, y3, x4, y4, xz2, yz2, zxy, zx2y2, z3x, z3y, z2x2y2, z2xy, zx3, zy3, z4x, z4y, z3x2y2, z3xy, z2x3, z2y3, z3x3, z3y3, z4x2y2, z4xy, z5x, z5y. Shims can be adjusted from -32767 to +32767 (Varian 40 Shims choice in Spectrometer Configuration window).
- 10 is a shim set in a Varian 14-shim supply with computer-controlled axial shims z1, z1c, z2, z2c, z3, z4, z5, and radial shims x1, y1, xz, yz, xy, x2y2, x3, y3. Shims can be adjusted from –2047 to +2047 (Varian 14 Shims choice in Spectrometer Configuration window).
- 11 is a shim set in a Varian 8-shim supply with computer-controlled axial shims z1, z2, and radial shims x1, y1, xz, yz, xy, x2y2. Shims can be adjusted from 32767 to +32767 (Whole Body Shims choice in Spectrometer Configuration window).
- 12 is a shim set in a Varian 26-shim supply with computer-controlled axial shims z1, z2, z3, z4, z5, and radial shims x1, y1, xz, yz, xy, x2y2, x3, y3, xz2, yz2, zxy, zx2y2, z3x, z3y, z2x2y2, z2xy, zx3, zy3, x4, y4. Shims can be adjusted

from –32767 to +32767 (Varian 26 Shims choice in Spectrometer Configuration window).

13 is a shim set in an Varian 29-shim supply with computer-controlled axial shims z1, z2, z3, z4, z5, z6, and radial shims x1, y1, xz, yz, xy, x2y2, x3, y3, xz2, yz2, zxy, zx2y2, z3x, z3y, z2x2y2, z2xy, zx3, zy3, z4x, z4y, z5x, z5y. Shims can be adjusted from -32767 to +32767 (Varian 29 Shims choice in Spectrometer Configuration window).

14 is a shim set in a Varian 35-shim supply with computer-controlled axial shims z1, z2, z3, z4, z5, z6, and radial shims x1, y1, xz, yz, xy, x2y2, x3, y3, x4, y4, xz2, yz2, zxy, zx2y2, z3x, z3y, z2x2y2, z2xy, zx3, zy3, z4x, z4y, z3x2y2, z3xy, z4x2y2, z4xy, z5x, z5y. Shims can be adjusted from -32767 to +32767 (Varian 35 Shims choice in Spectrometer Configuration window).

15 is the Varian 15 Shim. 16 is the Ultra 18 Shims.

See also: VnmrJ Installation and Administration

Related: config Display current configuration and possibly change it (M

readhw Read current values of acquisition hardware (C)

showconfig Show system configuration settings (M)

See also: Displays the system configuration settings in the text window. To print the

settings, enter the following in the VnmrJ command line:

printon showconfig printoff.

See also: *VnmrJ Installation and Administration* 

Related: config Display current configuration and possibly change it (M

showconsole Show system configuration settings (U)

Description: Displays console hardware configuration parameters and system versions. This

information is recorded during console bootup and represents the system hardware options recognized by the acquisition computer. The command is used

mainly when troubleshooting or performing diagnostics.

See also: NMR Spectroscopy User Guide

Related: ihwinfo Hardware status of console (C)

showfit Display numerical results of deconvolution (M)

Description: After a deconvolution, the results are written into file fitspec.outpar in

an abbreviated format. showfit converts these data to an output format more

suitable for examination and printing.

See also: NMR Spectroscopy User Guide

Related: fitspec Perform spectrum deconvolution (C)

plfit Plot deconvolution analysis (M)

usemark Use "mark" output as deconvolution starting point (M)

showloginbox Shows operator login dialog (M)

Description: Shows the login dialog for operators.

shownumx Show x position of number (P)

Description: Show the **X** position of the number. The bottom left of every spectrum is defined

as 0.

See also: User Programming

Related: shownumy y position counting from bottom left of every spectrum (P)

shownumy Show y position of number (P)

Description: Show the Y position of the number. The bottom left of every spectrum is defined

as 0.

See also: User Programming

Related: shownumx x position counting from bottom left of every spectrum (P)

showoriginal Restore first 2D spectrum in 3D DOSY experiment (M)

Description: Restores the first 2D spectrum in a 3D DOSY experiment (if it has been saved

by the dosy macro).

See also: NMR Spectroscopy User Guide

Related: dosy Process DOSY experiments (M)

showplotter Show list of currently defined plotters and printers (M)

Description: Shows a list of currently defined plotters and printers.

See also: NMR Spectroscopy User Guide

Related: plotter Plotter device (P)

printer
Printer device (P)

showplotq Display plot jobs in plot queue (M)

Description: Displays current plot jobs in the plot queue for the active plotter.

See also: NMR Spectroscopy User Guide

Related: killplot Stop plot jobs and remove from plot queue (C

showprintq Display print jobs in print queue (C)

showprintq Display print jobs in print queue (M)

Description: Displays current print jobs in the print queue for the active printer.

See also: NMR Spectroscopy User Guide

Related: killprint Stop print jobs and remove from print queue (C)

showplotq Display plot jobs in plot queue (M)

showprotunegui show the graphical interface while tuning (P)

Syntax: showprotunequi='argument'

Description: This is a global string parameter that does not exist by default. The user can

create it to force the ProTune GUI to be shown during normal tuning operation.

Arguments: 'n' — Do not force the GUI to be shown.

'y' — Show the GUI, except in automation.
'a' — Always show the GUI, even in automation.

Set showprotunegui='a' will cause ProTune to fail in automation unless the proper display permission has been set. Set the display permissions

on Linux systems by executing "xhost local:" on the Linux command line.

See also: NMR Spectroscopy User Guide

Related: protune Macro to start ProTune (M)

showrfmon Show RF Monitor Button in Hardware Bar (P)

Applicability: Imaging

Syntax: showrfmon=<value>

Description: Show RF Monitor Button in Hardware Bar.

Values: 1 show RF Monitor button.

-1 hide RF Monitor button.

See also: VnmrJ Imaging User Guide

showstat Display information about status of acquisition (M,U)

Syntax: (From VnmrJ) showstat< (remote system) >

(From UNIX) showstat <remote system>

Description: Displays information in the text screen about the status of acquisition on a

spectrometer. The command is similar to Acqstat, but displays the

information in a non-graphical manner and only once.

Arguments: remote system is the host name of a remote spectrometer. The default is to

display information about acquisition on the local system.

See also: NMR Spectroscopy User Guide

Related: Acqstat Bring up the acquisition status display (U)

sin Find sine value of an angle (C)

Syntax: sin(angle)<:n>

Description: Finds the sine value of an angle.

Arguments: angle is the angle given in radians.

n is a return value giving the sine of angle. The default is to display the sine

value in the status window.

Examples: sin(.5)

sin(val):sin\_val

See also: User Programming

Related asin Find arc sine of number (C)

atan Find arc tangent of a number (C)

cos Find cosine value of an angle (C)

Find exponential value (C)

exp Find exponential value (C)

In Find natural logarithm of a number (C) tan Find tangent value of an angle (C)

sine Find values for a sine window function (M)

Syntax: sine<(shift<,number\_points<,domain>)>

Description: Calculates appropriate values for parameters sb and sbs (if the domain

argument is 'f2') or for parameters sb1 and sbs1 (if the domain argument is 'f1') in order to achieve a sine window function. The value of the parameter

trace is used if the domain argument is not entered.

Arguments: If shift is greater than 1, the sbs parameter is calculated as 2\*sb/shift

(sbs1 is calculated as 2\*sb1/shift). sine (2) gives a "PI/2-shifted" sine

window, i.e., cosine weighting. sine (3) gives a "PI/3" shifted sine window, etc. If shift is less than or equal to 1, an unshifted sine window is used (sbs='n' or sbs1='n').

number\_points specifies the number of real points that the window function spans. The value of the window function for subsequent points is 0. number\_points must be greater than 0 and a multiple of 2. The default is ni\*2 if trace='f1', or np if trace='f2'.

domain is 'f1' or 'f2'. The default is the current setting of trace.

See also: NMR Spectroscopy User Guide

Related: np Number of data points (P)

sbSinebell const. in directly detected dimension (P)sb1Sinebell const. in 1st indirectly detected dimension (P)sbsSinebell shift const. in directly detected dimension (P)sbs1Sinebell shift const. in 1st indirectly detected dimension (P)sinesqFind values for a sine squared window function (M)

trace Mode for *n*-dimensional data display (P)

## sinebell Select default parameters for sinebell weighting (M)

Description: Generates initial guess at good sinebell weighting parameters by setting the sb

and sb1 parameters to one-half the acquisition time and turning off all other

weighting. Use sinebell in absolute-value 2D experiments only.

See also: NMR Spectroscopy User Guide

Related: pseudo Set default parameters for pseudo-echo weighting (M)

Sinebell const. in directly detected dimension (P)
Sinebell const. in 1st indirectly detected dimension (P)

#### sinesq Find values for a sine-squared window function (M)

Syntax: sinesq<(shift<,number points<,domain>)>

Description: Calculates appropriate values for parameters sb and sbs (if the domain

argument is 'f2') or for parameters sb1 and sbs1 (if the domain argument is 'f1') in order to achieve a sine-squared window function. The value of

parameter trace is used if the domain argument is not entered.

Arguments: shift sets the starting value for the window function. If shift is greater than

0, the starting value is given by sin p/shift; otherwise, if shift is less

than or equal to 0, the starting value is 0. The default value is 0.

number\_points specifies the number of real points that the window function spans. The value of the window function for subsequent points is 0. The number\_points argument must be greater than 0 and a multiple of 2. The

default is ni \*2 if trace='f1', or np if trace='f2'.

domain is 'f1' or 'f2'. The default is the current setting of trace.

See also: NMR Spectroscopy User Guide

Related: ni Number of increments in 1st indirectly detected dimension (P)

np Number of data points (P)

sbSinebell const. in directly detected dimension (P)sb1Sinebell const. in 1st indirectly detected dimension (P)sbsSinebell shift const. in directly detected dimension (P)

Find values for a sine window function (M)

trace Mode for *n*-dimensional data display (P)

size Returns the number of elements in an arrayed parameter (O)

Description: In MAGICAL programming, an operator that returns the number of elements in

an arrayed parameter.

Examples: r1 = size('d2')
See also: User Programming

Related: arraydim Dimension of experiment (P)

typeof Return identifier for argument type (O)
length Determine length of a string (C)

slfreq Measured line frequencies (P)

Description: Contains a list of measured line frequencies. In iterative spin simulation, a

calculated spectrum is matched to the lines in the list. The spinll macro fills in slfreq from the last line listing or a mark operation. Use assign to make assignments between the measured lines and the calculated transitions. slfreq

is a global parameter and is displayed by dla.

See also: NMR Spectroscopy User Guide .

Related: assign Assign transitions to experimental lines (M)

cla Clear all line assignments (M)

dla Display spin simulation parameter arrays (M)

fitspec Perform spectrum deconvolution (C)

mark Determine intensity of a spectrum at a point (C)
spinll Set up an slfreq array (M)

slw Spin simulation linewidth (P)

Description: Sets linewidth for individual transitions in the displayed spectrum. Only one

linewidth is provided, so all transitions must be given the same linewidth. If the Set Params button is used in setting up spin simulation parameters, slw is automatically set to the measured linewidth of the tallest line displayed.

slw is also the starting default linewidth for deconvolution calculations. This linewidth will be set automatically when deconvolution is operated using the menu mode and is bypassed if the usemark command has been used in

conjunction with two cursor input.

Values: 0.01 to 1e6. The typical value is 1. See also: *NMR Spectroscopy User Guide* 

Related: usemark Use "mark" output as deconvolution starting point (M)

smaxf Maximum frequency of any transition (P)

Description: Sets the maximum frequency limit for the calculation of the final simulated

spectrum. It should be set before the calculation is performed. If the Set Params button is used in setting up spin simulation parameters, smaxf is initialized to sp+wp; which assumes that you have already expanded the region of the spectrum that you wish to simulate before beginning the spin simulation

process.

Values: -1e10 to 1e10, in Hz. The typical value is the maximum chemical shift +50.

See also: NMR Spectroscopy User Guide

Related: sminf Minimum frequency of any transition (P)

sp Start of plot (P)
wp Width of plot (P)

Minimum frequency of any transition (P) sminf

Description: Sets the minimum frequency limit for the calculation of the final simulated

> spectrum. It should be set before the calculation is performed. If the Set Params button is used in setting up spin simulation parameters, sminf is initialized to sp, which assumes that you have already expanded the region of the spectrum

that you wish to simulate before beginning the spin simulation process.

Values: -1e10 to 1e10, in Hz. The typical value is 0.

See also: NMR Spectroscopy User Guide

Related: smaxf Maximum frequency of any transition (P)

> Start of plot (P) Width of plot (P) qw

smsport Sample Management System serial port connection (P)

Description: Sets which serial port on the host computer is connected to a Sample

> Management System (i.e., a sample changer). The value of smsport is set using the Sample Changer Serial Port label in the Spectrometer Configuration

window.

'a' sets the connection for serial port A. This value is the default. Values:

'b' sets the connection for serial port B.

See also: VnmrJ Installation and Administration; NMR Spectroscopy User Guide

Related: Display current configuration and possibly change it (M) config

Signal-to-noise ratio (P) sn

> Description: Sets a ratio for testing signal-to-noise. The testsn macro checks whether a

> > signal-to-noise ratio equal to sn has been achieved.

Values: Typical value is 35.

See also: NMR Spectroscopy User Guide

Related: dsn Measure signal-to-noise (C)

> getsn Get signal-to-noise estimate of a spectrum (M)

Test signal-to-noise of a spectrum (M) testsn

testct Check ct for resuming signal-to-noise testing (M)

Return ppm and peak width of solvent resonances (M) solppm

Syntax: solppm:chemical shift, peak width

Description: Returns to the calling macro information about the chemical shift and peak

> spread of solvent resonances in various solvents for either <sup>1</sup>H or <sup>13</sup>C, depending on the observe nucleus tn and the parameter solvent. This macro is used

"internally" by other macros only.

Arguments: chemical shift returns the chemical shift of the solvent in ppm.

peak width returns the approximate peak spread of solvent resonances.

See also: User Programming

Related: Lock solvent (P) solvent

> tn Nucleus for observe transmitter (P)

solvent Lock solvent (P)

Description: Contains one of a series of lock solvents from the /vnmr/solvents file,

which contains the <sup>2</sup>H chemical shift of each lock solvent. By editing the file, additional solvents can be added. Values for solvent are not case-sensitive

(e.g., solvent='C6D6' and solvent='C6d6' are identical)

The auto\_dir macro now controls most of the automation features, including

setting the value of solvent.

Values: Standard values in /vnmr/solvents include:

Deuterium Oxide CDCl3 MethyleneChloride
D2O Cyclohexane MethylAlcohol-d4

Acetone C6Dl2 CD2Cl2
CD3COCD3 Toluene CD3OD
Benzene C6D5CH3 Chloroform

C6D6 Acetic\_Acid
DMSO CD3COOD

See also: NMR Spectroscopy User Guide

Related: lastlk Last lock solvent used (P)

solvinfo Retrieve information from solvent table (C) tof Frequency offset for observe transmitter (P)

solvinfo Retrieve information from solvent table (C)

Syntax: solvinfo(solvent):\$chemical\_shift,\$name

Description: Retrieves solvent shift and solvent name from the solvent table.

Arguments: solvent is the name of a solvent from the /vnmr/solvents file. This

argument is not case-sensitive (e.g., ' c6d6' is the same as ' C6D6').

chemical shift returns the chemical shift of the solvent, in ppm.

name returns the name of the solvent. The name returned will match the case

of the letters (upper or lower) in /vnmr/solvents.

Examples: solvinfo('acetone'):\$shift

solvinfo('d2o'):\$shift,solvent

See also: NMR Spectroscopy User Guide

Related: lookup Look up words and lines from a text file (C)

solvent Lock solvent (P)

sort Sort real values of a parameter (M)

Syntax: sort(parametername<,sortType>:order,val

Description: Sorts the real values of a parameter. The sort macro is not used for parameters

holding string values. The default behavior is to the array into values of increasing value. A sortType can be given to sort into descending order ('r').

If only unique values are wanted, the 'u' sortType can be used. The 'ru'

sortType given unique values in descending order.

The name of a parameter is the first argument to sort. Two return values hold the results of the sort. The first return value is an array containing the original indexes of the sorted array. The second return value gives the sorted array.

Examples: With par=10,8,6,4,2 the display('par') command will show:

```
[1] = 10
[2] = 8
[3] = 6
[4] = 4
[5] = 2
```

The command sort ('par'): \$order, \$val will set:

```
$order=5,4,3,2,1
$val =2,4,6,8,10
```

#### sp Start of plot in directly detected dimension (P)

Description: Low-frequency limit of the display or plotted region of the spectrum. sp is

always stored in Hz, but can be entered in ppm by using the p suffix

(e.g., sp=2p sets the start of plot to 2 ppm).

See also: NMR Spectroscopy User Guide

Related: sp1 Start of plot in 1st indirectly detected dimension (P) sp2 Start of plot in 2nd indirectly detected dimension (P)

#### sp1 Start of plot in 1st indirectly detected dimension (P)

Description: Analogous to the sp parameter except that sp1 applies to the first indirectly

detected dimension of a multidimensional data set.

See also: NMR Spectroscopy User Guide

Related: sp Start of plot in directly detected dimension (P)

Start of plot in 2nd indirectly detected dimension (P)

#### sp2 Start of plot in 2nd indirectly detected dimension (P)

Description: Analogous to the sp parameter except that sp2 applies to the second indirectly

detected dimension of a multidimensional data set.

See also: NMR Spectroscopy User Guide

Related: sp Start of plot in directly detected dimension (P)

# spadd Add current spectrum to add/subtract experiment (C)

```
Syntax: (1) spadd< (multiplier<, shift>) >
```

(2) spadd('new')

(3) spadd('trace',index)

Description: Performs noninteractive spectral addition. The last displayed or selected

spectrum is added to the current contents of the add/subtract experiment (exp5). A multi-element add/subtract experiment can be created using the 'new' keyword. Individual spectra in a multi-element add/subtract experiment can be subsequently added to using the 'trace' keyword followed by an

index number of the spectrum.

Arguments: multiplier is a value to multiply each spectrum being added to the add/

subtract experiment (exp5). The normal range of multiplier would be +1  $\,$ 

to -1 but the range is actually unlimited. The default is 1.0.

shift is the number of data points to shift each spectrum. A positive value shifts the spectrum being added to a higher frequency, or to the left. A negative value shifts the spectrum to a lower frequency, or to the right. The default is 0.

'new' is a keyword to create a new spectrum in the add/subtract experiment.

'trace' is a keyword to select the spectrum given by the index number argument (index) and add it to the add/subtract experiment. The default is to add to the first spectrum in the add/subtract experiment.

index is the index number of the spectrum to be used as a target in a multielement add/subtract experiment.

Examples: spadd

spadd(.5,25)
spadd('new')
spadd('trace',2)

See also: NMR Spectroscopy User Guide

Related: add Add current FID to add/subtract experiment (C)

addi Start interactive add/subtract mode (C)

clradd Clear add/subtract experiment (C)

ds Display a spectrum (C)
jexp Join existing experiment (C)

select Select a spectrum without displaying it (C)

spmin Take minimum of two spectra in add/subtract experiment (C)
spsub Subtract current spectrum from add/subtract experiment (C)

## spcfrq Display frequencies of rf channels (M)

Description: Displays the parameters sfrq, dfrq, dfrq2, and dfrq3 with seven decimal

points (to nearest 0.1) to provide the exact frequencies of each rf channel. The

number of values displayed depends on numrfch.

Prior to VNMR version 4.3, spcfrq set the frequency of the observe channel.

The parameter sfrq now sets the frequency instead of spcfrq.

See also: NMR Spectroscopy User Guide

 $Related: \hspace{0.5cm} \begin{array}{c} \textbf{dfrq} \end{array} \hspace{0.5cm} Transmitter \ frequency \ of \ first \ decoupler \ (P)$ 

dfrq2 Transmitter frequency of second decoupler (P)
dfrq3 Transmitter frequency of third decoupler (P)

numrfch Number of rf channels (P)
setfrq Set frequency of rf channels

sfrq Transmitter frequency of observe nucleus (P)

## specdc3d 3D spectral dc correction (P)

Description: Sets whether a 3D spectral dc correction occurs. The spectral dc correction is

the last operation to be performed upon the data prior to forming linear combinations of the data, using the coefficients in the 3D coefficient file (coef), and then writing the data to disk. If specdc3d does not exist, it is

created by the macro par3d.

Values: A three-character string selected from 'nnn', 'nny', 'nyn', etc. Each

character may take one of two values: n for no spectral dc correction along the relevant dimension, and y for spectral dc correction along the relevant dimension. The first character refers to the  $f_3$  dimension (sw, np, fn), the second character refers to the  $f_1$  dimension (sw1, ni, fn1), and the third character refers to the  $f_2$  dimension (sw2, ni2, fn2). The default is 'nnn'.

See also: NMR Spectroscopy User Guide

Related: dc Calculate spectral drift correction (C)

fiddc3d 3D time-domain dc correction (P)

fn Fourier number in directly detected dimension (P)
fn1 Fourier number in 1st indirectly detected dimension (P)
fn2 Fourier number in 2nd indirectly detected dimension (P)

ft3d Perform a 3D Fourier transform (M)

ni Number of increments in 1st indirectly detected dimension (P)ni2 Number of increments in 2nd indirectly detected dimension (P)

Number of data points (P)

par3d Create 3D acquisition, processing, display parameters (C)

ptspec3d Region-selective 3D processing (P)

Sw Spectral width in directly detected dimension (P)
Sw1 Spectral width in 1st indirectly detected dimension (P)
Sw2 Spectral width in 2nd indirectly detected dimension (P)

#### spin Submit a spin setup experiment to acquisition (C)

Description: Regulates sample spinning according to the parameter spin, using the

acquisition computer. It also sets rf frequency, decoupler status, and

temperature.

See also: NMR Spectroscopy User Guide

Related: au Submit experiment to acquisition and process data (C)

change Submit a change sample experiment to acquisition (M) ga Submit experiment to acquisition and FT the result (C)

go Submit experiment to acquisition (C)

lock Submit an Autolock experiment to acquisition (C)

Submit change sample, autoshim experiment to acquisition (M)

Submit an Autoshim experiment to acquisition (C)

spin Sample spin rate (P)

Submit a setup experiment to acquisition (M)

#### spin Sample spin rate (P)

Description: Selects a regulated spin rate. The rate is changed when a sample is inserted or

spin, go, ga, au, or sample are entered.

Values: 0 indicates non-spinning operation.

5 to 39 are spinning rates.

'n' leaves the spin rate at the currently used value and does not wait for

regulated spinning before performing acquisition.

See also: NMR Spectroscopy User Guide

Related: au Submit experiment to acquisition and process data (C)

ga Submit experiment to acquisition and FT the result (C)

go Submit experiment to acquisition (C)

Submit change sample, Autoshim experiment to acquisition (M)

sethwSet values for hardware in acquisition system (C)spinSubmit a spin setup experiment to acquisition (C)

#### spincad Run SpinCAD program (C)

Applicability: SpinCAD Software.

Description: Opens the graphical pulse sequence generation utility.

See also: SpinCAD

Related: vnmr2sc VNMR to SpinCAD pulse sequence translator (M)

## spingen Compile SpinCAD pulse sequence (M,U)

Applicability: SpinCAD Software.

```
Syntax: (From VnmrJ)
                 spingen
                  spingen(pulsesequence)
                 spingen<(<option,>pulsesequence<,pulsesequence2>)>
                  spingen('-psg',pulsesequence)
                  spingen('-all',pulsesequence)
                  spingen('-dps',pulsesequence)
                 spingen pulsesequence < pulsesequence2,,>
                 spingen <option> pulsesequence < pulsesequence2,, >
                 spingen -psq pulsesequence
                 spingen -dps pulsesequence
                 spingen -all pulsesequence
     Description:
                 Compiles the SpinCAD pulse sequence. The most common usage is the first one
                 (spingen, with no arguments), which compiles the current pulse sequence.
                 Two or more options to SpinCAD compilation are: (1) '-psq' option:
                 compilation for the acquisition go command (2) '-dps' option: compilation
                 for dps usage and (3) '-all' option: include both of the above options and
                 compilation of any Java programs that the pulse sequence may use.
                 The spingen macro with no arguments does both the go and dps
                 compilations. Individual compilations for go ('-psg' option) and
                 dps ('-dps' option) can also be done (these are rarely used)
                 In case of SpinCAD sequences and C sequences having the same name, the last
                 compiled sequence will be used for the go command. The isspincad macro
                 can be used to check if the current sequence is SpinCAD or of C type.
                 Compilation of a SpinCAD sequence generates two files in the user's seqlib
                 directory, pulsesequence.psg and pulsesequence dps.psg, for
                 every source file pulsesequence. Compiled SpinCAD files are distinct from the
                 C files, in that they have .psq extension in the filenames. Java program files
                 (if used) must reside in ~/vnmrsys/spincad/classes directory. Java
                 programs are compiled and the class files placed in the same ~/vnmrsys/
                 spincad/classes directory. The spingen macro checks for any Java
                 files in /vnmr/spincad/classes directory, if it does not exit in the
                 users's classes directory.
                 Compilation of a SpinCAD sequence differs from the conventional compilation
                 of C sequences; it involves the expansion of any composites used;
                 transformation of parallel events to a format that Jpsg program can resolve.
     Arguments:
                 <no option> - compilations for go and dps
                  -psg - compilation for go only
                  -dps - compilation for dps only
                  -all - compilations for 90, dps, and also compile any Java programs called
                 from the SpinCAD sequence.
        See also:
                SpinCAD
        Related:
                                Display SpinCAD interface (M)
                 spincad
                 Set up a slfreq array (M)
spinll
         Syntax: spinll<('mark')>
     Description: Copies a list of frequencies to the slfreq parameter in iterative spin
                 simulation and runs dla. This macro also clears previous line assignments.
     Arguments:
```

'mark' is a keyword to copy the list of frequencies from the mark1d.out file to slfreq. The default is to copy the frequencies from the last line listing by nll or dll to the slfreq. Use the cursor and the mark button to place the

lines to be assigned in mark1d.out. Enter mark ('reset') to clear the file, and use n1 to move the cursor to the center of a selected line.

See also: NMR Spectroscopy User Guide

Related: dla Display line assignments (M)

dll Display listed line frequencies and intensities (C)
mark Determine intensity of the spectrum at a point (C)

nl Position the cursor at the nearest line (C)
nll Find line frequencies and intensities (C)

slfreq Measured line frequencies (P)

## spinner Open the Spinner Control window (C)

Description: Opens the Spinner Control window. This window has the following capabilities:

- Turn the sample spinner off.
- Turn the sample spinner on at a specified speed, in Hz.
- Enable spinner control from within an experiment using the spin parameter and the spin, go, ga, or au commands. This mode is the default.
- Alternatively, turn off experiment control of the sample spinner and allow only the Spinner Control window (and acqi and sethw) to set the spinning speed. This mode has the advantage that, often times, the spin parameter is different between experiments. Joining a different experiment and entering go can unexpectedly change the spinning speed. This alternate mode prevents this problem. In this mode, when a go, su, ga, or au is entered, the spin parameter is first set to the speed selected in the Spinner Control window and then the spin parameter is set to "Not Used."
- Select the style of spinner: low-speed style or a high-speed style. If the high-speed style of spinner (used for solids) is selected, the choice of setting the spinning speed or the air flow rate is provided. Setting the air flow rate is useful when setting up the solids spinning apparatus.

If the spinning speed is controlled only through the Spinner Control window, the action to be taken after a spinner error can be selected:

- Display a warning but continue acquisition.
- Stop acquisition and display a warning.

If experiment control of spinning speed is selected, these selections are faded because they are inoperative, and the selection of the action to be taken after a spinning speed error is provided by the parameter in.

See also: NMR Spectroscopy User Guide

Related: acqi Interactive acquisition display process (C)

auSubmit experiment to acquisition and process data (C)changeSubmit a change sample experiment to acquisition (M)gaSubmit experiment to acquisition and FT the result (C)

go Submit experiment to acquisition (C)

in Lock and spin interlock (P)

lock Submit an Autolock experiment to acquisition (C)

Submit change sample, autoshim experiment to acquisition (M)

sethw Set values for hardware in acquisition system (C)
shim Submit an Autoshim experiment to acquisition (C)

spin Sample spin rate (P)

Submit a setup experiment to acquisition (M)

#### Perform spin simulation calculation (C) spins

Syntax: spins<(options)>

Description: Performs a spin simulation, using the current spin system parameters. Refer to the description of spsm for setting up the parameters. Use dsp to display the spectrum resulting from the simulation. The output file is spins.list in the current experiment. This file includes the calculated transitions ordered by frequency.

> Line assignments are required for the iteration. These consist of a list of observed frequencies, which is stored in the arrayed parameter slfreq, and the line assignments stored in the array clindex. spinll copies the frequencies from the last line listing by nll or dll into the parameter slfreq. The line listing can be from an observed spectrum or from the results of deconvolution. After spinll, line assignments are most easily made by entering assign. dla displays the assignments. Single assignments can also be made by assign (transition number, line number), where transition number is the index of a transition and line number is the index of the measured line. Setting the line number argument to 0 deletes assignments. dla('long') produces an expanded display of assignments.

> Be aware that spin simulation line numbers and line list line numbers **are not the same.** Conventional line lists produced by dll number the lines from left to right (low- to high-field). The spin simulation software numbers lines according to a more complicated scheme, and these numbers are rarely if ever in frequency order.

> The parameters to be iterated are chosen by setting the string parameter iterate (e.g, iterate = 'A, B, JAB'). If several parameters have the same value due to symmetry, use iterate='A,B,C,JAB,JAC=JAB'. This string sets the iterated parameter JAC to JAB during the iteration. JAB must be defined as an iterated parameter in the string before it can be used at the right side of the equal sign. Sets of parameters with up to six members may be set up in this way. The member in the set that is used on the right side of the equal sign must always come first in the parameter display (e.g., JAB=JAC would be wrong). A parameter is held constant during iteration if it is not included in the iterate string.

> The command initialize iterate sets iterate to iterate all spins not named X, Y, or Z and the associated coupling constants.

> Following an iterative spin simulation, dga displays the new values of the coupling constants and chemical shifts. undospins restores a spin system as it was before the last iterative run. It returns the chemical shifts, coupling constants, and line assignments, making it possible to continue from this state with modified line assignments.

Note that major changes in the starting values of parameters may change the numbering of the energy levels and hence the line numbers. The line assignments would then be incorrect and would have to be reentered.

For a successful iteration, it is often necessary to keep some parameters fixed. For example, it is sometimes useful to alternately iterate couplings and shifts, keeping one group fixed while the other is iterated independently.

Arguments: The following variations of spins are available:

- spins ('calculate', 'energy') puts an energy-level table in the output file.
- spins('calculate', 'transitions') puts a second table of transitions ordered by transition number in the output file.
- spins ('display') and dsp are equivalent.
- spins('system','spinsystemname') and spsm('spinsystemname') are equivalent.

- spins('iterate') runs interactively to match experimental and calculated lines.
- spins('iterate', 'iteration') lists parameters after each iteration in the output file.
- spins('iterate'<, options>) provides for determining the chemical shifts and coupling constants to produce a spectrum that matches a table of observed lines. spins iterates until the rms (root-mean-square) error of the line matching meets a built-in test, unless it first reaches the value given by number\_iterations. Iteration also stops if the rms error increases.
- Put multiple list options into the second argument, separated by a blank (e.g., spins ('calculate', 'transitions energy')).

Examples: spins

spins('calculate','energy')

spins('iterate')

See also: NMR Spectroscopy User Guide

Related: assign Assign transitions to experimental lines (M)

clindex Index of experimental frequency of a transition (P)
dga Display parameter groups (spin simulation) (C)

dla Display line assignments (M)

dll Display listed line frequencies and intensities (C)

dsp Display calculated spectrum (C)

initialize\_iterate Set iterate to contain relevant parameters (M)

iterate Parameters to be iterated (P)
niter Number of iterations (P)

nll Find line frequencies and intensities (C)

slfreqMeasured line frequencies (P)spinl1Set up slfreq array (M)spsmEnter spin system (M)

undospins Restore spin system as before last iterative run (M)

## split Split difference between two cursors (M)

Description: Repositions the left-hand cursor halfway between its original position and the

position of the other cursor. This macro is very useful for finding the center of a powder pattern: place the two cursors on the horns of the pattern and then enter

split to give the center.

See also: NMR Spectroscopy User Guide

Related: delta Difference of two frequency cursors (P)

## spintype Spinner Type ((P)

Description: This global parameter determines which spinner hardware is used.

Values: 'liquids' for low speed spinning of 5 and 10 mm liquids samples

'tach' for high speed spinning of 5 and 7 mm Jacobsen probes

'mas' for high speed spinning using standalone spinner

'nano' for spinning of nano probes

'none' for no spinner controller is present, e.g. imaging

#### spmax Take the maximum of two spectra (C)

Description: Takes the maximum of two spectra, considered point-by-point in an absolute-

value sense. For example, if the two corresponding values are -2 and +3, the spmax spectrum will have +3; if the two values are +2 and -3, the spmax

spectrum will have -3 at that point.

## spmin Take minimum of two spectra in add/subtract experiment (C)

Description: Takes the minimum of two spectra, considered point-by-point in an absolute-

value sense. For example, if the two corresponding values are -2 and +3, the spmin spectrum will have -2; if the two values are +2 and -3, the spmin

spectrum will have +2 at that point.

The function of spmin is to essentially select for common features within two spectra while eliminating features that are not common between them. In particular, if two CP/MAS spectra are obtained at different spin rates, the peaks stay in the same place (and hence the spmin spectrum also contains the same peaks), but the sidebands move. If spectrum 1 has baseline where spectrum 2 has sideband, and spectrum 2 has baseline where spectrum 1 has sideband, then the spmin spectrum will contain only baseline in these regions, eliminating the

spinning sidebands.

See also: NMR Spectroscopy User Guide

Related: addi Start interactive add/subtract mode (C)

spadd Add current spectrum to add/subtract experiment (C)
spsub Subtract current spectrum from add/subtract experiment (C)

#### spsm Enter spin system (M)

Syntax: spsm(spin system)

Description: Enables entry of the spin system for spin simulation and creates and initializes

the appropriate parameters to describe the various chemical shifts and coupling constants. Chemical shifts can be entered for the X-nucleus, and the spectrum is calculated if that shift is in the window. Generally, however, it is not necessary to enter the X-nucleus chemical shift, and its value has no effect on the spectrum

of the remainder of the spin system.

Arguments: spin system is an alphanumeric string of upper-case letters for chemical

shift and coupling constant parameters. Chemical shifts are stored in parameters A through Z, and the coupling constants are stored in the parameters starting with JAB and ending with JYZ. Different nucleus types are handled by using letters starting with A for the first type, X for the second, and M for the third. Once created, these parameters are entered and modified in the usual way (e.g., A=78.5 JAC=5.6). Entry of chemical shifts in ppm is entered by using sfrq

(e.g., B=7.5\*sfrq).

Examples: spsm('AB')

spsm('A3B2')
spsm('AB2CMXY')

See also: NMR Spectroscopy User Guide

Related: sfrq Transmitter frequency of observe nucleus (P)

spins Perform spin simulation calculation (C)

#### spsub Subtract current spectrum from add/subtract experiment (C)

Syntax: (1) spsub< (multiplier<, shift>) >

(2) spsub('new')

(3) spsub('trace', index)

Description: Performs non-interactive spectral subtraction. The last displayed or selected

spectrum is subtracted from the current contents of the add/subtract experiment (exp5). A multi-element add/subtract experiment can be created using the 'new' keyword. Individual spectra in a multi-element add/subtract experiment can be subsequently subtracted from using the 'trace' keyword followed by

an index number of the spectrum.

Arguments: multiplier is a value to multiply each spectrum being subtracted from the

add/subtract experiment (exp5). The normal range of  ${\tt multiplier}$  would be

+1 to -1 but is actually unlimited. The default is 1.0.

shift is the number of data points to shift each spectrum. A positive value shifts the spectrum being added to a higher frequency, or to the left. A negative value shifts the spectrum to a lower frequency, or to the right. The default is 0.

'new' is a keyword to create a new spectrum in the add/subtract experiment.

'trace' is a keyword to select the spectrum given by the index number argument (index) and subtract it from the add/subtract experiment. The default is to subtract from the first spectrum in the add/subtract experiment.

index is the index number of the spectrum to be used as a target in a multi-

element add/subtract experiment.

Examples: spsub

spsub(.5,25)
spsub('new')
spsub('trace',2)

See also: NMR Spectroscopy User Guide

Related: clradd Clear add/subtract experiment (C)

ds Display a spectrum (C)
jexp Join existing experiment (C)

spadd Add current spectrum to add/subtract experiment (C)

select Select a spectrum without displaying it (C)

spmin Take minimum of two spectra in add/subtract experiment (C) sub Subtract current FID from add/subtract experiment (C)

#### sqcosine Set up unshifted cosine-squared window function (M)

Syntax: sqcosine<(<t1 inc><,t2 inc>)>

Description: Sets up an unshifted cosine-squared window function in 1, 2, or 3 dimensions.

The macro checks whether the data is 1D, 2D, and 3D.

Arguments: t1 inc is the number of t1 increments. The default is ni.

t2 inc is the number of t2 increments. The default is ni2.

See also: NMR Spectroscopy User Guide

Related: gaussian Set up unshifted Gaussian window function (M)

niNumber of increments in 1st indirectly detected dimension (P)ni2Number of increments in 2nd indirectly detected dimension (P)pi3ssbsqSet up pi/3 shifted sinebell-squared window function (M)pi4ssbsqSet up pi/4 shifted sinebell-squared window function (M)sqsinebel1Set up unshifted sinebell-squared window function (M)

sqdir Study queue directory (P)

Description: Specifies the full path directory where a study is stored. It is set when a new

study is created.

See also: NMR Spectroscopy User Guide, VnmrJ Walkup, VnmrJ Imaging User Guide

Related: autodir Automation directory absolute path (P)

globalauto Automation directory name (P)

save Save data (M)

sqname Study queue parameter template (P)
startq Start a chained study queue (M)

studyid Study identification (P)

sqname Study queue parameter template (P)
xminit Initialize an imaging study queue (M)

sqend End a study queue (M)

Description: End a study queue. Usually called by other macros, and not used from the

command line.

Related: sqfilemenu Study queue file menu commands (M)

sqexp Load experiment from protocol (M)

Applicability: Imaging

Description: Macro to load an experiment from a protocol.

Syntax: sqexp(experiment <, 'save'>)

The first argument is the name of the experiment, and is required. The second argument is an optional keyword 'save'. If specified, it first saves parameter changes to the current experiment in the study queue before loading the

parameters for the new experiment.

Examples: sqexp('epidw')

sqexp('spuls','save')

See also: VnmrJ Imaging User Guide

Related: apptype Application type (P)

execpars Set up the exec parameters (M)

sqfilemenu Study queue file menu commands (M)

Description: A macro to perform commands for the study queue operation. Usually the

macro is called from the study queue file menu located below the study queue

area, and not from the command line.

See also: VnmrJ Imaging User Guide

Related: cqinit Initialize liquids study queue (M)

cgreset Reset study queue parameters (M)

sqend End a study queue (M)

sqreset Reset study queue parameters for imaging (M)

xminit Initialize an imaging study queue (M)

sqmode Study queue mode (P)

Description: A global parameter that specifies the study queue mode. It is used to determine

if the study queue acquisition is chained or not.

See also: NMR Spectroscopy User Guide, VnmrJ Walkup

Related: start a chained study queue (M)

xmnext Find next prescan or next experiment in study queue (M)

xmwexp Processing macro for end of acquisition in study queue (M)

#### sgname Study queue parameter template (P)

Description: Stores a string in the global tree that determines where a study is stored. It is set

from the *Save data setup* dialog in the *Utilities* menu. Dollar signs (\$) are used to delimit a string to search for a parameter to be used in the study file name. Percent signs (%) are used to delimit a numeric extension, e.g. %Rn%, or time specifications. Strings from the sampleinfo file are not used, since studies are created in foreground, not automation. Text not delimited by dollar signs or percent signs is copied from sqname without any changes.

If sqname does not start with a slash mark (/), the study is stored in the path given by autodir or globalauto; otherwise the name is used as is. A revision number is automatically appended. Values: If sqname is a null string, it defaults to %R2%, and the resulting study id is a two-digit revision number. The resulting path and file name must be accessible (with read-write

permission) by that user.

Examples: sqname='s %DATE% %R3%' studyid='s 20040501 001'

sqname='s\_\$loc\$\_' studyid='s\_7\_01'
sgname='r\$vrack\$z\$vzone\$/well\$loc\$%R0%'

studyid='r1z3/well16'

See also: NMR Spectroscopy User Guide, VnmrJ Walkup

Related: autodir Automation directory absolute path (P)

autoname Prefix for automation data file (P)
globalauto Automation directory name (P)
sqdir Study queue directory (P)

sqname Study queue parameter template (P)

studyid Study identification (P)

Svfname Create path for data storage (C)

#### sgpars Create study queue parameters for imaging (M)

Applicability: Imaging

Description: A macro to create study queue parameters for imaging. Usually called by other

macros, and not used from the command line.

See also: VnmrJ Imaging User Guide

Related: fixpar Correct parameter characteristics in experiment (M)

## sqprotocol Macro to create protocols (M)

Applicability: Imaging

Description: A macro to create protocols for imaging applications. Called by the Make

protocols dialogs in the Utilities menu.

#### sqreset Reset study queue parameters for imaging (M)

Applicability: Imaging

Description: Reset study queue parameters for imaging. Usually called by other macros, and

not used from the command line.

sqrt Return square root of a real number (O)

Description: A operator in MAGICAL programming that returns the square root of a real

number. A negative argument to sqrt is evaluated to 0.0. Operator is not used

from the command line.

Examples: a = sqrt(b)

See also: User Programming

Related asin Find arc sine of number (C)

Find arc tangent of a number (C)
Find cosine value of an angle (C)
Find exponential value (C)

In Find natural logarithm of a number (C)tan Find tangent value of an angle (C)trunc Truncates real numbers (O)

typeof Return identifier for argument type (O)

sqsavestudy Macro to save study parameters for imaging (M)

Applicability: Imaging

Description: A macro to save study parameters in the imaging study queue. Usually called

by other macros, and not used from the command line.

See also: VnmrJ Imaging User Guide

Related: acquire Acquire data (M)

sqendEnd a study queue (M)studyparStudy parameters (P)

sqsinebell Set up unshifted sinebell-squared window function (M)

Syntax: sqsinebell<(<tl inc><,t2 inc>)>

Description: Sets up an unshifted sinebell-squared window function in 1, 2, or 3 dimensions.

The macro checks whether the data is 1D, 2D, and 3D.

Arguments: t1 inc is the number of t1 increments. The default is ni.

t2 inc is the number of t2 increments. The default is ni2.

See also: NMR Spectroscopy User Guide

Related: gaussian Set up unshifted Gaussian window function (M

niNumber of increments in 1st indirectly detected dimension (P)ni2Number of increments in 2nd indirectly detected dimension (P)pi3ssbsqSet up pi/3 shifted sinebell-squared window function (M)pi4ssbsqSet up pi/4 shifted sinebell-squared window function (M)sqcosineSet up unshifted cosine-squared window function (M)

srate Spinning rate for magic angle spinning (P)

Applicability: Systems with solids module.

Description: Set to the spinning speed for magic angle spinning (MAS). srate must be

correct for the pulse sequence set up by xpolar1 to run TOSS or dipolar dephasing correctly. If hsrotor='y', the measured spinning speed is

reported in srate for systems that have rotor synchronization.

Values:  $0 \text{ to } 10^7, \text{ in Hz.}$ 

See also: NMR Spectroscopy User Guide

Related: hsrotor Display rotor speed for solids operation (P)

xpolar1 Set up parameters for XPOLAR1 pulse sequence (M)

sread Read converted data into VnmrJ (C)

Syntax: sread(file<,template>)

Description: Reads 32-bit data files into VnmrJ. For Bruker data files in the AMX and AM

formats, each file must first be converted using the convertbru command

before sread can read the data in the file into VnmrJ.

Arguments: file is the name of a file containing data converted using convertbru.

template is the full path of a parameter template file, but without appending the .par extension on the file name. The default is bruker.par. If no parameter template is specified and bruker.par cannot be found in the user

or system parlib directory, sread aborts with an error message.

Examples: sread('brudata.cv','/vnmr/parlib/bruker')

See also: NMR Spectroscopy User Guide

Related: convertbru Convert Bruker data (M,U)

srof2 Calculate exact rof2 value for Cold Probes (M)

Applicability: Systems with Varian, Inc. Cold Probes

Description: Calculates the exact value needed for rof2 to result in a lp=0 condition for

the given sw. Works with either dsp='r' and fsq='y' or with dsp='i'.

Not compatible with gcomp.

Related: dsp Type of DSP for data acquisition (P)

rof2 Receiver gating time following pulse (P)

ss Steady-state transients (P)

Description: Sets the number of complete executions of the pulse sequence not accompanied

by data collection prior to the acquisition of the real data (sometimes known as *dummy scans*). If ss is positive, ss steady-state transients are applied on the first increment only, and if ss is negative, -ss steady-state transients are

applied at the start of each increment.

Values: 'n', -32768 to 32767

See also: NMR Spectroscopy User Guide; User Programming

ssecho Set up solid-state echo pulse sequence (M)

Applicability: Systems with a solids module.

Syntax: ssecho

Description: Converts a standard two-pulse experiment to a ready-to-run solid-state NMR

echo (SSECHO) pulse sequence.

See also: NMR Spectroscopy User Guide

ssechol Set up parameters for SSECHO1 pulse sequence (M)

Applicability: System with a wideline solids module.

Description: Sets up a parameter set for the quadrupole echo pulse sequence SSECHO1.

See also: NMR Spectroscopy User Guide

ssfilter Full bandwidth of digital filter to yield a filtered FID (P)

Description: Specifies the full bandwidth of the digital filter applied to the original FID to

yield a filtered FID for solvent subtraction. If ssfilter does not exist in the

current experiment, enter addpar('ss') to add it. The command addpar('ss') creates additional time-domain solvent subtraction parameters ssfilter, sslsfrq, ssntaps, and ssorder.

Values: 'n', 10 to sw/2, in steps of 0.1 Hz. The default is 100 Hz.

If ssfilter is set to a value and ssorder is set to some value, the zfs (zero-

frequency) option of solvent subtraction is selected.

If ssfilter is set to 'n', ("Not Used"), both the lfs (low-frequency

suppression) and zfs options are turned off.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

Fourier transform 1D data (C)

parfidss Create parameters for time-domain solvent subtraction (M)

ssntapsNumber of coefficients in the digital filter (P)sslsfrqCenter of solvent-subtracted region of spectrum (P)ssorderOrder of polynomial to fit digitally filtered FID (P)swSpectral width in directly detected dimension (P)wftWeight and Fourier transform 1D data (C)

sslsfrq Center of solvent-suppressed region of spectrum (P)

Description: Specifies the location of the center of the solvent-suppressed region of the

spectrum. If sslsfrq does not exist in the current experiment, enter

addpar ('ss') to add it. addpar ('ss') also creates time-domain solvent

subtraction parameters ssfilter, ssntaps, and ssorder.

Values: 'n' (or 0) specifies solvent suppresses a region centered about the transmitter

frequency. This is the default

Non-zero value shifts the solvent-suppressed region by sslsfrq Hz. Multiple regions may be suppressed by arraying the value of sslsfrq. Up to 4 values

are allowed.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

parfidss Create parameters for time-domain solvent subtraction (M)
ssfilter Full bandwidth of digital filter to yield a filtered FID (P)

Sorder

Number of coefficients in the digital filter (P)

Sorder of polynomial to fit digitally filtered FID (P)

ssntaps Number of coefficients in digital filter (P)

Description: Specifies the number of taps (coefficients) to be used in the digital filter for

solvent subtraction. If ssntaps does not exist in the current experiment, enter addpar ('ss') to add it. addpar ('ss') also creates time-domain solvent

subtraction parameters ssfilter, sslsfrq, and ssorder.

Values: Integer from 1 to np/4. The default is 121. An odd number is usually best.

The more taps in a filter, the flatter the passband response and the steeper the transition from passband to stopband, giving a more rectangular filter.

For the lfs (low-frequency suppression) option, the default is suitable.

For the zfs (zero-frequency suppression) option, a value between 3 and 21 usually works better.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

ft Fourier transform 1D data (C)

Number of increments in 1st indirectly detected dimension (P)

np Number of points (P)

parfidssCreate parameters for time-domain solvent subtraction (M)ssfilterFull bandwidth of digital filter to yield a filtered FID (P)sslsfrqCenter of solvent-suppressed region of spectrum (P)ssorderOrder of polynomial to fit digitally filtered FID (P)

wft Weight and Fourier transform 1D data (C)

## ssorder Order of polynomial to fit digitally filtered FID (P)

Description: Specifies the order of the polynomial to fit the digitally filtered FID if the zfs

(zero-frequency suppression) option is selected for solvent subtraction. ssorder is not used if the lfs (low-frequency suppression) option is selected. If ssorder does not exist in the current experiment, enter addpar ('ss') to add it. addpar ('ss') also creates time-domain solvent subtraction

parameters ssfilter, sslsfrq, and ssntaps.

The solvent subtraction option (zfs or lfs) is selected as follows:

• If ssorder and ssfilter are both set to values, zfs is selected.

• If ssorder='n' and ssfilter is set to a value, lfs is selected.

• If ssorder='n' and ssfilter='n', zfs and lfs are both turned off.

Values: 'n', integer from 1 to 20. The default is 'n'.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

parfidss Create parameters for time-domain solvent subtraction (M)
ssfilter Full bandwidth of digital filter to yield a filtered FID (P)
sslsfrq Center of solvent-suppressed region of spectrum (P)
Ssntaps Number of coefficients in the digital filter (P)
wft Weight and Fourier transform 1D data (C)

## stack Stacking mode for processing and plotting arrayed spectra (M)

Syntax: stack(mode)

Description: When processing and plotting arrayed 1D spectra, VnmrJ automatically

determines if the *stacking mode* is horizontal, vertical or diagonal from the number of traces and the number of lines in the spectrum. If you do not want this automatic function (or it makes an undesirable decision), you can override it by placing the stack macro in the experiment startup macro or by calling

stack before processing (or reprocessing) a spectrum. The macro

autostack switches back to automatic determination of the stack mode by

destroying the parameter stackmode.

Arguments: mode is one of the stacking modes 'horizontal', 'vertical', or

'diagonal'.

See also: NMR Spectroscopy User Guide

 $Related: \quad \text{autostack} \qquad \text{Automatic stacking for processing and plotting arrays } (M)$ 

procarray Process arrayed 1D spectra (M)

plarray Plot arrayed 1D spectra (M)
stackmode Stacking control for processing (P)

stackmode Stacking control for processing arrayed 1D spectra (P)

Description: Controls whether stacking for processing arrayed 1D spectra is automatic or

nonautomatic. The *automatic stacking mode* can be overridden by creating and setting stackmode in the startup macro or before calling procplot or procarray. The autostack macro switches back to automatic

determination of the stack mode by destroying this parameter.

Values: 'horizontal', 'vertical', or 'diagonal'.

See also: NMR Spectroscopy User Guide

Related: autostack Automatic stacking for processing and plotting arrays (M)

procarray Process arrayed 1D spectra (M)
procplot Automatically process FIDs (M)

stack Fix stacking mode for processing and plotting arrayed spectra (M)

startq Start a chained study queue (M)

Description: Start a chained acquisition for a study queue.

Related: sqmode Study queue mode (P)

Find next prescan or next experiment in study queue (M)

status Display status of sample changer (C,U)

Applicability: Systems with an automatic sample changer.

Syntax: status<(directory<,config file>)>

(From UNIX) status directory <config file>

Description: Displays a status window with a summary of all experiments and a scrollable

list of individual experiments. Individual experiments are selected by clicking anywhere on the experiment of interest. status updates as the state of an automation run changes. If an experiment finishes or a new experiment is

added, the status display is updated.

Arguments: directory is the path to the directory where the done queue (doneQ) is

stored. In the UNIX shell, a directory path is required. In VnmrJ, a directory

path is optional. The default is the automation mode directory.

config file is the name of a user-supplied file that customizes status for

local use. Refer to the manual *User Programming* for details.

Examples: (From VnmrJ) status

(From VnmrJ) status('/home/vnmr1/AutoRun 621')

(From UNIX) status /home/vnmr1/AutoRun\_621 mystatus

See also: VnmrJ Walkup; User Programming

 $Related: \quad \begin{array}{ll} \textbf{autodir} & \quad & Automation \ directory \ absolute \ path \ (P) \end{array}$ 

autoname Prefix for automation data file (P)

enter Enter sample information for automation run (C,U)

stdld Apptype macro for Standard 1D experiments (M)

Applicability: Liquids

Description: Perform the actions for Standard 1D protocols to set up, process, and plot

experiments.

See also: NMR Spectroscopy User Guide, VnmrJ Walkup

Related: apptype Application type (P)

execpars Set up the exec parameters (M)

#### stdshm Interactively create a method string for autoshimming (M)

Syntax: stdshm

Description: Creates a method string to be used in adjusting the spinning controls z1, z2,

z3, and z4 when a sample is changed. If non-spin controls also need adjusting,

further shimming operations are required.

The method string is constructed in answer to questions about the sample length, the time available for shimming, and the solvent  $T_1$  or, in FID shimming, the  $T_1$  of the sample. In asking about sample height, stdshm assumes that z3 and z4 need adjusting only with short samples; therefore, select "sample height will vary" if z3 and z4 shimming is definitely wanted.

Try lock shimming first to see if it produces a satisfactory result. Lock shimming requires a much shorter shimming time than FID shimming and usually adjusts z1 and z2 just as well. If lock shimming is unsatisfactory, try FID shimming. Again, when z3 and z4 adjustment is required, lock shimming is faster, but FID shimming is more effective. stdshm displays the estimated shimming time, permitting revision when the time is too long.

To shim after running stdshm, enter method='std' (for lock shimming) or method='fidstd' (for FID shimming). Then enter shim or set the wshim parameter to shim before the start of acquisition.

Note that the command  ${\color{red} {\tt newshm}}$  is much like  ${\tt stdshm}$  but that  ${\color{red} {\tt newshm}}$ 

provides more flexibility in making method strings

See also: NMR Spectroscopy User Guide

Related: dshim Display a shim method string (M)

method Autoshim method (P)

newshm Interactively create a shim method with options (M) shim Submit an Autoshim experiment to acquisition (C) wshim Conditions when shimming is performed (P)

## sth Minimum intensity threshold (P)

Description: Intensity threshold above which transitions are printed and included in the

simulated spectrum. Transitions whose intensity falls below this threshold are

omitted from the simulation.

Values: 0 to 1.00. A typical value is 0.05. See also: *NMR Spectroscopy User Guide* 

Related: spins Perform spin simulation calculation (C)

Enter spin system (M)

th Threshold (P)

#### string Create a string variable (C)

Syntax: string(variable)

Description: Creates a string variable without a value.

Arguments: variable is the string variable to be created.

Examples: string('strvar1')

See also: User Programming

strtext Starting point for LP data extension in np dimension (P)

Description: Specifies inclusively the complex time-domain data point at which LP (linear

prediction) data extension (alteration) is to begin in the np dimension. Enter addpar('lp') to create strtext and other np dimension LP parameters

in the current experiment.

Values: 1 to np/2

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

lpalg LP algorithm in np dimension (P)

np Number of data points (P)

strtlp Starting point for LP calculation in np dimension (P)

strtext1 Starting point for LP data extension in ni dimension (P)

Description: Specifies inclusively the complex time-domain data point at which LP (linear

prediction) data extension (alteration) is to begin in the ni dimension. Enter addpar('lp',1) to create strtext1 and other ni dimension LP

parameters in the current experiment.

Values: 1 to ni/2

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

lpalg1 LP algorithm in ni dimension (P)

Number of increments in 1st indirectly detected dimension (P)

strtlp1 Starting point for LP calculation in ni dimension (P)

strtext2 Starting point for LP data extension in ni2 dimension (P)

Description: Specifies inclusively the complex time-domain data point at which LP (linear

prediction) data extension (alteration) is to begin in the ni2 dimension. Enter addpar('lp',2) to create strtext2 and other ni2 dimension LP

parameters in the current experiment.

Values: 1 to ni2/2

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

lpalg2 LP algorithm in ni 2 dimension (P)

Number of increments in 2nd indirectly detected dimension (P)

Starting point for LP calculation in ni2 dimension (P)

strtlp Starting point for LP calculation in np dimension (P)

Description: Specifies the first complex, time-domain data point to be used in calculating the

complex linear prediction (LP) coefficients in the np dimension. If

lpopt='b', the strtlp-th complex time-domain data point and the ensuing
(2\*lpfilt-1) data points are used in this calculation. If lpopt='f', the

strtlp-th complex time-domain data point and the preceding (2\*lpfilt-1) data points are used in this calculation. Enter

addpar('lp') to create strtlp and other np dimension LP parameters in

the current experiment.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

lpalg LP algorithm in np dimension (P)

lpfilt LP coefficients to calculate in np dimension (P)
lpnupts LP number of data points in np dimension (P)
lpopt LP algorithm data extension in np dimension (P)
strtext Starting point for LP data extension in np dimension (P)

#### strtlp1 Starting point for LP calculation in ni dimension (P)

Description: Specifies the first complex, time-domain data point to be used in calculating the

complex linear prediction (LP) coefficients in the  $\tt ni$  dimension. It functions analogously to  $\tt strlp$ . Enter  $\tt addpar('lp',1)$  to create  $\tt strlp1$  and

other ni dimension LP parameters in the current experiment.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

lpalg1 LP algorithm in ni dimension (P)

lpfilt1LP coefficients to calculate in ni dimension (P)lpnupts1LP number of data points in ni dimension (P)lpopt1LP algorithm data extension in ni dimension (P)strtext1Starting point for LP data extension in ni dimension (P)

#### strtlp2 Starting point for LP calculation in ni2 dimension (P)

Description: Specifies the first complex, time-domain data point to be used in calculating

complex linear prediction (LP) coefficients in the ni2 dimension. strtlp2 functions analogously to strlp. Enter addpar('lp',2) to create strtlp2 and other ni2 dimension LP parameters in the current experiment.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

lpalg2 LP algorithm in ni2 dimension (P)

lpfilt2 LP coefficients to calculate in ni2 dimension (P)
lpnupts2 LP number of data points in ni2 dimension (P)
lpopt2 LP algorithm data extension in ni2 dimension (P)
strtext2 Starting point for LP data extension in ni2 dimension (P)

#### studyid Study identification (P)

Applicability: Liquids

Description: Specifies the relative directory where a study is stored. In Walkup, it is relative

to autodir. In imaging, it is relative to globalauto; It is set when a new

study is created.

See also: NMR Spectroscopy User Guide, VnmrJ Walkup

Related: autodir Automation directory absolute path (P)

globalauto Automation directory name (P)
sqdir Study queue directory (P)

sqname Study queue parameter template (P)

## studypar Study parameters (P)

Applicability: Liquids, Imaging

Description: A global parameter that contains the list of parameters saved with a study. If the

parameter does not exist, it is created by cgsavestudy for liquids or

sqsavestudy for imaging when a study is saved.

See also: NMR Spectroscopy User Guide, VnmrJ Walkup, VnmrJ Imaging User Guide

Related: cqsavestudy Macro to save study queue parameters (M)

sqsavestudy Macro to save study parameters for imaging (M)

studystatus Study status (P)

Applicability: VnmrJ Walkup

Description: The status of a study for a sample. The status is set from the status of the

experiments within the study by the macro cqsavestudy.

See also: VnmrJ Walkup

Related: cqsavestudy Macro to save study queue parameters (M)

studytime Study time (P)

studytime Study time (P)

Applicability: Walkup

Description: The total time it takes to run a study. It is set by the xmt ime macro when a study

is created.

See also: VnmrJ Walkup

Related: xmsubmit Submit sample(s) to the study queue (M)

wmtime Update the study queue time (M)

#### su Submit a setup experiment to acquisition (M)

Description: Sets up the system hardware to match the current parameters but does not

initiate data acquisition. Typical uses of su are to change the system frequency in preparation for probe tuning, to change the sample temperature in advance of beginning an experiment (or after a variable temperature experiment is run), and to turn the decoupler on or off. If load='y', su can be used to set shim values. su also sets lock parameters (lockpower, lockgain, lockphase) and the field offset parameter (z0).

su does *not* delete any existing data in the current experiment (only go, ga, and au do that). Everything that su does is also done by go, ga, and au.

Shim DAC values are automatically loaded when the acquisition system boots up; if the acquisition system has been recently rebooted, su must be entered before acqi or gtune can be run.

See also: NMR Spectroscopy User Guide

Related: acqi Interactive acquisition display process (C)

au Submit experiment to acquisition and process data (C) change Submit a change sample experiment to acquisition (M) ga Submit experiment to acquisition and FT the result (C)

go Submit experiment to acquisition (C) load Load status of displayed shims (P)

lock Submit an Autolock experiment to acquisition (C)

lockgainLock gain (P)lockphaseLock phase (P)lockpowerLock power (P)

qtune Tune probe using swept-tune graphical tool (C)

sample Submit change sample, autoshim experiment to acquisition (M)

Submit an Autoshim experiment to acquisition (C)
Spin
Submit a spin setup experiment to acquisition (C)

z 0 Z0 field position (P)

#### sub Subtract current FID from add/subtract experiment (C)

```
Syntax: (1) sub<(multiplier<,'new'>)>
```

(2) sub('new')

(3) sub('trace', index)

#### Description:

Subtracts the last displayed or selected FID from the current contents of the add/subtract experiment (exp5). lsfid and phfid can be used to shift or phase rotate the selected FID before it is subtracted from the data in add/subtract experiment. A multi-FID add/subtract experiment can be created by using the 'new' keyword. Individual FIDs in a multi-FID add/subtract experiment can subsequently be subtracted by using the 'trace' keyword followed by the index number of the FID.

#### Arguments:

multiplier is a value that the FID is to be multiplied by before being subtracted from the add/subtract experiment (exp5). The default is 1.0.

'new' is a keyword to create a new FID element in an add/subtract experiment.

'trace' is a keyword to use the next argument (index) as the number of the FID to subtract from in an add/subtract experiment. The default is to subtract from the first FID in a multi-FID add/subtract experiment.

index is the index number of the FID to be used as a target in a multi-FID add/subtract experiment.

#### Examples: sub

sub(0.75)
sub('new')
sub('trace',2)

See also: NMR Spectroscopy User Guide

Related: add Add current FID to add/subtract experiment (C)

clradd Clear add/subtract experiment (C)

Number of complex points to left-shift ni interferogram (P)

phfid Zero-order phasing constant for np FID (P)
select Select a spectrum without displaying it (C)

spsub Subtract current spectra from add/subtract experiment (P)

#### substr Select a substring from a string (C)

#### Applicability: VnmrJ

## Description:

Picks a substring or word out of a string, replace, or delete a set of characters from a string and returns the result to the string variable \$n1. The position of the first character of the word and the number of characters of the word are returned to \$n2 and \$n3 if these string variables are supplied.

#### Arguments:

'string' string or a string variable.

word\_number is the number of the word to select. Words are counted sequential beginning with the first word of the string as 1.

index is the number of characters counted from the first character of the string or a string variable containing this number.

length is the number of characters in the substring.

new\_string is string or a string variable to replace the contents of string at the position specified by index and length and pass the resulting string to the return string variable.

'delimiter' is a keyword that requires the 'delimiter\_char' argument to specify that the argument that follows specifies the delimiter(s).

'delimiter\_char' is a string of characters to use as delimiters to separate words.

Default delimiters are space and tab " \t".

\$n1 is the return string variable containing the searched for text.

\$n2 is the return variable containing the position of the first character of the word in the string.

\$num is the return string variable containing the number of characters in the
word specified by word\_number and contained within the delimiters.

#### Examples: Search examples:

#### Text substitution examples:

Explicit text substitution and passing the result to the return string variable. substr('abcdefg', 2, 3, '1234'):n1 string n1='a1234efg'

Text substitution in a string variable using results held in return string variable from a previous search. Start with the following text held in a string variable:

```
n1='There are 10 samples to be run' substr(n1,4):n2,$f,$num sets strings n2=samples,$f=14, and $num=7 substr(n1,$f,$num,'experiments'):n3
```

Counts 14 characters (\$f=14) from the beginning of n1, substitutes the word experiments for the 7 character (\$num=7) word in n1, and passes the new string to the return string variable setting

n3='There are 10 experiments to be run'

See also: User Programming

Related: length Determine length of a string (C)
string Create a string variable (C)

#### suselfrq Select peak, continue selective excitation experiment (M)

Syntax: suselfrq

Description: Sets up selective frequency pulse, power, and shape and continue with the

selective excitation experiment. Used by Noesyld, and TOCSYlD.

See also: NMR Spectroscopy User Guide, VnmrJ Walkup

Related: Noesyld Change parameters for NOESY1D experiment (M)

setselinv Set up selective inversion (M)

setselfrqc Select selective frequency and width (M)

TOCSY1D Change parameters for TOCSY1D experiment (M)

## svdat Save data (C)

Syntax: svdat(file<,'f'|'m'|'i'|'b'>)

Description: Outputs current data from the current experiment to a file. Integer data is scaled

when it is written.

Arguments: file is the name of the data file. The file is created in the current directory

VnmrJ is in unless a full directory path is given. If a file of the same name already exists, the user will queried to overwrite the file. If a fully qualified filename is not given, the file will be created in VnmrJ's current directory.

'f'|'m'|'i'|'b' defines how the data is to be written out: 'f' is 32-bit floating point, 'm' or 'i' is 16-bit integer scaled to 12 bits, and 'b' is 8-bit byte integer. The default is 'f'.

Floating point data is not scaled when written.

Integer data is scaled when written. A data value x is scaled as ax+b where:

```
a = (vs*grays1*numgray)/64.0
b = numgray*(0.5-(grays1*grayctr/64.0))
```

where numgray (see below) has a default of 4096 for 'm' and 'i' formats and a default of 256 for the 'b' format, graysl has a default of 1, and grayctr has a default of 32.0.

To scale 16-bit integer data other than 12-bits, the global parameter numgray can be created using create (numgray, real, global) and set to the value  $2^n$ , where n is the number of bits desired. For example, to scale to 15-bits, set numgray=32768.

The display parameters <code>graysl</code> and <code>grayctr</code> are used to save data files for ImageBrowser.

Examples: svdat(rathead, 'b')

See also: *VnmrJ Imaging NMR* 

Related: create Create new parameter in parameter tree (C)

grayctr Gray level window adjustment (P)
grays1 Gray level slope (contrast) adjustment (P)

#### svf Save FIDs in current experiment (M)

Syntax: svf<(file<,'nolog'><,'arch'><,'force'><,'nodb'>)>

Description: Saves parameters, text, and FID data in the current experiment to a file. No data

is removed from the current experiment; svf merely saves a copy of the data in a different file. You can enter rt to retrieve the complete data set, or enter

rtp to retrieve parameters only.

Arguments: file is the name of the file, with the suffix .fid added, to be created to save

the data. The default is the system prompts for a file name. You are warned if you attempt to overwrite a file that already exists. In fact, if data has been acquired with the file parameter set, the data does not need to be saved. It is

already stored in a named file.

'nolog' is a keyword to not save the log file with the data. The default is to save the log file.

'arch' is a keyword to assume that the data goes to a database and appends to the (or creates a) doneQ file with information that can be used by the command status.

If force is given, you are not warned and the older parameter set is removed. nodb is a keyword to prevent svp from adding information to a database. This prevention is useful if temporary parameter files are saved that will soon be removed.

Examples: svf

svf('/home/vnmr1/mydatafile')

See also: NMR Spectroscopy User Guide
Related: file File name (P)

rt Retrieve FID (M)
rtp Retrieve parameters (M)

status Display status of all experiments (C)

## svfdf Save FID data in FDF format (M)

Syntax: svfdf(directory)

Description: Saves raw data from the FID file of the current experiment as an FDF (Flexible

Data Format) file. Data is saved in multiple files, with one trace per file. The files are named fid0001.fdf,fid0002.fdf, etc. The procpar file from

the current experiment is also saved in the same directory.

The FDF file format is described in the manual *User Programming*. Note that the data is complex (FDF type="complex"), and the FDF ordinate = {"intensity", "intensity"}, indicating that each point consists of a pair of intensities. The FDF headers also contain the following special fields:

• nfile gives the sequential number of this file in the series.

• ct is the value of the ct parameter. The data should be divided by ct to give the average signal intensity for one scan.

• scale gives the power of two scaling factor for the data. The data should be multiplied by 2<sup>scale</sup> to give the true values.

Arguments: directory name is the directory in which to store the files. The

extension . dat is appended to the given name.

Examples: svfdf(curexp+'/raw')

See also: User Programming

Related: ct Completed transients (P)

#### svfdir Directory for non-study data (P)

Description: Specifies the directory where data is saved when not using a study in VnmrJ.

See also: NMR Spectroscopy User Guide, VnmrJ Walkup

Related: fidsave Save data (M)

Save data (M)

svfname Filename parameter template for non-study data ((P)

#### Svfname Create path for data storage (C)

Applicability: Automation

Syntax: Svfname: \$path

Svfname(name template):\$path

Svfname(name template, suffix):\$path

 

#### Description:

Determines the name used to store data. This command provides the functionality of the autoname parameter without being in automation mode.

Svfname default naming command with alternate suffixes is svfname and the default directory is svfdir. Svfname does not read a sample info file. A suffix is specified as the second argument. Use a suffix of " to access ordinary files and directories. Arguments used with Svfname are constructed the same way arguments are constructed for autoname.

The name is prefixed with using the value of the parameter autodir or userdir+'/data/' if name\_template is a relative path.

The default suffix is .fid.

#### Arguments:

syfname is default naming parameter.

svfdir is default directory parameter.

name\_template (no quotes) is string that contains keywords separated by substitution specifiers to represent the data storage path. Substitution specifiers in this template are either a percent sign (%) or a dollar sign (\$). The keywords are obtained using % substitution specifiers or VNMR parameters using \$ substitution specifiers.

Percent sign (%) substitution specifier is used to scan for the text specified by keyword between the first percent sign in the template string and the next percent sign. The text specified by the keyword between the % substitution specifiers is passed to \$path.

The following percent substitutions (% keywords) for time and date are obtained from the system clock, not from the sample info file:

Keyword	Format	Description
%DATE%	YYYYMMDD	4-digit year, 2-digit month, 2-digit day
%TIME%	HHMMSS	2-digit hour, 2-digit minute, 2-digit second
%YR%	YYYY	4-digit year
%YR2%	YY	2-digit year
%MO%	MM	2-digit month
%DAY%	DD	2-digit day
%HR%	НН	2-digit hour
%MIN%	MM	2-digit month
%SEC%	SS	2-digit second

Dollar sign (\$) substitution specifier is used with the Svfname command to interpreted a VNMR parameter and substitute the value of this parameter a suffix.

Numeric parameters are truncated and represented as a string with the form: <optional string>parameter value<optional string>. The
name\_template, pw=\$pw\$usec, with vnmr parameter pw having a value
of 12.3 produces pw=12usec01 which is appended to .fid (or .img) and
passed to \$path.

A comma separated excluded suffix list appends a string based on the suffixes and excluded suffixes to the path. Using the keyword 'replacespaces' uses underscores (\_) in place of spaces ' ' in the resulting path name. The keyword 'keepspaces' retains spaces in the resulting path name.

'keepspaces' | 'replacespaces' is an optional argument (includes quotes) that uses either of the following keywords: replacespaces or keepspaces. The argument is accepted if the third argument is a list of suffixes. The action is the same as described for the third argument

Version number is specified by %Rn% where n is an integer from 0 to 9 (default 2), as follows:

Description n=0 no revision digits are appended (all names must be uniquely constructed without these revision digits). 1 to 9 revision number is padded with leading zeroes to form an n-digit number. If more places are needed than specified, more zeroes are used. >9 Rnn is still used as a search string in the sam-(more than pleinfo file. %Rn% must be specified at the end of one digit) the name template string. The revision digits are always appended except if %R0% is used. no %Rn% default of %R2% is used

See also: NMR Spectroscopy User Guide, VnmrJ Walkup

Related: autoname Determines path for data storage during an automation run (C)

autoname Temple determining the path where is data stored (P)

sqname Study queue parameter template (P)
svfname Specifies the filename template (P)

#### svfname Filename parameter template for non-study data ((P)

Description: Specifies the filename template where data is saved when not using a study in

VnmrJ. The template is constructed using the same keywords and delimiter,

dollar sign (\$) and percent sign (%), as autoname.

Examples: If svfdir=userdir+'/data', the result from fidsave is:

svfname='\$pslabel\$ \$tn\$ ' -> userdir+'/data/

Proton H1 01.fid'

svfname='%DATE%/t%TIME%%R0%' -> userdir+'/data/

20040501/t113005.fid'

See also: NMR Spectroscopy User Guide, VnmrJ Walkup

Related: fidsave Save data (M)

Svfname Create path for data storage (C)
sqname Study queue parameter template (P)

Save data (M)

svfname Filename parameter template for non-study data ((P)

#### svp Save parameters from current experiment (M)

Syntax: svp(file) <(file<,'force'><,'nodb'>)>

Description: Saves parameters from current experiment to a file. The parameter set can be

retrieved with the rtp and rt macros. svp reflects any changes made in parameters up to the moment of entering svp, including acquisition parameters

(unlike macro svf).

Arguments: file is the name of the file, with the suffix .par added, to be created to save

the parameters. The default is the system prompts for a file name. You are warned if you attempt to overwrite a parameter set that already exists.

If force is given, you are not warned and the older parameter set is removed. nodb is a keyword to prevent svp from adding information to a database. This prevention is useful if temporary parameter files are saved that will soon be removed.

Examples: svp('/vnmr/stdpar/P31')

svp('/usr/george/testdata')

See also: NMR Spectroscopy User Guide

Related: rt Retrieve FID (M)

rtp Retrieve parameters (M)

Save FIDs in current experiment (M)

## svs Save shim coil settings (C)

Syntax: svs(file)<:status>

Description: Saves all shim coil settings except Z0 to a file.

Arguments: file is the name of a file for saving the shim coil settings. If the file name is

an absolute path, svs uses it with no modifications. Otherwise, svs saves the shim in the first application directory for which it has write permission.

The svs command reports where it stored the shims, unless it is requested to return the status.

status is a return variable with one of the following values after svs finishes:

- 0 indicates sys failed to store shim file.
- 1 indicates svs stored the shim file, either as an absolute path or in the shims directory of the first application directory.
- >=2 indicates svs stored the file in shims directory of the second, third, or later application directory.

Examples: svs('acetone')

svs('bb10mm'):r1

See also: NMR Spectroscopy User Guide

Related: rts Retrieve shim coil settings (C)

#### svs Spin simulation vertical scale (P)

Description: Vertical scale for simulated spectrum.

Values: 0 to 1e10. A typical value is 200.

See also: NMR Spectroscopy User Guide

Related: spins Perform spin simulation calculation (C)

Enter spin system (M)

#### svtmp Move experiment data into experiment subfile (M)

Syntax: svtmp<(file)>

Description: Moves the experiment data (parameters, FID, and transformed spectrum) from

current experiment into a subdirectory inside curexp+'/subexp'. Unlike the macro cptmp, the experiment data is no longer accessible in the current

experiment; only a copy of the parameters is still present.

Arguments: file is the name of the subfile that receives the experiment data. The default

name is either the transmitter nucleus (if seqfil='s2pul') or the pulse

sequence name.

Examples: svtmp

svtmp('cosy')

See also: NMR Spectroscopy User Guide

Related: cptmp Copy experiment data into experiment subfile (M)

curexp Current experiment directory (P)

rttmp Retrieve experiment data from experiment subfile (M)

seqfil Pulse sequence name (P)

# sw Spectral width in directly detected dimension (P)

Description: Sets the total width of the spectrum to be acquired, from one end to the other.

All spectra are acquired using quadrature detection. The spectral width determines the sampling rate for data, which occurs at a rate of 2 \* sw points per second (actually sw pairs of complex points per second). Note that the sampling rate itself is not entered, either directly or as its inverse (known on some systems as the *dwell time*).

If a value of sw is entered whose inverse is not an even multiple of the time base listed above, sw is automatically adjusted to a slightly different value to give an acceptable sampling rate.

To enter a value in ppm, append the character p (e.g., sw=200p).

If a DSP facility is present in the system (i.e., dsp='i' or dsp='r') and

oversampling in the experiment has not been turned off by setting oversamp='n', then the oversampling factor will be recalculated.

Values: Number, in Hz. The range possible is based on the system:

100 Hz to 500 kHz.

solids systems: up to 5 MHz.

See also: NMR Spectroscopy User Guide

Related: dp Double precision (P)

dsp Type of DSP for data acquisition (P)
oversamp Oversampling factor for acquisition (P)
setlp0 Set parameters for zero linear phase (M)

Sw2 Spectral width in 1st indirectly detected dimension (P)
Sw2 Spectral width in 2nd indirectly detected dimension (P)
Sw3 Spectral width in 3rd indirectly detected dimension (P)

# sw1 Spectral width in 1st indirectly detected dimension (P)

Description: Analogous to the sw parameter except that sw1 applies to the first indirectly

detected dimension of a multidimensional data set. The increment of the variable evolution time d2 is automatically calculated from sw1. The number of increments for this dimension is set by ni. To create sw1 in the current

experiment, as well as ni and phase, enter addpar ('2d').

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

Incremented delay in 1st indirectly detected dimension (P)

Number of increments in 1st indirectly detected dimension (P)

phase Phase selection (P)

Sw Spectral width in directly detected dimension (P)

Sw2 Spectral width in 2nd indirectly detected dimension (P)

Sw3 Spectral width in 3rd indirectly detected dimension (P)

#### sw2 Spectral width in 2nd indirectly detected dimension (P)

Description: Analogous to the sw parameter except that sw2 applies to the second indirectly

detected dimension of a multidimensional data set. The increment of the variable evolution time d3 is automatically calculated from sw2. The number of increments for this dimension is set by ni2. To create sw2 in the current experiment, as well as d3, ni2, and phase2, enter addpar('3d').

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

d3 Incremented delay for 2nd indirectly detected dimension (P)

ni2 Number of increments in 2nd indirectly detected dimension (P)

phase 2 Phase selection for 3D acquisition (P)

Sw Spectral width in directly detected dimension (P)

Sw1 Spectral width in 2nd indirectly detected dimension (P)

Sw3 Spectral width in 3rd indirectly detected dimension (P)

#### sw3 Spectral width in 3rd indirectly detected dimension (P)

Description: Analogous to the sw parameter except that sw3 applies to the third indirectly

detected dimension of a multidimensional data set. The increment of the variable evolution time d4 is automatically calculated from sw3. The number of increments for this dimension is set by ni3. To create sw3 in the current experiment, as well as d4, ni3, and phase3, enter addpar ('4d').

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

Incremented delay for 3rd indirectly detected dimension (P)

Number of increments in 3rd indirectly detected dimension (P)

par4d Create 4D acquisition parameters (C)
phase3 Phase selection for 4D acquisition (P)

Sw Spectral width in directly detected dimension (P)
Sw1 Spectral width in 1st indirectly detected dimension (P)
Sw2 Spectral width in 2nd indirectly detected dimension (P)

# sysgcoil System gradient coil (P)

Description: Specially reserved string parameter that specifies which physical gradient set is

currently installed, and allows convenient updating of important gradient characteristics when one gradient set is interchanged for another. The value to sysgcoil is assigned to the parameter gcoil when joining experiments or

retrieving parameter sets.

This parameter is set in the Spectrometer Configuration window to the name of the gradient set in use. Once set, it is then available to all experiments and to all

users.

See also: VnmrJ Installation and Administration; VnmrJ Imaging NMR

Related: config Display current configuration and possibly change it (M)

gcoil Current gradient coil (P)
gmax Maximum gradient strength (P)

setgcoil Assign sysgcoil configuration parameter (M)

#### system System type (P)

Description: A global parameter that sets the basic type of system: spectrometer or data

station. The value is set using the System Type label in the Spectrometer

Configuration window.

Values: 'spectrometer' is a spectrometer system (Spectrometer choice in

Spectrometer Configuration window).

'datastation' is a system used as a data station (Data Station choice in Spectrometer Configuration window). Acquisition is not allowed in this setting.

See also: VnmrJ Installation and Administration

Related: config Display current configuration and possibly change it (M)

Console System console type (P)

systemdir VnmrJ system directory (P)

Description: Contains path to VnmrJ system directory, typically /vnmr. The UNIX

environmental variable vnmrsystem initializes systemdir at bootup.

See also: NMR Spectroscopy User Guide

# T

t1  $T_1$  exponential analysis (M)

 $T_1$  exponential analysis with short output table (M)

 $T_2$  exponential analysis (M)

t2s  $T_2$  exponential analysis with short output table (M) tabc Convert data in table order to linear order (M)

tan Find tangent value of an angle (C)

tape Read tapes from VXR-style system (M,U)
tape Control tape options of files program (P)
tchan RF channel number used for tuning (P)

Send Tcl script to Tcl version of dg window (C)

temp

Open the Temperature Control window (C)

temp Sample temperature (P)
tempcal Temperature calculation (C)

tempcalc Measure approximate sample temperature in Cold Probes (M)

testacquire Test acquire mode (P)

testct Check ct for resuming signal-to-noise testing (M)

testsn Test signal-to-noise of a spectrum (M)
teststr Find which array matches a string M)

Display text or set new text for current experiment (C)

textis Return the current text display status (C)
textvi Edit text file of current experiment (M)

th Threshold (P)

Threshold for integrating peaks in 2D spectra (P)

thadj Adjust threshold for peak printout (M)

Display experiment time or recalculate number of transients (M)

tin Temperature interlock (P)

First-order baseline correction (P)

Left-shift FID to time-domain cursor (M)

tmsref Reference 1D proton or carbon spectrum to TMS (M)

Nucleus for observe transmitter (P)

Set up parameters for TNCOSYPS pulse sequence (M) tncosyps tndqcosy Set up parameters for TNDQCOSY pulse sequence (M) tnmqcosy Set up parameters for TNMQCOSY pulse sequence (M) Set up parameters for TNNOESY pulse sequence (M) tnnoesy Set up parameters for TNROESY pulse sequence (M) tnroesy Set up parameters for TNTOCSY pulse sequence (M) tntocsy Tocsy Convert the parameters to a TOCSY experiment (M) Tocsy1d Convert the parameter set to a Tocsy1d experiment (M)

Set up the tocsyHT experiment (M)

Frequency offset for observe transmitter (P)

Observe transmitter power level with linear amplifiers (P)

Observe transmitter fine power (P)

Observe transmitter linear modulator power (P)

trace Mode for *n*-dimensional data display (P)

Sample changer tray slots (P)

Set up parameters for TROESY pulse sequence (M)

Truncate real numbers (O)

tshift Adjust tau2 to current cursor position (M)
tugain Amount of receiver gain used by qtune (P)

tune Assign a frequency to a channel for probe tuning (C)
tunehf Tune both H1 and F19 on an HFX probe (M)
tunematch Default match target, in percent of optimum (P)

tunemethod Method to use for tuning (P)

tuneResult Message indicating how well the tuning succeeded (P)

tunesw Width of the tuning sweep in Hz (P
tupwr Transmitter power used in tuning (P)
typeof Return identifier for argument type (O)

# t1 $T_1$ exponential analysis (M)

Description: Processes data obtained using an array of values of the parameter d2 for a  $T_1$ 

experiment. It runs expfit, which does an exponential curve fitting that determines the value of  $T_1$ . The output is matched to the equation:

M(t) = (M(0) - M0) \* exp(-t/T1) + M0

where M0 is the equilibrium Z magnetization and M(0) is the magnetization at time zero (e.g., immediately after the 180° pulse for an inversion recovery  $T_1$  experiment). Notice that this equation will fit inversion recovery data (for which M(0) is approximately equal to -M0) or saturation recovery data (for which

M(0) is 0).

The required input is the file fp.out from fp and the values of the arrayed parameter. The  $T_1$  analysis is done for all the peaks listed in fp.out. Peaks are selected for analysis by entering fp (index1, index2, . . .) before running the analysis. The output file is the analyze.list in the current experiment. The file analyze.out is used by exp1 to display the results. The output of the analysis program shows  $T_1$  and its standard deviation, but does not explicitly show M(0), M0, or their standard deviations. The M(0) and M0 values can be found in "raw" form in analyze.out in the current experiment, but their standard deviations are not part of the program output.

See also: NMR Spectroscopy User Guide

Related: d2 Incremented delay in 1st indirectly detected dimension (P)

expfit Make least squares fit to polynomial or exponential curve (C)

fp Find peak heights (C)

 $T_1$  exponential analysis with short output table (M)

 $T_2$  exponential analysis (M)

 $T_2$  exponential analysis with short output fable (M)

# t1s $T_1$ exponential analysis with short output table (M)

Description: Performs the same analysis as t1 but produces a short output table showing

only a summary of the measured relaxation times.

See also: NMR Spectroscopy User Guide

Related:  $t_1$   $T_1$  exponential analysis (M)

# t2 $T_2$ exponential analysis (M)

Description:

Processes data obtained using an array of values for the base time parameter bt for a  $T_2$  experiment. It runs <code>expfit</code>, which does an exponential curve fitting that determines the value of  $T_2$ . The output is matched to the equation:

M(t) = (M(0) - M(inf))\*exp(-t/T2) + M(inf)

where M(0) is the magnetization at time zero (i.e., the full magnetization excited by the observe pulse) and M(inf) is the xy-magnetization at infinite time (zero unless the peak is sitting on an offset baseline).

The required input is the file fp.out from fp and the values of the arrayed parameter. The  $T_2$  analysis is done for all the peaks listed in fp.out. Peaks are selected for analysis by entering fp (index1, index2, . . .) before running the analysis. The output file is the file analyze.list in the current experiment. The file analyze.out is used by exp1 to display the results. The output of the analysis program shows  $T_2$  and its standard deviation, but does not explicitly show M(0), M(inf), or their standard deviations. The M(0) and M(inf) values can be found in "raw" form in analyze.out in the current experiment, but their standard deviations are not part of the program output.

See also: NMR Spectroscopy User Guide

Related: expfit Make least squares fit to polynomial or exponential curve (C)

fp Find peak heights (C)  $T_1$  exponential analysis (M)

t1s  $T_1$  exponential analysis with short output table (M) t2s  $T_2$  exponential analysis with short output fable (M)

# t2s $T_2$ exponential analysis with short output table (M)

Description: Performs the same analysis as t2 but produces a short output table showing

only a summary of the measured relaxation times.

See also: NMR Spectroscopy User Guide

Related:  $t_2$   $T_2$  exponential analysis (M)

#### tabc Convert data in table order to linear order (M)

Syntax: tabc<(dimension)>

Description: Converts arbitrarily ordered data obtained under control of an external AP table to linear monotonic order, suitable for processing in VnmrJ. The data must have

been acquired according to a table in the tablib directory.

Imaging and other 2D experiments are normally acquired so that the order of the incremented acquisition parameter, such as the phase-encode gradient, is linear and monotonic. For a standard imaging experiment, this linear order means that the phase-encode gradient progresses from a starting negative value monotonically up through zero to a positive value (e.g., -64, -63, -62, ..., -1, 0, 1, ..., 62, 63). The ft2d program assumes this structure in its operation.

Data from table-driven 2D pulse sequences is used by entering tabc *only once* before normal 2D processing and/or parameter storage. In this situation, tabc takes no arguments and is executed by entering tabc in the command window. A simple check is done by tabc to prevent it from being executed more than once on the same data set.

2D data is expected to be in the standard VnmrJ format, but if the 2D data is in the compressed format, setting dimension to 1 converts the data. tabc supports all 2D data types recognized by VnmrJ: arrayed, compressed multislice, and arrayed compressed multislice,

3D data is expected to be in the compressed/standard format, in which there are ni standard 2D planes of data (the third dimension), each consisting of nf compressed FIDs (the second dimension). Setting dimension to 3 reorders 3D data acquired with an external table.

tabc reads the file fid in the acqfil subdirectory of the current experiment. Before the data is reordered, this file is written to the file fid.orig in the same acqfil directory. If for any reason tabc fails or results in an unpredictable or undesired transformation, the original raw data can be recovered by moving fid.orig back to fid. To gain more disk space, you can delete fid.orig after you are satisfied that conversion is successful.

Use tabc on saved data that has been loaded into an experiment or on data in an experiment that has just been acquired but not yet saved. In the first case, converted data must be resaved for the saved data set to reflect conversion.

tabc requires that data must have the same number of "traces" as the table elements. It does not support any of the advanced features of table expansion (e.g., the entire table must be explicitly listed in the table file), and expects to find only one table in a file; whether the table is t1 or t60 is unimportant.

Arguments: dimension specifies the type of data to be converted: 1 for 2D compressed

data, 2 for 2D standard data, or 3 for 3D compressed/standard data. The default

is 2.

Examples: tabc

tabc(1) tabc(3)

See also: VnmrJ Imaging NMR

Related: flashc Convert compressed 2D data to standard 2D format (C)

ft2d Fourier transform 2D data (C)

Number of increments in 1st indirectly detected dimension (P)

nf Number of FIDs (P)

# tan Find tangent value of an angle (C)

Syntax: tan(angle)<:n>

Description: Finds the tangent of an angle.

Arguments: angle is an angle, in radians.

n is the return value giving the tangent of angle. The default is to display the

tangent value in the status window.

Examples: tan(.5)

tan(val):tan val

See also: User Programming

Related: atan Find arc tangent value of a number (C)

cos Find cosine value of an angle (C)
exp Find exponential value of a number (C)
ln Find natural logarithm of a number (C)
sin Find sine value of an angle (C)

# tape Read tapes from VXR-style system (M,U)

Description: Displays the contents of a VXR-style (Gemini, VXR-4000, or XL) 9-track tape

for use with VnmrJ or reads one or several files from the tape into the current directory. Note that the *write* option is not supported (i.e., VnmrJ only *reads* 

tapes in a VXR-style format and does not write to a tape).

Arguments: device is the tape drive device name. The default value is /dev/rst8. For

AIX systems, device should be /dev/rmt0. If the default value is not set properly or another device name is wanted, be sure to type -d and a space before the device name you want to input.

type is the type of tape to be accessed. '-q' or '-s' select the 1/4-inch tape unit ("streaming" or cartridge tape); this is the default. '-9', '-h', or '-n' select the 1/2- inch tape unit (open reel tape drive).

option is one of the following:

- 'help' is a keyword to display help on the use of the system.
- 'cat' is a keyword to display a catalog of files on tape.
- 'read' is a keyword to read one or more files. This option requires that the files be listed as the next argument.
- 'rewind' is a keyword to rewind tape (1/2-inch tape only).
- 'quit' is a keyword to release the tape drive (1/2-inch tape only).

file1, file2, ... are the names of one or more files to be read. Wildcard characters (\* and ?) can be used.

Examples: tape('cat')

tape('-h','read','mydata')

tape -h read mydata

tape -d /dev/rmt/01b read mydata

Related: decomp Decompose a VXR-style directory (C)

vxr\_unix Convert VXR-style text files to UNIX format (M,U)

# tape Control tape options of files program (P)

Description: Defines device that files program accesses when it is instructed to read or

write to a tape. The parameter tape is in the user's global parameter tree.

Values: Name of a device. The default device is /dev/rst8. If tape does not exist

or is set to the null string (two single quotes with no space between), files uses its default device value. Notice that different computers define tape drives differently. For VnmrSGI, tape='/dev/tapens' is appropriate. For

Solaris, tape='/dev/rmt/0mb'.

Related: files Interactively handle files (C)

# tchan RF channel number used for tuning (P)

Description: Set by the protune macro.

See also: NMR Spectroscopy User Guide

Related: protune Macro to start ProTune (M)

atune ProTune Present (P)

mtune Tune probe using swept-tune graphical display (M)

tugain Receiver gain used in tuning (P)
tunesw Width of the tuning sweep in Hz (P
tupwr Transmitter power used in tuning (P)

tcl Send Tcl script to Tcl version of dg window (C)

Syntax: tcl(script)

Description: Sends a Tcl (Tool Command Language) script to the Tcl version of the dq

window. If this window is not active, this command does nothing.

Arguments: script is any legal Tcl script.

See also: User Programming

Related: dg Display group of acquisition/processing parameters (C)

# temp Open the Temperature Control window (C)

Applicability: Systems with a variable temperature (VT) controller.

Description: Opens the Temperature Control window, which has the following capabilities:

• Turn temperature control off.

• Set temperature control on at a specified temperature in degrees C.

• Enable temperature control from within an experiment using the temperature and the su, go, ga, or au macros. This mode is the default.

• Alternatively, turn off experiment control of the temperature and allow only the Temperature Control window (and sethw) to set the temperature. This mode has the advantage that, often times, temp is different between experiments. Joining a different experiment and entering go can unexpectedly change the temperature. This mode prevents this problem.

• Resetting the temperature controller when the temperature cable is reconnected to a probe.

See also: NMR Spectroscopy User Guide

Related: acqi Interactive acquisition display process (C)

au Submit experiment to acquisition and process data (M)
qa Submit experiment to acquisition and FT the result (M)

Submit experiment to acquisition (M)

readhw Read current values of acquisition hardware (C)
sethw Set values for hardware in acquisition system (C)
su Submit a setup experiment to acquisition (M)

temp Sample temperature (P)
tin Temperature interlock (P)

#### temp Sample temperature (P)

Applicability: Systems with a variable temperature (VT) module.

Description: Sets the temperature of sample.

Values: 'n' or -150 to +200, in steps of 0.1°C. 'n' instructs the acquisition system

not to change the VT controller and to ignore temperature regulation throughout

the course of the experiment.

See also: NMR Spectroscopy User Guide

Related: readhw Read current values of acquisition hardware (C)

Den the Temperature Control window (C)

tempcal Temperature calculation (C)
tin Temperature interlock (P)

Variable temperature cutoff point (P)

# tempcal Temperature calculation (C)

Applicability: Systems with a variable temperature (VT) module.

Syntax: tempcal(solvent)<:temperature>

Description: For exact determination of sample temperature when using the VT unit, a

temperature calibration curve must be made for each probe used. All data, such as gas flow, must be noted. Use samples of ethylene glycol for high-temperature calibration, and use samples of methanol for low-temperature calibration. To make the calculation:

• Bring the sample to the desired temperature and allow sufficient time for equilibration, then obtain a spectrum.

• Next, align two cursors on the two resonances in the spectrum, then enter tempcal('e') for ethylene glycol, or enter tempcal('m') for methanol. The temperature is calculated based on the difference frequency between the cursors.

Arguments: solvent is the sample solvent: 'glycol', 'e', or 'g' for ethylene glycol,

or 'methanol' or 'm' for methanol.

temperature returns the calculated value of the sample temperature. The

default is the system displays the value.

Examples: tempcal('glycol')

tempcal('m'):temp

See also: NMR Spectroscopy User Guide

# tempcalc Measure approximate sample temperature in Cold Probes (M)

Applicability: Systems with Varian, Inc. Cold Probes

Description: Measure the approximate sample temperature and the actual sample

temperature gradient and generate a report. Requires a ~1% HOD CH<sub>3</sub>CN

sample.

# testacquire Test acquire mode (P)

Description: Allows test acquisitions to be done while a study queue is active, without using

the study queue. When this mode is enabled, acquisitions do not update the status of the currently loaded experiment in the study queue, and data is not saved in the study queue. This mode is set from the Test mode check box in the

Acquisition menu or from the command line.

Syntax: testacquire=<'y' or 'n'>

Values: 'y' test acquire mode enabled

'n' test acquire mode disabled

Related: acquire Acquire data (M)

ave Save data (M)

# testct Check ct for resuming signal-to-noise testing (M)

Description: Used by the testsn macro to decide when to resume testing of signal-to-

noise. See the description of testsn for details.

See also: NMR Spectroscopy User Guide

Related: ct Completed transients (P)

testsn Test signal-to-noise of a spectrum (M)

#### testsn Test signal-to-noise of a spectrum (M)

Description:

Part of the automatic periodic signal-to-noise testing that occurs during various automated acquisitions, most notably c13. Transforms the data using fn=16000, and then baseline corrects, setting the left-most 10% of the spectrum and the right-most 2% as baseline. After the baseline correction, testsn uses getsn to calculate the signal-to-noise.

- If signal-to-noise exceeds the desired goal in parameter sn (found in the standard carbon parameter set /vnmr/stdpar/c13), testsn aborts the experiment using the command halt, which initiates processing according to the wexp parameter.
- If signal-to-noise is not reached, testsn estimates the signal-to-noise ratio at the end of the experiment. If signal-to-noise target will not be reached by then, it cancels subsequent signal-to-noise testing, but allows the experiment to proceed.
- If the signal-to-noise target will be reached before the end of the experiment, it saves the estimated number of transients required to reach the goal in the parameter r7 (using a conservative estimate), and then sets the processing at future blocks to be only testet, which simply tests if ct is greater than r7, and, if so, resumes testing of signal-to-noise with testsn.

See also: NMR Spectroscopy User Guide

Related: c13 Automated carbon acquisition (M)

fn Fourier number in directly detected dimension (P) getsn Get signal-to-noise estimate of a spectrum (M)

halt Abort acquisition with no error (C) r1-r7 Real parameter storage for macros (P)

sn Signal-to-noise ratio (P)

Check ct for resuming signal-to-noise testing (M)

wexp

Specify action when experiment completes (C)

#### teststr Find which array matches a string M)

Syntax: teststr(parameter,string <,tree>):\$ret

Description:

The teststr command requires at least two arguments. The first is the name of a string parameter. The first argument must generally be enclosed in single quotes. The teststr command needs the name of the parameter, not its values. The second is a string. The optional third argument is the parameter tree. The default is current.

Macro parameters can be used as the first argument. In this case, the third argument must be 'local'.

This command sets \$ret to the index of the array element that matches the second argument. If none of the array values of the parameter match the second argument, a zero is returned.

Examples:

```
n1='hello','labas','gidday','hola','bonjour','ciao'
teststr('n1','labas'):r1
```

sets r1=2, since 'labas' matches element 2 of the n1 array.

The elements do not need to be single words. For example, n1='good night', 'labanaktis', 'bonne nuit', 'gute

Nacht','boa noite','buonas noces'

teststr('n1','boa noite'):r1

sets r1=5. The strings must match exactly, including upper and lower case

```
teststr('n1','qute nacht'):r1
```

sets r1=0, since the lower case n in nacht does not match the upper case N in Nacht.

For local dollar variables, the 'local' argument must be used. Again, enclose the name of the local parameter in single quotes.

```
$greet='hello','labas','gidday','hola', 'ciao'
teststr('$greet','labas','local'):r1
```

#### text Display text or set new text for current experiment (C)

Syntax: text<(text string)><:string variable>

Description: Associated with each experiment is a text file, consisting of a block of text, that

can be used to describe the sample and experiment. text allows displaying the text file and changing the text file for the current experiment. A UNIX text editor, such as vi, or the macro textvi can also be used to edit the text file of

the current experiment.

Arguments: text string is a string of text that replaces the existing text file. The default

is to display the text file in the current experiment. The characters  $\ \ \$  or  $\ \$ n can be used in the string to denote a new line, and the characters  $\ \ \ \$ t can be used to

denote a tab (see example below).

string\_variable returns the text in text\_string as a string variable. Thus, for example, the text:nl and text(nl+'cosy experiment') commands, where nl is a string, can be used in a macro to add a "cosy experiment" to the text. An equivalent operation using the atext command

would be atext ('cosy experiment').

Examples: text('Sample 101\tCDCl3\\13 February')

See also: NMR Spectroscopy User Guide

Related: atext Append string to the current experiment text (M)

ctext Clear the text of the current experiment (C)

curexp Current experiment directory (P)

dtext Display a text file in the graphics window (C)

puttxt Put text file into another file (C)
textvi Edit text file of current experiment (M)

vnmrprint Print text files (U)

# textis Return the current text display status (C)

Syntax: (1) textis(command):\$yes\_no

(2) textis:\$display\_command

Description: Determines if a command given by the user currently controls the text window

(syntax 1) or returns the name of the command currently controlling the text

window (syntax 2).

Arguments: command is the name of a command that potentially may be controlling the text

window

\$yes\_no returns 1 if command controls the text window, or 0 if it does not.
\$display command returns the name of the command currently controlling

the text window.

Examples: textis:\$display

if (\$display = 'dg') then . . . endif

See also: User Programming

Related: graphis Return the current graphics display status (C)

textvi Edit text file of current experiment (M)

Description: Edits the text file of the current experiment using the UNIX text editor vi.

textvi is equivalent to the command vi (curexp+'/text').

See also: NMR Spectroscopy User Guide

Related: edit Edit a file with user-selectable editor (M)

Display text or set new text for current experiment (C)

vi Edit text file with vi editor (M)

th Threshold (P)

Description: Sets threshold for printout of peak frequencies so that peaks greater than th on

the plot appear on any peak listings. th is always bipolar (i.e., negative peaks

greater in magnitude than th also appear in peak listings).

Values: 0 to 1e9, in mm.

See also: NMR Spectroscopy User Guide

Related: thadj Adjust threshold for peak printout (M)

th2d Threshold for integrating peaks in 2D spectra (P)

Description: Used by 112d when determining the bounds of a peak and calculating its

volume. To create the 2D peak picking parameters th2d and xdiag in the

current experiment, enter addpar ('112d').

Values: From 0.0 to 1.0. If th2d=1.0, 112d integrates all points in the peak that

are above the current threshold for the spectrum (i.e., the portion of the peak that can be seen in a contour plot of the spectrum). A smaller value causes 112d to integrate a larger area when determining the volume of a peak. If th2d=0.5, for example, 112d integrates all points in a peak that are above 0.5 times the

current threshold.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

Automatic and interactive 2D peak picking (C)

xdiag Threshold for excluding diagonal peaks when peak picking (P)

thadj Adjust threshold for peak printout (M)

Syntax: thadj<(max\_peaks<,noise\_mult<,llarg1<,llarg2>>>)>

Description: Adjusts the threshold th so that no more than a specified maximum number of

peaks are found in a subsequent line listing (see nll) and so that th is at least

a specified noise multiplier times the root-mean-square noise level.

Arguments: max\_peaks is the maximum number of peaks in the displayed spectral range.

The default is wc/4 (i.e., the threshold is adjusted such that ppf will produce

a "reasonable" number of lines with any width of plot).

 $\verb"noise_mult" is a noise multiplier used to calculate the minimum value for \verb"th"$ 

from the size of the root-mean-square noise.

llarg1 is the noise mult argument (the default is 3) to the nll command

used inside this macro

llarg2 is the keyword argument ('pos', 'neg', 'all'; the default is

'all'.) to the nll command used inside this macro.

Examples: thadj

thadj(50)

thadj(200,4)

```
thadj (200,4,2)
thadj (200,4,2,'pos')
```

See also: NMR Spectroscopy User Guide

Related: nll Find line frequencies and intensities (C)

> Plot teak frequencies over spectrum (M) ppf

t.h Threshold (P)

vsadj Automatic vertical scale adjustment (M)

vsadj2 Automatic vertical scale adjustment by powers of two (M) Automatic vertical scale adjustment for <sup>13</sup>C spectra (M) vsadjc Automatic vertical scale adjustment for <sup>1</sup>H spectra (M) vsadjh

Width of chart (P) WC

#### time Display experiment time or recalculate number of transients (M)

time<(<hours,>minutes)> Syntax:

Description: Estimates the acquisition time or recalculates the number of transients so that

> the total acquisition time is approximately the requested time. The parameters looked at when calculating the time per transient are d1, d2, d3, at, ni, sw1,

ni2, and sw2.

Arguments: hours and minutes are numbers making up a time to be used by the system

to recalculate the parameter nt so that the total acquisition time is

approximately the time requested; the default (no arguments) is for the system to estimate the acquisition time for a 1D, 2D, or 3D experiment using the

parameters in the current experiment.

Examples: time

time(2,45)

See also: NMR Spectroscopy User Guide

Related: Acquisition time (P)

d1

First delay (P)

42 Incremented delay in 1st indirectly detected dimension (P) d3 Incremented delay in 2nd indirectly detected dimension (P)

Display experiment time (C) exptime

ni Number of increments in 1st indirectly detected dimension (P) Number of increments in 2nd indirectly detected dimension (P) ni2

Number of transients (P) nt.

Spectral width in 1st indirectly detected dimension (P) sw1 Spectral width in 2nd indirectly detected dimension (P) SW2

#### tin Temperature interlock (P)

Description: Controls error handling based on temperature regulation. If temperature

regulation is lost, tin can be used to select whether an error is generated and acquisition is halted or whether a warning is generated and acquisition continues. In both cases, the lost regulation will cause werr processing to occur, thus providing a user-selectable mechanism to respond to VT failure.

Values: 'n' turns off the temperature interlock feature

> 'w' indicates the variable temperature regulation light is monitored during the course of the experiment and, if it starts to flash (regulation lost), a warning is generated; however, acquisition is not stopped.

> 'y' indicates the variable temperature regulation light is monitored during the course of the experiment and, if it starts to flash (regulation lost), the current

data acquisition is stopped. The acquisition will not resume automatically if

regulation is regained.

See also: NMR Spectroscopy User Guide

Related: in Lock and spin interlock (P)

werr When error (P)

tlt First-order baseline correction (P)

Description: When spectral display is active, the command dc turns on a linear drift

correction (baseline correction). The result of this operation includes

calculating a first-order baseline correction parameter tlt. The calculation is made by averaging of a small number of points at either end of the display and

drawing a straight line baseline between them.

See also: NMR Spectroscopy User Guide

Related: cdc Cancel drift correction (C)

dc Calculate spectral drift correction (C)

1v1 Zero-order baseline correction (P)

tmove Left-shift FID to time-domain cursor (M)

Description: Provides an alternative method of left shifting time-domain data. To use this

method, position the right time cursor at the place that should be the start of the

FID, then enter tmove. This adjusts lsfid to left-shift the FID.

See also: NMR Spectroscopy User Guide

Related: lsfid Number of complex points to left-shift np FID (P)

tmsref Reference 1D proton or carbon spectrum to TMS (M)

Syntax: tmsref:tms found

Description: Tries to locate a TMS line. If found, tmsref re-references the spectrum to the

TMS line and returns a 1 to the calling macro; if not found, tmsref returns 0 and the referencing is left as it was. In the case of other signals (e.g., from silicon grease) immediately to the left of the TMS line (even if they are higher than the reference line), tmsref tries avoiding those by taking the rightmost line in that area, as long as it is at least 10% of the main Si-CH $_3$  signal. Large signals within

0.6 ppm for <sup>1</sup>H (or 6 ppm for <sup>13</sup>C) to the right of TMS may lead to

misreferencing.

Arguments: tms found returns 1 if a TMS line was located or returns 0 if not.

See also: NMR Spectroscopy User Guide

Related: c13 Automated carbon acquisition (M)

h1 Automated proton acquisition (M)

tn Nucleus for observe transmitter (P)

Description: Changing the value of tn causes a macro (tn) to be executed that extracts

values for sfrq and tof from lookup tables. The tables, stored in the directory

/vnmr/nuctables, are coded by atomic weights.

Values: In the lookup tables, typically given by 'H1', 'C13', 'P31', etc. The value

tn='lk' sets the deuterium frequency, and also holds the lock current and switches the relay in the automated deuterium gradient shimming module, if present, so that deuterium signal may be observed without disturbing lock. The

frequency is the same as tn='H2'.

See also: NMR Spectroscopy User Guide

Related: dn Nucleus for first decoupler (P)

dn2 Nucleus for second decoupler (P)
dn3 Nucleus for third decoupler (P)

Transmitter frequency of observe nucleus (P)
tof
Frequency offset for observe transmitter (P)

tncosyps Set up parameters for TNCOSYPS pulse sequence (M)

Description: Sets up a homonuclear correlation experiment (phase-sensitive version) with

water suppression.

See also: NMR Spectroscopy User Guide

tndqcosy Set up parameters for TNDQCOSY pulse sequence (M)

Applicability: Systems with a linear amplifier on the observe channel and a T/R switch.

Description: Sets up a 2D J-correlation experiment with water suppression.

See also: NMR Spectroscopy User Guide

tnmqcosy Set up parameters for TNMQCOSY pulse sequence (M)

Applicability: Systems with hardware digital phaseshifter for transmitting with direct-

synthesis rf; otherwise, software small-angle phaseshifter for transmitting with

the old-style rf is used.

Description: Sets up a multiple-quantum filtered COSY experiment with water suppression.

See also: NMR Spectroscopy User Guide

tnnoesy Set up parameters for TNNOESY pulse sequence (M)

Applicability: Systems with a linear amplifier on the observe channel and a T/R switch.

Description: Sets up a 2D cross-relaxation experiment with water suppression.

See also: NMR Spectroscopy User Guide

tnroesy Set up parameters for TNROESY pulse sequence (M)

Description: Sets up a rotating-frame NOE experiment with water suppression.

See also: NMR Spectroscopy User Guide

tntocsy Set up parameters for TNTOCSY pulse sequence (M)

Applicability: Systems with T/R switch, computer-controlled attenuators, and linear

amplifiers on observe channel.

Description: Sets up a total-correlation spectroscopy experiment (HOHAHA) with water

suppression.

See also: NMR Spectroscopy User Guide

Tocsy Convert the parameters to a TOCSY experiment (M)

Description: Convert parameters to a TOCSY experiment.

See also: NMR Spectroscopy User Guide

Related: ftldac Combined arrayed 2D FID matrices (M)

ft2dacCombined arrayed 2D FID matrices (M)wft1dacCombined arrayed 2D FID matrices (M)wft2dacCombined arrayed 2D FID matrices (M)

#### Tocsy1d Convert the parameter set to a Tocsy1d experiment (M)

Description: Convert the parameter set to a Tocsyld experiment.

See also: NMR Spectroscopy User Guide

Related: Proton Set up parameters for <sup>1</sup>H experiment (M).

selld Selective 1D protocols to set up (M).

# tocsyHT Set up the tocsyHT experiment (M)

Description: Sets up parameters for a Hadamard-encoded tocsy experiment.

See also: NMR Spectroscopy User Guide

Related: htofs1 Hadamard offset in ni (P)

fn1 Fourier number in 1st indirectly detected dimension (P)

ni Number of increments in 1st indirectly detected dimension (P)

ft2d Fourier transform 2D data (C)

sethtfrq1 Set Hadamard frequency list from a line list (M)

Tocsy Set up parameters for a TOCSY pulse sequence (M)

htfrq1 Hadamard frequency list in ni (P)

#### tof Frequency offset for observe transmitter (P)

Description: Controls the exact positioning of the transmitter. As the value assigned to tof

increases, the transmitter moves to a higher frequency (toward the left side of the spectrum). The minimum step size of tof is determined by the type of rf hardware in the spectrometer. The limit is specified using the Step Size label in the Spectrometer Configuration window. Systems with broadband style rf (rftype='b') generally have 100-Hz resolution; all other systems have 0.1

Hz resolution.

Values: Approximate, depends on frequency-100000 to 100000, in Hz.

See also: NMR Spectroscopy User Guide

Related: config Determine current configuration and possibly change it (M)

dof Frequency offset for first decoupler (P)
dof 2 Frequency offset for second decoupler (P)
dof 3 Frequency offset for third decoupler (P)

rftype Type of rf generation (P)

# tpwr Observe transmitter power level with linear amplifiers (P)

Applicability: Systems with a linear amplifier on the observe channel.

Description: Controls transmitter power. The value of the attenuator upper safety limit is set

using the Upper Limit label in the Spectrometer Configuration window.

Depending on hardware adjustments, the system may saturate at a given value

of tpwr (i.e., values above a certain value may give equal output).

Values: On systems with 63-dB attenuator installed: 0 to 63 (63 is maximum power), in

units of dB. About 55 to 60 is normal. Lower values (e.g., 49) might be used for

water suppression experiments like 1-3-3-1.

On systems with 79-dB attenuator installed: –16 to 63 (63 is maximum power),

in units of dB.

CAUTION: Continuous power greater than 2 watts in a switchable probe will

damage the probe. Always carefully calibrate power to avoid exceeding 2 watts. The maximum value for tpwr on a 200-MHz, 300-MHz, or 400-MHz system with a linear amplifier on the decoupler channel has been set to 49, corresponding to about 2 watts of power. Before using tpwr=49 for continuous decoupling, ensure safe

operation by measuring the output power. This should be done during

system installation and checked periodically by the user.

See also: NMR Spectroscopy User Guide

Related: cattn Coarse attenuator (P)

config Determine current configuration and possibly change it (M)
dpwr Power level for first decoupler with linear amplifiers (P)

dpwr2 Power level for second decoupler (P)
dpwr3 Power level for third decoupler (P)
dpwrf First decoupler fine power (P)

fattn Fine attenuator (P)

**Observe** transmitter fine power (P)

# tpwrf Observe transmitter fine power (P)

Applicability: Systems with a fine attenuator on the observe transmitter channel.

Description: Controls the transmitter fine attenuator. Systems with this attenuator are

designated using the Fine Attenuator label in the Spectrometer Configuration window. The fine attenuator is linear and spans  $60~\mathrm{dB}$  or  $6~\mathrm{dB}$ . If tpwrf is not

present, enter create ('tpwrf', 'integer')
setlimit('tpwrf', 4095, 0, 1) to create it.

Values: 0 to 4095, where 4095 is maximum power. If tpwrf does not exist in the

parameter table, a value of 4095 is assumed.

See also: NMR Spectroscopy User Guide

Related: config Determine current configuration and possibly change it (M)

dpwr Power level for first decoupler with linear amplifiers (P)

dpwrf First decoupler fine power (P)

fattn Fine attenuator (P)

Dbserve transmitter power level with linear amplifier (P)

Observe transmitter linear modulator power (P)

# tpwrm Observe transmitter linear modulator power (P)

Description: Controls the power level on the observe transmitter linear modulator. The fine

power control is linear and spans 0 to tpwr.

Values: 0 to 4095, where 4095 is maximum power. If tpwrm does not exist in the

parameter table, a value of 4095 is assumed.

See also: NMR Spectroscopy User Guide

Related: config Determine current configuration and possibly change it (M)

dpwrf First decoupler fine power (P)

fattn Fine attenuator (P)

# trace Mode for *n*-dimensional data display (P)

Description: Sets the multidimensional data display mode.

Values: 'f1' displays the  $f_1$  axis horizontally and allows  $f_1$  traces to be displayed.

'f2' displays the f<sub>2</sub> axis horizontally and allows f2 traces to be displayed.

'f3' displays the f3 axis horizontally and allows f3 traces to be displayed if the

data set is 3D.

See also: NMR Spectroscopy User Guide

# traymax Sample changer tray slots (P)

Applicability: Systems with an automatic sample changer.

Description: Specifies the type of sample changer. It also can be used to disable the sample

changer. The value is set using the Sample Changer label in the Spectrometer

Configuration window.

Values: 0 is setting for no sample changer present or, if a sample changer is attached, to

disable the changer (None choice in the Spectrometer Configuration window).

9, 50, 100, 96, 48 are traymax values that indicate the number of sample slots for the corresponding sample changer (9 is for Carousel, 50 is for SMS/ASM 50 Sample, 100 is for SMS/ASM 100 Sample, 96 is for VAST, and 48 is for NMS,

768 for 768AS).

See also: VnmrJ Installation and Administration; VnmrJ Walkup

Related: config Display current configuration and possibly change it (M)

# troesy Set up parameters for TROESY pulse sequence (M)

Description: Sets up parameters for the transverse cross-relaxation experiment in a rotating

frame.

See also: NMR Spectroscopy User Guide

# trunc Truncate real numbers (O)

Description: In MAGICAL programming, an operator that truncates real numbers.

Examples: \$3 = trunc(3.6)
See also: User Programming

Related: acos Find arc cosine of number (C)

asin Find arc sine of number (C)
atan Find arc tangent of a number (C)
cos Find cosine value of an angle (C)
exp Find exponential value (C)

In Find natural logarithm of a number (C)
tan Find tangent value of an angle (C)
sqrt Return square root of a real number (O)
typeof Return identifier for argument type (O)

# tshift Adjust tau2 to current cursor position (M)

Applicability: Systems with a solids module.

Description: Adjusts tau2 to make the current time cursor position the start of acquisition.

As the time-domain cursor can move between points, this macro allows the accurate adjustment of tau2 so as to start another acquisition exactly at the top

of an echo.

See also: User Guide: Solid-State NMR

#### tugain Receiver gain used in tuning (P)

Description: Used internally by the protune macro to set the receiver gain.

See also: NMR Spectroscopy User Guide

Related: protune Macro to start ProTune (M)

atune ProTune Present (P)

mtune Tune probe using swept-tune graphical display (M)

tchan RF channel number used for tuning (P)

tunematch Default match target, in percent of optimum (P)

tunesw Width of the tuning sweep in Hz (P tupwr Transmitter power used in tuning (P)

# tune Assign a frequency to a channel for probe tuning (C)

Syntax: (1) tune(freq1,<freq2,freq3,freq4>)

(2) tune (chan1, freq1, <chan2, freq2, ...>)

Description: Assigns a frequency to a channel when tuning the probe. The frequency

assignment remains in effect (as a tune frequency) until the next su or go command is executed. Although only the first synthesizer is connected to the tuning system, the console is programmed to set this synthesizer to the desired frequency based on the channel shown on the CHAN readout on the TUNE

INTERFACE unit.

The tune program has two formats. If syntax 1 is used, frequencies are assigned to channels based on the order of the arguments. The first argument is interpreted and assigned to the first (observe) channel, the second argument is assigned to the second (decoupler) channel. A third or fourth argument would be interpreted and assigned in a similar manner.

If syntax 2 is used, the arguments are entered in pairs, with the first argument specifying the rf channel and the next argument specifying the frequency.

tune selects the format based on the first argument. If the first argument is a name for an rf channel, syntax 2 is assumed; otherwise, syntax 1 is used.

Arguments:

freq1, freq2, freq3, and freq4 specify the frequency of the rf channel as a value in MHz (e.g., 200 or 300) or indirectly using the nucleus for tuning the probe (e.g., 'H1' or 'C13'). If a nucleus is entered, it must be found in the nucleus table. The frequency of any channel without an argument is unaffected. For example, tune ('H1', 'C13', 'N15') sets the first channel to tune at the <sup>1</sup>H, the second channel at <sup>13</sup>C, and the third channel at <sup>15</sup>N. If a fourth channel is present, it is not affected. Entering tune ('H1', 'C13', 200) assigns the same frequencies for the first and second channels but the third channel tunes to 200 MHz, regardless of the proton frequency.

chan1, chan2, chan3, and chan4 specify the channel directly:

- 'todev' or 'ch1' specify channel 1 (observe transmitter).
- 'dodev' or 'ch2' specify channel 2 (first decoupler).
- 'do2dev' or 'ch3' specify channel 3 (second decoupler).
- 'do3dev' or 'ch4' specify channel 4 (third decoupler).

Only one of these keywords is used per channel (do not enter the channel using just its number). If a channel does not have a keyword entered as an argument, that channel is not affected (e.g., tune ('ch4', 'P31') selects the frequency corresponding to <sup>31</sup>P on the fourth channel, but leaves the first three channels unaffected).

Examples: tune('H1','C13','N15')

tune('H1','C13',200)
tune('ch4','P31')

See also: NMR Spectroscopy User Guide

Related: dfrq Transmitter frequency of first decoupler (P)

dfrq2 Transmitter frequency of second decoupler (P)
dfrq3 Transmitter frequency of third decoupler (P)
go Submit experiment to acquisition (C)

mtune
Tune probe using swept-tune graphical display (M)
qtune
Tune probe using swept-tune graphical tool (C)
sfrq
Transmitter frequency of observe nucleus (P)
spcfrq
Display frequencies of rf channels (M)
su
Submit a setup experiment to acquisition (C)

tune Assign frequencies (C)

# tunehf Tune both H1 and F19 on an HFX probe (M)

Syntax: tunehf < ('x') >

Description: Tune both H1 and F19 on an HFX probe. Including the optional argument,

tunehf ('x') also tunes the low band channel to dn (dfrq).

Arguments: 'x'—low band channel to dn (dfrq)

See also: NMR Spectroscopy User Guide

Related: protune Macro to start ProTune (M)

# tunematch Default match target, in percent of optimum (P)

Description: The default match target, in percent of optimum. This local real parameter must

be created. It is used as the match criterion in calls of the form

protune (599.96)

See also: NMR Spectroscopy User Guide and VnmrJ Walkup

Related: protune Macro to start ProTune (M)

**create** Create new parameter in a parameter tree (C)

atune ProTune Present (P)

mtune Tune probe using swept-tune graphical display (M)

tchan RF channel number used for tuning (P)
tugain Receiver gain used in tuning (P)
tunesw Width of the tuning sweep in Hz (P)
tupwr Transmitter power used in tuning (P)

#### tunemethod Method to use for tuning (P)

Applicability: Liquids, VnmrJ Walkup, Automation

Description: Specify probe tuning method. Methods are located in:

\$home/vnmrsys/tune/methods for local user or
/vnmr/tune/methods for access by all users.

The method determines the nucleus to tune and how coarse or fine the probe is

tuned as a percentage of the optimal pw.

Values: 'lohi' -tune low band to medium criterion then tune high band to medium

criterion

'<name>' - user defined method.

See also: NMR Spectroscopy User Guide and VnmrJ Walkup

Related: atune ProTune Present (P)

protune Macro to start ProTune (M)wtune Specify when to tune (P)

tuneResult Message indicating how well the tuning succeeded (P)

Description: Message indicating how well the tuning succeeded. This local string parameter

is created by ProTune and set to a string describing the result of the tuning. The first word of the message will be "ok" if tuning is successful, "failed" if it fails, and "Warning:" if tuning was not done but the experiment should proceed.

See also: NMR Spectroscopy User Guide and VnmrJ Walkup
Related: protune Macro to start ProTune (M)

tunesw Width of the tuning sweep in Hz (P)

Description: Sets the width of the tuning sweep in Hz and is set by the protune macro.

See also: NMR Spectroscopy User Guide and VnmrJ Walkup

Related: protune Macro to start ProTune (M)

atune ProTune Present (P)

mtune Tune probe using swept-tune graphical display (M)

tchan RF channel number used for tuning (P)
tuqain Receiver gain used in tuning (P)

tunematch Default match target, in percent of optimum (P)

Transmitter power used in tuning (P)

tupwr Transmitter power used in tuning (P)

Description: The transmitter power used in tuning. The aptune pulse sequence uses this to

set the transmitter power. Set by the protune macro.

See also: NMR Spectroscopy User Guide and VnmrJ Walkup

Related: protune Macro to start ProTune (M)

atune ProTune Present (P)

mtune Tune probe using swept-tune graphical display (M)

tchan RF channel number used for tuning (P)
tugain Receiver gain used in tuning (P)

tunematch Default match target, in percent of optimum (P)

tunesw Width of the tuning sweep in Hz (P

typeof Return identifier for argument type (O)

Syntax: typeof

Description: In MAGICAL programming, an operator that returns an identifier (0 or 1) for

the type (real or string) of an argument.

Examples: if typeof('\$1') then \$arg=1 else \$arg=\$1 endif

See also: User Programming

Related isreal Utility macro to determine a parameter type (M)

isstringUtility macro to determine a parameter type (M)onMake a parameter active or test its state (C)

Return number of elements in an arrayed parameter (O)



ultra8 Selects the Ultra 8 shim configuration (M)
ultra18 Selects the Ultra 18 shim configuration (M)

undospinsRestore spin system as before last iterative run (M)undosyRestore original 1D NMR data from sub experiment (M)

unit Define conversion units (C)

unlock Remove inactive lock and join experiment (C)
updatepars Update all parameter sets saved in a directory (M)

updateprobe Update probe file (M)

updaterev Update after installing new VnmrJ version (M)

updtgcoil Update gradient coil (M)

updtparam Update specified acquisition parameters (C)

**usemark** Use "mark" output as deconvolution starting point (M)

**userdir** VnmrJ user directory (P)

**usergo** Experiment setup macro called by go, ga, and au (M)

userfixpar Macro called by fixpar (M)

# ultra8 selects the Ultra 8 shim configuration (M)

Syntax: ultra8

Description: The ultra8 macro selects the Ultra 8 shim configuration and selects an

appropriate template for the dgs command and manual shim panel.

Administrator privilege is required to change the shim configuration. The shims

are: z1c z2c x1 y1 xz yz xy x2y2.

Related: ultra18 selects the Ultra 18 shim configuration (M)

#### ultra18 Select 18 shim configuration for Ultra 18 shim power supply (M)

Syntax: ultra18

Description: Selects the 18 shim configuration for the Ultra 18 shim power supply and selects

an appropriate template for the dgs command and manual shim panel. Administrator privilege is required to change the shim configuration.

The shims are: z1 z1c z2 z2c z3c z4c x1 y1 xz yz xy x2y2

x3 y3 xz2 yz2 zxy zx2y2

Related: ultra8 selects the Ultra 8 shim configuration (M)

# undospins Restore spin system as before last iterative run (M)

Description: Returns the values of the line assignments and the chemical shifts and coupling

constants existing before the last iterative adjustment with

spins('iterate'), and then runs spins. The parameters are returned
from the file spini.inpar and the transitions from the file spini.savela

in the current experiment.

See also: NMR Spectroscopy User Guide

Related: spins Perform spin simulation calculation (C)

undosy Restore original 1D NMR data from sub experiment (M)

Description: Restores the 1D DOSY data stored by the dosy macro (if data exists) by

recalling the data stored in the file subexp/dosy2Ddisplay in the current experiment. undosy and redosy enable easy switching between the 1D DOSY data (spectra as a function of gzlvll) and the 2D DOSY display

(signal as a function of frequency and diffusion coefficient).

See also: NMR Spectroscopy User Guide

Related: dosy Process DOSY experiments (M)

redosy Restore 2D DOSY display from subexperiment (M)

#### unit Define conversion units (C)

Syntax: unit<(suffix,label,m<,tree><,'mult'|'div'> \

,b<,tree><,'add'|'sub'>)>

Description: Defines a linear relationship that can be used to enter parameters with units. The

unit is applied as a suffix to the numerical value (e.g., 10k, 100p). The definition of the linear relations follows the traditional y=mx+b equation,

where x is the input value and y is the converted result.

 $Entering \ the \ unit \ command \ with \ no \ arguments \ displays \ all \ currently \ defined$ 

units. To remove a unit, define the unit with a 0 for the slope.

A convenient place to put unit commands for all users is in the bootup

macro. Put private unit commands in a user's login macro.

Arguments: suffix is a string identifying the name for the unit. The length of the string is

limited to 12 characters.

label is a string for the name to be displayed when the axis parameter is set to the value of the suffix (if the suffix is only a single character). The length of the string is limited to 12 characters.

m is the slope of the linear relationship, defined either as a numerical value or as the name of a parameter. If a parameter name is used, it may be optionally followed with the parameter tree to use (argument tree) and by another optional keyword that specifies whether the parameter value should be a multiplier (keyword 'mult') or divisor (keyword 'div').

tree is the parameter tree to use (i.e., 'current', 'processed', 'global', or 'systemglobal'). The default tree is 'current'.

'mult' is a keyword that specifies that a parameter value used for the slope should be a multiplier. This is the default for the slope.

'div' is a keyword that specifies that a parameter value used for the slope should be a divisor.

b is the intercept of the linear relationship, defined either as a numerical value or as the name of a parameter. If a parameter name is used, it may be optionally followed with the parameter tree to use (argument tree) and by another optional keyword that specifies whether the parameter value should be added (keyword 'add') or subtracted (keyword 'sub').

'add' is a keyword that specifies that a parameter value used for the intercept should be a added. This is the default for the intercept.

'sub' is a keyword that specifies that a parameter value used for the intercept should be a subtract.

Examples: unit

Displays all currently defined units

unit('k','kHz',1000) r1=10k will set r1 to 10000

unit('p','ppm','reffrq','processed')

r1=10p will set r1 to 10\*reffrq, where reffrq from processed tree

unit('p','',0)

r1=10p will set r1 to 10 and give an error "unknown unit p"

unit('F','degF',5/9,-32\*5/9) r1=212F will set r1 to 100 (degrees C)

unit('C','degC',9/5,32) r1=100C will set r1 to 212 (degrees F)

See also: NMR Spectroscopy User Guide, User Programming

Related: axis Axis label for displays and plots (P)

bootup Macro executed automatically when VnmrJ is activated (M)

# unlock Remove inactive lock and join experiment (C)

Syntax: unlock(exp number, 'force')

Description: In attempting to join another experiment, the jexp command may abort

claiming the experiment is locked. This feature prevents two users from processing the same experimental data at the same time, which could corrupt the data (a "user" can also be a background operation invoked by the same user, such as in wexp processing). This lock can be left behind if the program or the

computer crashes.

The unlock command removes the lock if it is inactive and joins the unlocked experiment. The command will fail if the lock is still active (i.e., the process that made the lock is still executing) or if the lock was placed on the experiment by a remote host. The latter situation can only occur when one or more nodes are

sharing the same file system (and experimental data).

Arguments: exp number is the number of the experiment from 1 to 9 to be unlocked.

force unlocks an experiment under all circumstances and joins the unlocked

experiment.

Examples: unlock(3)

See also: NMR Spectroscopy User Guide

Related: jexp Join existing experiment (C)

# updatepars Update all parameter sets saved in a directory (M)

Syntax: updatepars(directory)

Description: Corrects saved parameter sets. Starting with VNMR version 4.2, all parameters,

upper limit, lower limit, and step sizes have been tightened. Further additions were made in VNMR 4.3. updatepars searches a directory for parameter and FID files and corrects the procpar files found. This macro overwrites parameters in the current experiment. The corrections applied to the parameter sets are defined by the parfix macro. Because updatepars uses the current

experiment to process the parameter sets, the experiment chosen for running

updatepars should not contain a valuable data set.

Arguments: directory is the name of the directory to be searched.

Examples: updatepars('myparlib')

updatepars('mydata')

See also: NMR Spectroscopy User Guide

Related: parfix Update parameter sets (M)

parversion Version of parameter set (P)

updateprobe Update probe file (M)

Syntax: updateprobe(<probe|'tmplt'><,'system'>)

Description: Updates the current existing probe file or probe template.

Arguments: probe is the probe parameter to update. The default is the current probe

parameter value.

'tmplt' is a keyword to update the local probe template. The default is the

current probe file.

'system' is a keyword to update the system template or probe file, providing you have write permission to the file. The default is to update the local template

or probe file.

Examples: updateprobe

updateprobe('autosw')

updateprobe('autosw','system')

updateprobe('tmplt')

See also: NMR Spectroscopy User Guide

Related: addparams Add parameter to current probe file (M)

getparam Receive parameter from probe file (M)
setparams Write parameter to current probe file (M)

updaterev Update after installing new VnmrJ version (M)

Description: Updates experiment parameters and the global file following installation of a

new VNMR software version. updaterev is called by the makeuser

command during the installation process.

See also: VnmrJ Installation and Administration

updtgcoil Update gradient coil (M)

Applicability: Systems with three-axis gradients.

Description: Creates the gooil parameter, if it does not exist, and sets it to the current value

of the system gradient coil sysgcoil. updtgcoil only executes if gradients

are configured in the system.

The updtgcoil macro is called when a new experiment is joined or new parameters are read into an experiment; however, it is only called at these times if the gcoil parameter exists. If sysgcoil is set to a gradient table name and if the values of sysgcoil and gcoil are different, a message is displayed in the Status window to let the user know that the gradient coil parameters have

been updated.

updtqcoil can be called directly if the user wants to update the parameter set

with the gcoil and gradient table parameters.

See also: NMR Spectroscopy User Guide; User Programming; VnmrJ Imaging NMR

Related: gcoil Read data from gradient calibration tables (P)

sysgcoil System gradient coil (P)

updtparam Update specified acquisition parameters (C)

Description: Enables interactive updating of specified acquisition parameters.

See also: SpinCAD

Related: psgupdateoff Prevent update of acquisition parameters (C)

psgupdateon Enable update of acquisition parameters (C)

usemark Use "mark" output as deconvolution starting point (M)

Description: In some cases it is not possible to produce a line list that is a suitable starting

point for a deconvolution (e.g., lines may overlap so severely that a line list does not find them). In this case, or in any case, the results of a "mark" operation during a previous spectral display (ds) may be used to provide a starting point. If the "mark" has been made with a single cursor, the information in the file mark1d.out contains only a frequency and intensity, and the starting

linewidth is taken from the parameter slw.

If the "mark" is made with two cursors, placed symmetrically about the center of each line at the half-height point, markld.out contains two frequencies and an intensity. In this case, the starting frequency is taken as the average of the two cursor positions; the starting linewidth is taken as their difference (thus

allowing different starting linewidths for each line).

See also: NMR Spectroscopy User Guide

Related: ds Display a spectrum (C)

Spin simulation linewidth (P)

userdir VnmrJ user directory (P)

Description: Stores the full UNIX path of the directory that contains a user's private VnmrJ

files. These include a user's private maclib, menulib, shims, psglib, experiments, etc. This parameter is initialized at bootup by the UNIX

environmental variable vnmruser.

Values: Typical value is /home/vnmr2/vnmrsys

See also: NMR Spectroscopy User Guide

Related: curexp Current experiment directory (P)

systemdir VnmrJ system directory (P)

usergo Experiment setup macro called by go, ga, and au (M)

Description: Called by macros qo, qa, or au before starting an experiment. The user

typically creates usergo as a means to set up general experiment conditions.

See also: NMR Spectroscopy User Guide

Related: au Submit experiment to acquisition and process data (M)

ga Submit experiment to ac acquisition and FT the result (M)

go Submit experiment to acquisition (M)

go\_ Pulse sequence setup macro called by go, ga, and au (M)

userfixpar Macro called by fixpar (M)

Description: Called by the macro fixpar to provide an easy mechanism to customize

parameter sets.

See also: NMR Spectroscopy User Guide

Related: fixpar Correct parameter characteristics in experiment (M)

U



vast1d Set up initial parameters for VAST experiments (M)

vastget Selects and displays VAST spectra (M)

vastglueAssemble 1D datasets into a 2D (or pseudo-2D) datasets (M)vastglue2Assemble 1D datasets into a 2D (or pseudo-2D) datasets (M)vastgoTurn off LC stop flow automation, start VAST automation (M)

vbg Run VNMR processing in background (U)

vf Vertical scale of FID (P)

Edit text file with vi text editor (M)

vibradd Display relative amplitudes of Cold Probe vibrations (M)

vjhelp Display VnmrJ help (U)
vn Start VNMR directly (U)

vnmr Start VNMR in current windowing system (U)
vnmr2sc VNMR to SpinCAD pulse sequence translator (M)

vnmr\_accounting Open Accounting window (U)
vnmrexit Exit from the VNMR system (C)

vnmrj Start VnmrJ (U)

vnmrjcmd() Commands to invoke the GUI popup (C)

vnmrplot Plot files (U)
vnmrprint Print text files (U)
vo Vertical offset (P)

Vertical position of spectrum (P)

vpaction Set initial state for multiple viewports (M)
vpf Current vertical position of FID (P)

vpfi Current vertical position of imaginary FID (P)
vpset3def Set the viewport state to three default viewports (M)

vpsetup Set new viewports (M)
vs Vertical scale (P)

vs2d Vertical scale for 2D displays (P)
vsadj Automatic vertical scale adjustment (M)

vsadj 2 Automatic vertical scale adjustment by powers of 2 (M)
vsadj c Automatic vertical scale adjustment for <sup>13</sup>C spectra (M)
vsadj h Automatic vertical scale adjustment for <sup>1</sup>H spectra (M)

vsprojVertical scale for projections and traces (P)vtairflowVariable Temperature Air Flow (P)vtairlimitsVariable Temperature Air Flow Limits (P)

Variable temperature cutoff point (P)

vtcomplvl Variable temperature compensation for gradient shimming (P)

Variable temperature controller present (P)

Variable temperature wait time (P)

vxr unix Convert VXR-style text files to UNIX format (M,U)

V

vast1d Set up initial parameters for VAST experiments (M)

Applicability: Systems with VAST accessory.

Description: Sets up initial VAST parameters from the /vnmr/stdpar directory or from

the user's stdpar directory if the appropriate file exists there. Any changes made to the files in these directories are reflected in the setup. The file /vnmr/stdpar/vastld.par contains the "default" parameters for VAST spectra and should be modified as needed to produce spectra under desirable conditions. After running vastld, the solvent parameter can be set by choosing it from the list of solvents listed in /vnmr/solvents.

See also: NMR Spectroscopy User Guide

vastget Selects and displays VAST spectra (M)

Applicability: Systems with VAST accessory.

Syntax: vastget(<well>, <well>, ...)>

Description: Selects and displays the spectra from any arbitrary well or wells using the well

label(s) as arguments. the spectra are displayed in a dss stacked plot.

Arguments: well is the well label from which you want to select and display spectra. The

wells are labeled [A->H][1-8].

Examples: vastget('B6','B7','C11','G3')

See also: NMR Spectroscopy User Guide

vastglue Assemble 1D datasets into a 2D (or pseudo-2D) datasets (M)

Applicability: Systems with the VAST accessory.

Syntax: vastglue(<rack,<zone>)

vastglue(<glue order>,<plate>)

Description: Used to artificially reconstruct a 2D datasets from a series of 1D data sets having

similar filenames. It is crucial to ensure that the format of the file names of each of the 1D data sets is identical. vastglue reads in each 1D file, in succession, and adds it to the previous data, but in a 2D format. It assumes that file names are of the format obtained when using the default setting of autoname (autoname=''). If autoname has been redefined, use a macro like vastglue2. Save the resulting reconstructed 2D datasets in the normal

manner using svf.

Arguments: rack is the rack number; the default is 1. If you enter a rack number, you must

also enter a zone number.

zone is the zone number; the default is 1. If you want to specify a zone

number, you must enter a rack number.

glue order is the specific glue order to be defined based on the order defined in a plate\_glue file. If glue order is specified, you can provide a plate number as the second argument and used with the glue order

argument.

See also: NMR Spectroscopy User Guide

Related: autoname Prefix for automation data file (P)

vastglue2 Assemble related 1D datasets into a 2D (or pseudo-2D) datasets (M)

vastq1ue2 Assemble 1D datasets into a 2D (or pseudo-2D) datasets (M)

Applicability: Systems with the VAST accessory

Syntax: vastqlue2<(number) >

Description: Used to artificially reconstruct a 2D data set from a series of 1D datasets having

similar filenames. It is crucial to ensure that the format of the file names of each of the 1D datasets is identical. vastglue2 reads in each 1D file, in succession, and adds it to the previous data, but in a 2D format. It assumes that file names

are of the format obtained using a nondefault setting of autoname

(autoname='filename\_R%RACK:%\_Z%ZONE:%\_S%SAMPLE#:%\_'). This definition must be hard coded into the macro by the user. If autoname has not been redefined, use a macro like vastglue. Save the resulting

reconstructed 2D data set in the normal manner using svf.

Arguments: number is used to specify that only spectra from 1 through number are to be

glued. The default is to glue all the spectra stored in the current directory that

have the proper file name format (from 1 through arraydim).

See also: NMR Spectroscopy User Guide

Related: autoname Prerix for automation data file (P)

vastglue Assemble related 1D datasets into a 2D (or pseudo-2D) data set (M)

# vastgo Turn off LC stop flow automation, start VAST automation (M)

Applicability: Systems with the LC-NMR and VAST accessory

Description: Turns off LC stopped flow use of automation and starts VAST automation run.

#### vbg Run VNMR processing in background (U)

Syntax: (From UNIX) vbg exp number command string refix>

Description: Enables user to perform VNMR tasks in the background. vbg (for "VNMR

background processing") must be run from within a UNIX shell, and no foreground or other background processes can be active in the designated experiment (e.g., if you are working in exp2 in VNMR (in the foreground), you

cannot execute background processing in exp2 as well).

Foreground processing causes a lock file to be placed in the appropriate experiment. The file has a format such as f.1268, where 1268 indicates the process number in the process table (accessed in UNIX by entering the command ps -e). Background processing causes a lock file to be in the appropriate experiment as well. This file has a format such as b.4356, where 4356 indicates the process number. By displaying the files within an experiment, the user can readily determine whether any foreground or

background processes are active in that experiment.

Arguments: exp\_number is the number of the experiment, from 1 to 9, in the user's

directory in which the background processing is to take place.

command\_string is the command string to be executed by VNMR in the background. Double quotes enclosing the string are mandatory (e.g.,

"fn=4096 fn1=2048 wft2da").

prefix is a prefix to be added to the name of the log file, making the name prefix\_bgf.log. The default name is exp\_number\_bgf.log, where exp\_number is the experiment number. The log file is placed in the

experiment in which the background processing takes place.

Examples: (From UNIX) vbg 1 "wft2da bc('f1')"

(From UNIX) vbg 3 "vsadj pl pscale pap page" plotlog

See also: User Programming



# vf Vertical scale of FID (P)

Description: In normalized intensity (nm) mode, vf is the height of the largest FID. In

absolute intensity (ai) mode, vf is a multiplier that is adjusted to produce a desired vertical scale, using the appearance on the display screen as a guide (full

scale on the screen gives full scale on the plotter).

vf can be entered in the usual way or interactively controlled by clicking the middle mouse button in the graphics window during a FID display (click above

the FID to increase vf or below the FID to decrease it).

Values: 1e-6 to 1e9, in mm (in nm mode) or as a multiplier (in ai mode).

See also: NMR Spectroscopy User Guide

Related: ai Select absolute intensity mode (C)

df Display a single FID (C)

nm Select normalized intensity mode (C)

sf Start of FID (P) wf Width of FID (P)

# vi Edit text file with vi text editor (M)

Syntax: vi(file)

Description: Invokes the UNIX text editor vi for editing the file name given. On the Sun

workstation, a popup screen contains the editing window. On the GraphOn terminal, the main screen becomes the editing window. vi is a powerful text editor, but its user interface is limited: the mouse is not used, menus are not available, and status information is virtually nonexistent.

vi operates in three modes: the *command mode* (for moving the cursor and editing text), the *insert mode* (for inserting text into the file), and the *last line* 

mode (for special operations). Each mode is described below.

#### Command mode

vi starts up in the command mode. In this mode, user commands consist mostly of a single character, sometimes in combination with another character, or a number, or both. A number preceding a command typically defines how many times a command should be executed (e.g., 3dd means delete three lines). The commands available include the following:

3G go to the start of line 3

0 (zero) go to the start of the current line

\$ go to the end of the current line

Return or + go to start of next line

- (hyphen) go to start of previous line
Ctrl-d scroll down (forward) half a screen
Ctrl-f scroll forward by a full screen
Ctrl-u scroll up (back) half a screen
Ctrl-b scroll back by a full screen

/expression find next expression and jump to its first character ?expression find previous expression, jump to its first character

n find next expression (from the last search)

N find previous expression (from the last search)

dd delete one line and put it into the buffer

3dd delete three lines and put them into the buffer

dw delete word



х	erase one character forward (under cursor)
X	erase one character backwards (before cursor)
3x	erase three characters forward
rcharacter	erase character and replace with character
ZZ	write if necessary and quit vi
	(period) repeat the last command
u	undo the last command
J	join the next line to the current line
yy or Y	yank one line and put into a buffer (called yank buffer)
р	put contents of yank buffer after the cursor
P	put contents of yank buffer before the cursor
"aY	yank line into buffer a (buffers b to z also available)
"ap	put contents of buffer a below current line
"aP	put contents of buffer a above current line

Because there is no command line, these commands do not show up on the screen but are *executed immediately* (without pressing the Return key).

#### Insert mode

In the insert mode, characters typed on the keyboard (except for the Esc key) show up in the text. The insert mode is entered by typing one of the following commands from the command mode:

a text Esc	append text after the current cursor position
A text Esc	append text to the end of current line
i text Esc	insert text before current cursor position
cw word Esc	change word from current cursor position to end
2cw words Esc	change two words from current cursor position to end
o text Esc	open line below current line and append text
O text Esc	open line above current line and append text

The only way to exit the insert mode is by pressing the Esc key, which leads back to the command mode. Unfortunately, there is no indication on the screen whether  $\forall$  is in the command mode or in the insert mode. Inexperienced users often press the Esc key to make sure they are still in the command mode. The Esc key can also be used to avoid execution of commands that have been typed partially (e.g., the number has been typed, but not the last character).

You can insert special (normally nondisplayable) characters into the text if they are preceded by a Ctrl-v (e.g., entering Ctrl-v Ctrl-q is displayed in the text as ^Q).

#### Changing selected occurrences

The following actions find one or more occurrences of a particular word and change it to another word:

- First, type /word and press Return, where / is a forward slash and word is word you want to change.
- Next, press n as necessary until you reach the occurrence of the word you want to change.
- Finally, type cw newword and press Esc, where newword is replacement word.
- To repeat for another occurrence of word, press n as necessary to scan forward, and then type . (a period) to repeat cw newword (or whatever was the last change)

V

Changing selected occurrences of an expression (one or more words) is similar. To change two words, for example, take the same actions as above but use the command 2cw (or c2w) instead.

#### Last line mode

The last line mode is initiated with a colon; thereafter, commands such as the following can be used (press Return to execute these commands):

:r filename read file named filename (insert in currently open file)
 :w write (save) file
 :w filename write under a new file named filename
 :e filename edit a different file named filename
 :q quit vi (only possible if file has been written back)
 :wq write back file (save changes) and quit vi

:q! quit vi without saving changes

Exiting from vi is accomplished by using the ZZ command in the command

mode, or with the :q, :wq, or :q! commands in the last line mode.

This description lists only a selection of the most important commands. For

This description lists only a selection of the most important commands. For more information on vi, refer to UNIX books and manuals.

Examples: vi(userdir+'/psglib/apt.c')

vi(curexp+'/text')

See also: User Programming

Related: edit Edit a file with user-selectable editor (M)

Edit a parameter and its attributes with vi text editor (M)

macrovi Edit a user macro with the vi text editor (C)
menuvi Edit a menu with the vi text editor (M)
textvi Edit text file of current experiment (M

# vibradd Display relative amplitudes of Cold Probe vibrations (M)

Applicability: Systems with Varian, Inc. Cold Probes

Description: Display the relative amplitudes of the vibrations reaching the probe. Requires a

doped HOD sample.

# vjhelp Display VnmrJ help (U)

Syntax: vjhelp file:///vnmr/jhelp/jhelp.html

Description: Displays the VnmrJ help in a Web browser.

# vn Start VNMR directly (U)

Syntax: (From UNIX) vn <-display Xserver> <-fn font> &

Description: Starts the VNMR application directly without checking the operating system

and attempting to run the window manager.

Arguments: -display Xserver specifies X server display (e.g., hostname: 0.0).

The default is the environment set by the  ${\tt DISPLAY}$  variable.

-fn font specifies the size of the font displayed (e.g., 9x15, 8x13, or 7x13). The default is the font set in the .Xdefaults file. Note that the size

of the font affects the size of the VNMR window.

Examples: vn &

vn -display hostname:0.0 &

vn -font 8x13 &

See also: NMR Spectroscopy User Guide

Related: vnmr Start VNMR (U)

vnmr Starts VnmrJ (U)

Applicability: VnmrJ

Syntax: vnmr

Description: Starts the VnmrJ application

See also: NMR Spectroscopy User Guide

Related: vnmrj Start VnmrJ (U)

#### vnmr2sc VNMR to Spin

#### VNMR to SpinCAD pulse sequence translator (M)

Syntax: vnmr2sc<('sequence\_name'<,rfchannels<,gradchannels>>)>

Description: Converts the pulse sequence pointed to by the seqfil parameter in the current

VNMR parameter set from a C program into a SpinCAD pulse sequence. The conversion result is stored in the local spincad/psglib under the same name as the C pulse sequence (i.e., the name stored in the seqfil parameter), but without the .c extension.

vnmr2sc uses dps output to generate the SpinCAD code, i.e., the pulse sequence must be compiled and must be displayable with dps. Pulse sequences that do not compile with the dps option cannot be translated. For the same reason, vnmr2sc cannot translate features that do not show up in dps. This means that go-time decisions (such as flag-based C if constructs) will not show up in the translated SpinCAD sequence. In such cases, you have two options:

- Translate the sequence several times, once for each of the relevant flag settings. That is, generate several (simpler) SpinCAD pulse sequences from a single C sequence.
- Translate the sequence once (preferably with all options turned on), then manually insert the necessary if statements and other missing elements using SpinCAD.

#### Arguments:

sequence\_name is an optional argument that permits the name of the resulting SpinCAD pulse sequence to be specified. By default, vnmr2sc creates a SpinCAD sequence with the name specified in the seqfil parameter (i.e., the SpinCAD sequence has the same name as the C pulse sequence). sequence\_name is particularly useful if a C sequence is to be translated into multiple SpinCAD sequences; see the examples.

rfchannels is an optional numeric argument specifying the number of rf channels. Use it when you want the SpinCAD sequence to address more rf channels. By default, vnmr2sc determines the number of rf channels from the source sequence. You can only *increase* the number of rf channels. If you specify 0 rf channels, the number of rf channels is left unchanged.

gradchannels is a second optional numeric argument specifying the number of gradient channels or axes. Use it when you want to convert a nongradient sequence to a gradient sequence or when you want the SpinCAD sequence to address more gradient axes than the source sequence. By default, vnmr2sc determines the number of gradient axes from the source sequence. You can only *increase*, not decrease, the number of gradient axes.

Examples: vnmr2sc

setup('H1','CDCl3') hmqc null=0.2 vnmr2sc

null=0 mbond='y' vnmr2sc('hmbc')

V

```
vnmr2sc('gcosy',2,3)
nt=256 vnmr2sc
vnmr2sc(4,1)
vnmr2sc(0,1)
```

See also: SpinCAD Manual

Related: dps Display pulse sequence (C)

spincad Run SpinCAD program (C)

# vnmr accountingOpen Accounting window (U)

Description: Opens a window for creating and maintaining cost accounting data for groups

of users on a spectrometer system. The program accommodates multiple rate schedules for spectrometer usage. A calendar tool can be used to define holidays for holiday rates. There is no limit on the number of rates that can be defined. Multiple printers can be selected.

Any user can view the accounting information (enter cd /vnmr/bin followed by ./vnmr\_accounting), but to update information, the user must have root privileges.

See also: System Installation and Administration

Related: operator Operator name (P)

operatorlogin Sets work space and parameters for the operator (M)

#### vnmrjcmd() Commands to invoke the GUI popup (C)

Syntax: vnmrjcmd('command1','command2',..., parametername)

vnmrjcmd('command1','command2',...<, callback>)

 $\label{population:population} Description: \quad The \, \texttt{vnmrjcmd} \, (\,) \, \, commands \, are \, needed \, in \, order \, to \, invoke \, the \, GUI \, popup \, in \, \, commands \, are \, needed \, in \, order \, to \, invoke \, the \, GUI \, popup \, in \, \, commands \, are \, needed \, in \, order \, to \, invoke \, the \, GUI \, popup \, in \, \, commands \, are \, needed \, in \, order \, to \, invoke \, the \, GUI \, popup \, in \, \, commands \, are \, needed \, in \, order \, to \, invoke \, the \, GUI \, popup \, in \, \, commands \, are \, needed \, in \, order \, to \, invoke \, the \, GUI \, popup \, in \, \, commands \, are \, needed \, in \, order \, to \, invoke \, the \, GUI \, popup \, in \, \, commands \, are \, needed \, in \, order \, to \, invoke \, the \, GUI \, popup \, in \, \, commands \, are \, needed \, in \, order \, to \, invoke \, the \, GUI \, popup \, in \, \, commands \, are \, needed \, in \, order \, to \, invoke \, the \, GUI \, popup \, in \, \, commands \, are \, needed \, in \, order \, to \, invoke \, the \, GUI \, popup \, in \, \, commands \, are \, needed \, in \, order \, to \, invoke \, the \, GUI \, popup \, in \, \, commands \, are \, needed \, in \, order \, to \, invoke \, the \, GUI \, popup \, in \, \, commands \, are \, needed \, in \, order \, to \, invoke \, the \, GUI \, popup \, in \, \, commands \, are \, needed \, in \, order \, to \, invoke \, the \, GUI \, popup \, in \, \, commands \, are \, needed \, in \, order \, to \, invoke \, the \, GUI \, popup \, in \, \, commands \, are \, needed \, invoke \, are \, commands \, are \, needed \, invoke \, are \, commands \, are \, needed \, are \, commands \, are \, needed \, are \, commands \, are \, c$ 

which the user enters the parameters.

Note that vnmrbg and VnmrJ cannot be easily synchronized. When a macro invokes VnmrJ via vnmrj cmd, the VnmrJ thread runs independently and the macro continues on and takes action without otherwise having knowledge of VnmrJ. In order to have events associated with required parameters occur in the proper order, a callback strategy was devised. In simple terms, the vnmrj commands can have a callback string such that when the required parameters are established in VnmrJ, vnmrbg can be re-invoked - the foremost example of this is re-entering the 'go' macro after the parameters are established in VnmrJ.

Examples: Sends parameters one at a time to VnmrJ to be eventually displayed in an entry popup:

```
vnmrjcmd('reqpar','warngui','set', 'real',
parametername)
vnmrjcmd('reqpar','warngui','set', 'string',
parametername)
```

Display a GUI panel listing required parameters sent from vnmrbg in the previous 'set' option above:

```
vnmrjcmd('reqpar','warngui','show')
vnmrjcmd('reqpar','warngui','show', callback)
```

The callback is a command string to be sent back to vnmrbg, if needed. See the reqpartest macro source code for examples of how to use callback.

See also: VnmrJ User Programing

Related go Submit experiment to acquisition (M)

regpartest Tests whether required parameters are set (M)

vnmrexit Exit from the VNMR system (C)

Description: Exits from the VNMR system in a graceful manner by writing parameters and

data to the disk, removing lock files, and restoring the terminal (if on a GraphOn). To provide flexibility when exiting VNMR, the macro <code>exit</code> calls

vnmrexit to exit from VNMR.

CAUTION: When you exit from the VNMR user interface on your X display system,

whether you are using an X terminal or a Sun computer, and whether you are using OpenWindows, CDE, or Motif, you must first exit from any copy of VNMR running on your system. Failure to do this can cause current

parameter values and even current data to be lost.

vnmrj Start VnmrJ (U)

Applicability: VnmrJ

Syntax: vnmrj

Description: Starts the VnmrJ application

See also: NMR Spectroscopy User Guide; VnmrJ Walkup

Related: vnmr Starts VnmrJ (U)

vnmrplot Plot files (U)

Syntax: (From UNIX) vnmrplot <file>

Description: A UNIX command that plots files from inside VNMR commands. To plot a file,

you should use the page command, which uses vnmrplot internally.

Arguments: file is the name of the file to be plotted.

See also: *NMR Spectroscopy User Guide*Related: vnmrprint Print text files (U)

vnmrprint Print text files (U)

Syntax: (From UNIX) vnmrprint printfile <printcap>

<printer type <clear|file>>

Description: A UNIX command installed as part of the VNMR system to print text files. The

printon and printoff commands use vnmrprint to print files.
vnmrprint can also be used to delete a print file or save a print file to a

different name.

Arguments: printfile is the name of the text file to be printed.

printcap is a UNIX printcap entry (e.g. LaserJet\_300) for the printer to print the text file. The default is the printer selected by the -p option of the

UNIX 1p command.

printer\_type is the type of printer from the list of VNMR printers (e.g., LaserJet\_300). printer\_type is required as an argument when it is desired to clear the printer file or save the printer file to another name.

clear is a keyword to delete the current print file. Deleting this file also
requires that the printfile, printcap, and printer\_type arguments

be entered so that clear is the fourth argument.

V

file is the name of the file to use in saving the printfile. If a file with the name specified already exists, it is overwritten. Saving the file also requires that the printfile, printcap, and printer\_type arguments be entered so

that file is the fourth argument.

Examples: vnmrprint /vnmr/psglib/tocsy.c LaserJet 300

vnmrprint myfile LaserJet 300 LaserJet 300 clear

vnmrprint myfile ps PS AR yourfile

See also: NMR Spectroscopy User Guide

Related: printoff Stop sending text to printer and start print operation (C)

printon Direct text output to printer (C)

vnmrplot Plot files (U)

#### vo Vertical offset (P)

Description: Sets the vertical offset, for 1D data sets, of the each spectrum in a stacked

display with respect to the previous spectrum. The parameter ho sets the horizontal offset. For a "left-to-right" presentation, ho is typically negative; for

a "bottom-to-top" presentation, vo is positive.

For 2D data sets, the parameter wc2 sets the distance between the first and last

trace and the vo parameter is inactive.

Values: Number, in mm.

See also: NMR Spectroscopy User Guide

Related: ho Horizontal offset (P)

wc2 Width of chart in second direction (P)

# vp Vertical position of spectrum (P)

Description: Contains vertical position of spectrum with respect to the bottom of the display

or plotter.

Values: -200 to +200, in mm.

See also: NMR Spectroscopy User Guide

Related: vpf Current vertical position of FID (P)

vpfi Current vertical position of imaginary FID (P)

# vpaction Set initial state for multiple viewports (M)

Applicability: VnmrJ Walkup

Description: Sets the initial state for multiple viewports. Used by the viewport editor dialog

under Edit -> Viewports.

See also: User Programming

Related: jcurwin Work space numbers of all viewports (P)

jviewportlabel Work space labels for all viewport buttons (P)

jviewports
Viewport layout (P)

## vpf Current vertical position of FID (P)

Description: Contains the current vertical position of an FID. To create this parameter and

the other FID display parameters axisf, crf, deltaf, dotflag, and vpfi

(if the parameter set is older and lacks these parameters), enter

addpar('fid').

Values: Number, in mm. If vpf=0, the FID is positioned in the middle of the screen.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

axisf Axis label for FID displays and plots (P)

crf Current time-domain cursor position (P)

deltaf Difference of two time-domain cursors (P)

dotflag Display FID as connected dots (P)

vpfi Current vertical position of imaginary FID (P)

Vertical position of spectrum (P)

# vpfi Current vertical position of imaginary FID (P)

Description: Contains the current vertical position of the imaginary part of an FID. To create

this parameter and the other FID display parameters <code>axisf</code>, <code>crf</code>, <code>deltaf</code>, <code>dotflag</code>, and <code>vpf</code> (if the parameter set is older and lacks these parameters),

enter addpar('fid').

Values: Number, in mm. In vpfi=0, the imaginary part is positioned in the middle of

the screen.

vp

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

axisf
Axis label for FID displays and plots (P)

Crf
Current time-domain cursor position (P)

deltaf
Difference of two time-domain cursors (P)

dotflag
Display FID as connected dots (P)

Vp
Vertical position of spectrum (P)

Vpf
Current vertical position of FID (P)

#### vpset3def Set the viewport state to three default viewports (M)

Description: Sets the number of viewports to three, and resets the viewport button labels.

See also: User Programming

Related: jcurwin Work space numbers of all viewports (P)

jviewportlabel Work space labels for all viewport buttons (P)

iviewports
Viewport layout (P)

# vpsetup Set new viewports (M)

Description: Sets the viewports from the selections made in the viewport editor dialog. For

each viewport, it checks the work space number to join, then joins the

appropriate work space.

See also: User Programming

Related: jcurwin Work space numbers of all viewports (P)

jviewportlabel Work space labels for all viewport buttons (P)

jviewports
Viewport layout (P)

# vs Vertical scale (P)

Description: In normalized (nm) mode, vs is the height of the largest peak in the spectrum.

In absolute intensity (ai) mode, vs is a multiplier that is adjusted to produce a desired vertical scale, using the appearance on the display screen as a guide (full scale on the screen gives full scale on the plotter). vs can be entered in the usual

way or interactively controlled by clicking the middle mouse button.

Values: 1e-6 to 1e9, in mm (in nm mode) or as a multiplier (in ai mode).

V

See also: NMR Spectroscopy User Guide

Related: ai Select absolute intensity mode (C)

isadj Adjust integral scale (M)

nm Select normalized intensity mode (C)thadj Adjust threshold for peak printout (M)vsadj Automatic vertical scale adjustment (M)

vsadjc
vsadjc
Automatic vertical scale adjustment by powers of two (M)
Automatic vertical scale adjustment for <sup>13</sup>C spectra (M)
vsadjh
Automatic vertical scale adjustment for <sup>1</sup>H spectra (M)

#### vs2d Vertical scale for 2D displays (P)

Description: Sets a multiplier for 2D spectra and images that is adjusted to produce a desired

vertical scale for display or plotting. vs2d takes the place of vs for 2D data display and can be adjusted by explicitly setting it to a value or by clicking the middle mouse button when pointing to a point on a 2D display. If vs2d does

not exist, it can be created by running par2d.

See also: NMR Spectroscopy User Guide

Related: par2d Create 2D acquisition, processing, and display parameters (M)

vs Select vertical scale (C)

vsproj Adjust vertical scale for projections and traces (M)

## vsadj Automatic vertical scale adjustment (M)

Syntax: vsadj<(height)>

Description: Automatically sets the vertical scale vs in the absolute intensity (ai) mode so

that the largest peak is at the requested height.

Arguments: height is the desired height, in mm, of the largest signal in the displayed

portion of the spectrum. The default is 0.9\* (wc2max-vp-sc2).

Examples: vsadj

vsadj (100)

See also: NMR Spectroscopy User Guide

Related: ai Select absolute intensity mode (C)

isadj Adjust integral scale (M)

thadj Adjust threshold for peak printout (M)

Vs Vertical scale (P)

vsadjc Automatic vertical scale adjustment by powers of two (M)
vsadjc Automatic vertical scale adjustment for <sup>13</sup>C spectra (M)
vsadjh Automatic vertical scale adjustment for <sup>1</sup>H spectra (M)
wc2max Maximum width of chart in second direction (P)

## vsadj2 Automatic vertical scale adjustment by powers of 2 (M)

Syntax: vsadj2<(height)>:scaling factor

Description: Adjusts the vertical scale by powers of two as required for expansion plots (see

aexppl for more information).

Arguments: height is desired height of largest (or largest relevant) signal in displayed

portion of the spectrum. The default is 0.9\* (wc2max-vp-sc2).

scaling\_factor returns to the calling macro the ratio of the new compared

to the old value of vs.

Examples: vsadj2

vsadj2(50):r1

See also: NMR Spectroscopy User Guide

Related: aexppl Automatic expansions plot (M)

isadj Adjust integral scale (M)

sc2 Start of chart in second direction (P)
thadj Adjust threshold for peak printout (M)
vp Vertical position of spectrum (P)

Vs Vertical Scale (P)

vsadj Automatic vertical scale adjustment (M)

vsadjc Automatic vertical scale adjustment for <sup>13</sup>C spectra (M)
vsadjh Automatic vertical scale adjustment for H1 spectra (M)
wc2max Maximum width of chart in second direction (P)

# vsadjc Automatic vertical scale adjustment for 13C spectra (M)

Syntax: vsadjc<(height)>

Description: Functionally the same as the macro vsadj, except excludes solvent and TMS

signals from the carbon spectra for the adjustment of vs.

Arguments: height is desired height of largest (or largest relevant) signal in displayed

portion of the spectrum. The default is 0.9\* (wc2max-vp-sc2).

Examples: vsadjc

vsadjc(wc2max-sc2-wc2-5)

See also: NMR Spectroscopy User Guide

Related: isadj Adjust integral scale (M)

thadj Adjust threshold for peak printout (M)

Vs Vertical Scale (P)

vsadj Automatic vertical scale adjustment (M)

vsadj2 Automatic vertical scale adjustment by powers of two (M) vsadjh Automatic vertical scale adjustment for H1 spectra (M)

# vsadjh Automatic vertical scale adjustment for <sup>1</sup>H spectra (M)

Syntax: vsadjh<(height<,do\_not\_ignore\_solvent>)>

Description: Works as the same as the macro vsadj, except disregards solvent and TMS

signals from proton spectra and, if from the remaining spectrum the highest line is more than three times as high as the second highest line, the spectrum is scaled to this second highest signal (otherwise the highest signal is taken as

relevant).

Arguments: height is desired height of largest (or largest relevant) signal in displayed

portion of the spectrum. If height is 0 or a negative value, it defaults to 0.9\*(wc2max-vp-sc2), which is also the default with no arguments.

do\_not\_ignore\_solvent is any second argument. If present, it signals vsadjh to not ignore the solvent line and regard the solvent line as normal signal (i.e, only exclude the TMS line). This argument was added for the situation where frequently there are high "real" signals at the position of the solvent line. Such signals could otherwise be regarded as solvent line and would

then be ignored. This could then lead to overscaling in the result.

Examples: vsadjh

vsadjh(0.7\*wc2max)

V

See also: NMR Spectroscopy User Guide

Related: isadj Adjust integral scale (M)

Start of chart in second direction (P)
thadj Adjust threshold for peak printout (M)

VS Vertical scale (P)

vsadj Automatic vertical scale adjustment (M)

vsadjc Automatic vertical scale adjustment by powers of two (M) vsadjc Automatic vertical scale adjustment for <sup>13</sup>C spectra (M)

# vsproj Vertical scale for projections and traces (P)

Description: Sets a multiplier that is adjusted to produce a desired vertical scale for

projections or traces of 2D data sets. vsproj can be explicitly adjusted by setting it to a value or by clicking the middle mouse button when pointing at the projection or trace. When interactively adjusting the scale with the mouse, the higher the pointer is in the trace display, the larger the vertical scale. If the parameter does not exist, it can be created by running the par2d macro.

See also: NMR Spectroscopy User Guide

Related: par2d Create 2D acquisition, processing, and display parameters (M)

vs Select vertical scale (C)

vs2d Adjust vertical scale for 2D displays (M)

## vtairflow Variable Temperature Air Flow (P)

Description: This global parameter sets the VT air flow, in 1/min. The adjustment is coarse,

+/- 1 l/min. If there is not enough air flow available it may not reach the

requested value.

Values: 0 - 25

Related: pin Pneumatics router interlock (P)

vtairlimits Variable temperature air flow limits (P)

## vtairlimits Variable Temperature Air Flow Limits (P)

Description: This global parameter determines the range of safe VT air flow, as indicated by

the LEDs on the flow meter. It sets the LEDs on the air flow meter, upper and lower LEDs are orange, in between are green. As long as the ball in the air flow meter is next to a green LED the air flow is considered safe. If the air flow drops or increases such that the ball is next to an orange LED, the pneumatics box will turn the VT Controller off and notify the experiment, provided the switch is in the 'run' position. A bit value of 1 sets an unsafe orange state, a bit value of 0

sets a safe green state. To create the parameter:

create('vtairlimits','integer','global')
setlimit('vtairlimits',1023,0,1,'global')

Examples: a value of 775 or 0x307 will set the two lower and the three upper LEDs

(orange) and clear the remaining 5 in between (green). Note that the upper bits determine the lower LEDs. If the parameter does not exist the value defaults to

0x307 for liquids; 0x200 for solids.

Values: 0 - 1023

Related: pin Pneumatics router interlock (P)

tin Temperature interlock (P) vtairflow Variable temperature air flow (P)

Variable temperature cutoff point (P) vtc

Applicability: Systems with a variable temperature (VT) module.

Description: Sets a VT cutoff point. Above this temperature, VT air flows straight into the

> probe, past the heater, then past the sample. Below this temperature, air goes first through the heat exchange bucket, for cooling by the heat exchange fluid,

and then into the probe and past the heater.

Values: 0 to 50, in degrees celsius. vtc is typically set 5°C higher than the supply gas

used for VT regulation.

See also: NMR Spectroscopy User Guide

Related: Sample temperature (P)

> Temperature interlock (P) tin

Variable temperature compensation for gradient shimming (P) vtcomplvl

Description: Specifies the level of VT compensation used by gradient shimming.

Values: 0, disable VT compensation.

1, enable VT compensation

2, enable VT compensation with extra gradient dephasing.

Related: Get parameters and files for gmapz pulse sequence (M) gmapz

> gmapsys Run gradient autoshimming, set parameters, map shims (M) Number of z-axis shims used by gradient shimming (P) qzsize

temp Sample temperature (P)

vttype Variable temperature controller present (P)

Variable temperature controller present (P) vttype

Description: In the Spectrometer Configuration window, this parameter specifies whether a

> variable temperature (VT) controller is present or not on the system. The value is set using the VT Controller label in the Spectrometer Configuration window.

When entered from command line in VNMR, control of the variable temperature (VT) controller from the current experiment is either engaged (vttype=2) or disengaged (vttype=0). The current state of the variable

temperature (VT) controller is not changed when vttype is set in the command window.

The variable temperature (VT) controller setting in Spectrometer Configuration

is not affected by entering vttype on the command line.

Values: 2 is setting for VT controller (Present choice in Spectrometer Configuration

window).

0 is setting for no VT controller (Not Present choice in Spectrometer

Configuration window).

If temp='some temperature' while vttype=2 and vttype is then changed to Examples:

> vttype=0 on the command line, the variable temperature (VT) controller will continue regulate the sample at the value set by temp. While vttype=0 changes

to temp will have no effect.

See also: VnmrJ Installation and Administration; NMR Spectroscopy User Guide

Related: Display current configuration and possibly change values (M) config

> masvt Type of variable temperature system (P)

V

vtwait Variable temperature wait time (P)

Applicability: Systems with a variable temperature (VT) module.

Description: Sets a time for establishing temperature regulation. If temperature interlock

tin is set and regulation is not established after the time set by vtwait, VNMR displays the message "VT FAILURE" and aborts the experiment.

Values: Number, in seconds, A typical value is 180 seconds.

See also: NMR Spectroscopy User Guide

Related: pad Preacquisition delay (P)

tin Temperature interlock (P)

vxr unix Convert VXR-style text files to UNIX format (M, U)

Syntax: (From VNMR) vxr\_unix(VXR\_file<,UNIX\_file>)

(From UNIX) vxr unix VXR file UNIX file

Description: Converts a VXR-style text file (from a Gemini, VXR, or XL system) to the

UNIX format.

Arguments: VXR file is the name of the input file, which must be a text file.

UNIX file is the name of the output file after conversion. The names of the

input and output files must be different.

Examples: (From VNMR) vxr unix('oldtextfile','newtextfile')

(From UNIX) vxr unix oldtextfile newtextfile

See also: NMR Spectroscopy User Guide

Related: convert data set from a VXR-style system (C,U)

decomp Decompose a VXR-style directory (C)



w Who is using system (C)
walkup Walkup automation (M)

waltz WALTZ decoupling present (P)

wbs Specify action when bs transients accumulate (C)

wbs When block size (P)
wc Width of chart (P)

wc2 Width of chart in second direction (P)

wcmax Maximum width of chart (P)

 wc2max
 Maximum width of chart in second direction (P)

 wdone
 Specify action when experiment is done (C)

 wdone
 Specify action when experiment is done (P)

 werr
 Specify action when error occurs (C)

werr When error (P)

wet Flag to turn on or off wet solvent suppression ((P) Wetld Set up parameters for wet  ${}^{1}H$  experiment (M)

wetdqcosySet up parameters for a WETDQCOSY pulse sequence (M)wetgcosySet up parameters for a WETGCOSY pulse sequence (M)wetghmqcpsSet up parameters for a WETGHMQCPS pulse sequence (M)wetghsqcSet up parameters for a WETGHSQC pulse sequence (M)wetgmqcosySet up parameters for a WETGHSQC pulse sequence (M)wetitSet up and create pulse shapes for Wet1d experiment (M)wetnoesySet up parameters for a WETNOESY pulse sequence (M)

wetpeaks Number of peaks for wet solvent suppression (P)
wetpwxcal Set up parameters for a WETPWXCAL pulse sequence (M)

wettntocsy Set up parameters for a WETTNTOCSY pulse sequence (M)

wetshape Shape for pwwet pulses (P)

wexp Specify action when experiment completes (C)

wexp When experiment completes (P)

wf Width of FID (P)

wf1 Width of interferogram in 1st indirectly detected dimension (P)
wf2 Width of interferogram in 2nd indirectly detected dimension (P)

wft Weight and Fourier transform 1D data (C)
wftld Weight and Fourier transform f<sub>2</sub> for 2D data (C)

wftlda Weight and Fourier transform phase-sensitive data (M)

wftldac Combine arrayed 2D FID matrices (M)
wftld Weight and Fourier transform 2D data (C)

wft2da Weight and Fourier transform phase-sensitive data (M)

wft2dac Combine arrayed 2D FID matrices (M)

wftt3 Process f<sub>3</sub> dimension during 3D acquisition (M)
which Display which command or macro is used (M)
wnt Specify action when nt transients accumulate (C)

wnt When number of transients (P)

W

wp Width of plot in directly detected dimension (P)
wp1 Width of plot in 1st indirectly detected dimension (P)
wp2 Width of plot in 2nd indirectly detected dimension (P)

write Write formatted text to a device (C)

writefid Write numeric text file using a FID element (C)
writeparam Write one of more parameters to a file (C)
writespectrum Write a spectrum to a binary file (C)

wrtp Command string executed after rtp command (P)
wsram Send hardware configuration to acquisition console (C)

wshim Conditions when shimming is performed (P)

wtfile User-defined weighting in directly detected dimension (P)
wtfile1 User-defined weighting in 1st indirectly detected dimension (P)
wtfile2 User-defined weighting in 2nd indirectly detected dimension (P)

wtgen Compile user-written weighting functions (M,U)

wti Interactive weighting (C)

wtia Interactive weighting for 2D absorptive data (M)

wtune Specify when to tune (P)

wtunedone What to do after ProTune tuning is done (P)

wysiwyg Set plot display or full display (P)

# w Who is using system (C)

Description: Displays information about users currently on the system. It functions like the

UNIX command of the same name.

See also: User Programming

#### walkup Walkup automation (M)

Description: Enables using sample changers for continuous "walk-up" operation. Click on

Utilities -> New automation run to run this macro from the VnmrJ Walkup interface. The macro creates a new automation directory each day with the name auto\_yyyy.mm.dd, where yyyy is the year, dd is the day of the month, and mm is the month (e.g., auto\_20040601). The automation directory is saved in a directory specified by the global parameter globalauto. walkup creates the directory globalauto and the parameter globalauto, and then

sets the globalauto parameter.

See also: VnmrJ Walkup

Related: enter Enter sample information for automation run (M,U)

globalauto Automation directory name (P)

#### waltz WALTZ decoupling present (P)

Description: Sets whether system is equipped for WALTZ decoupling. The value is changed

by normal parameter entry rather than using the Spectrometer Configuration

window.

Values: 'n' sets WALTZ decoupling not present.

'y' sets WALTZ decoupling present.

See also: VnmrJ Installation and Administration



#### wbs Specify action when bs transients accumulate (C)

Syntax: wbs(string)

Description: Specifies what action to take when bs transients accumulate. The command

wbs sets the corresponding *parameter* wbs. Using the command, rather than setting the parameter value explicitly, notifies the acquisition process that the associated parameter value has changed. Thus, the desired operation can be

effected even if the experiment has already started.

Arguments: string is a string argument containing the command or macro to be executed

when this event happens. The string must be enclosed in single quotes. If single quotes are required *within* the text string, place a backslash character before each of the interior single quotes ( $\$ '). Maximum length of the string is 256 characters. To turn off wbs processing, enter wbs ('''), where the argument is

two single quotes with no space between.

Syntax: wbs('dg wft')

wbs('mf(3)')

wbs('')

See also: NMR Spectroscopy User Guide

Related: bs Block size (P)

makefid Make a FID element using numeric text input (C)

phfid Zero-order phasing constant for np FID (P)

wbs When block size (P)

werr Specify action when error occurs (C)

wexp Specify action when experiment completes (C)
wnt Specify action when nt transients accumulate (C)

#### wbs When block size (P)

Description: Invokes an action to occur automatically after each bs block of transients is

completed. For example, wbs='wft' results in an automatic weighting and Fourier transformation after each bs transients. To specify no wbs processing, set wbs to the null string. If the acquisition has already started, the wbs

command must be used to change this parameter.

Values: Command, macro, or null string (wbs='', where the value is given by two

single quotes with no space between them).

See also: NMR Spectroscopy User Guide

Related: bs Block size (P)

wbs Specify action when bs transients accumulate (C)

# wc Width of chart (P)

Description: Specifies the width of the chart (plotting or printing area).

Values: 5 to wcmax, in mm.

See also: NMR Spectroscopy User Guide

Related: wc2 Width of chart in second direction (P)

wcmax Maximum width of chart (P)

# wc2 Width of chart in second direction (P)

Description: Specifies width of chart (plotting or printing area) along the second axis (or y

axis) of a 2D contour plot or 2D "stacked display." For plots made in the cutoff mode, wc2 specifies the width of the plotted area along the y-axis.

Values: Width, in mm.

See also: NMR Spectroscopy User Guide

Related: cutoff Data truncation limit (P)

Horizontal offset (P) ho

902 Start of chart in second direction (P) wcmax Maximum width of chart (P)

wc2max Maximum width of chart in second direction (P)

#### wcmax Maximum width of chart (P)

Description: Specifies the maximum width of a chart (plotting or printing area). Set when

plotter or printer is installed.

Values: Width, in mm.

See also: NMR Spectroscopy User Guide Related: Width of chart (P)

> Width of chart in second direction (P) WC2

#### wc2max Maximum width of chart in second direction (P)

Specifies the maximum width of a chart (plotting or printing area) in the second Description:

direction (y-axis). Set when the plotter or printer is installed.

Values: Width, in mm.

See also: NMR Spectroscopy User Guide

Related: wc2 Width of chart in second direction (P)

> wcmax Maximum width of chart (P)

#### Specify action when experiment is done (C) wdone

Syntax: wdone(string)

Description: Specifies the action to take when the experiment is done, after wexp has been

> executed. The wdone command sets the corresponding parameter wdone. Using the command, rather than setting the parameter value explicitly, notifies the acquisition process that the associated parameter value has changed and the

desired operation is effected even if the experiment has already started.

Arguments: The string argument contains the command or macro to be executed when

> the experiment is done. The string must be enclosed in single quotes. If single quotes are required within the text string, place a backslash character before each of the interior single quotes (\'). Maximum length of the string is 256

characters.

' ' (null string) turns off wdone processing.

Related: Specify action when experiment completes (C) wexp

#### wdone Specify action when experiment is done (P)

wdone'<command, macro, or null string >'

Invokes a single action to occur just after wexp is executed. As with wexp, it Description:

> is executed automatically after the experiment is finished, which can occur at the end of a single FID or after the last fid in a multi-FID experiment. To specify no wdone processing, set wdone to the null string. If the acquisition has already started, the wdone command must be used to change the wdone



parameter. For wdone to execute after an experiment finishes and after wexp has executed, start the experiment with the au command.

If the wexp action sets the wdone parameter, the new value of the wdone

parameter will be executed and the old value will be ignored.

Arguments: Any command, macro, or null string (e.g. wdone='').

Related: acquire Acquire data (M)

Submit experiment to acquisition and process data (M)

wexp When experiment completes (P)

# werr Specify action when error occurs (C)

Syntax: werr(string)

Description: Specifies what action to take if an error occurs during acquisition. The

command werr sets the corresponding parameter werr. Using the command, rather than setting the parameter value explicitly, notifies the acquisition process that the associated parameter value has changed. Thus, the desired operation can be effected even if the experiment has already started.

Arguments: string is a string argument containing the command or macro to be executed

when this event happens. The string must be enclosed in single quotes. If single quotes are required *within* the text string, place a backslash character before each of the interior single quotes (\'). Maximum length of the string is 256 characters. To turn off werr processing, enter werr (''), where the argument

is two single quotes with no space between them.

Examples: werr('react')

werr('')

See also: NMR Spectroscopy User Guide

Related: wbs Specify action when bs transients accumulate (C)

werr When error (P)

wexp Specify action when experiment completes (C)
wnt Specify action when nt transients accumulate (C)

#### werr When error (P)

Description: Specifies a macro (e.g., werr='react') that will take appropriate action

when an error occurs during acquisition. To specify no werr processing, set werr to the null string. If the acquisition has already been started, the werr command must be used to change the werr parameter. Arrayed parameter acqstatus provides the error code to werr in acqstatus [1] and acqstatus [2]. For a list of error codes, refer to the description of acqstatus or view the file acq errors in directory /vnmr/manual.

Values: Macro or null string (werr='', where the value is given by two single quotes

with no space between them).

See also: NMR Spectroscopy User Guide

Related: acgstatus Acquisition status (P)

react Recover from error conditions during werr processing (M)

werr Specify action when error occurs (C)

W

wet Flag to turn on or off wet solvent suppression ((P)

Description: Specifies if wet solvent suppression is turned on or off. It is now a standard

option in many liquids pulse sequences, including Wet1d and sequences of

apptype hetero2d and homo2d.

Related: apptype Application type (P)

hetero2dExecute protocol actions of apptype hetero2d (M)homo2dExecute protocol actions of apptype homo2d (M)std1dExecute protocol actions of apptype std1d (M)Wet1dSet up parameters for a WET1D pulse sequence (M)

Wet1d Set up parameters for wet <sup>1</sup>H experiment (M)

Description: Set up parameters for wet <sup>1</sup>H experiment.

wetdqcosy Set up parameters for a WETDQCOSY pulse sequence (M)

Applicability: Systems with LC-NMR accessory.

Description: Sets up for a WETDQCOSY LC-NMR experiment.

See also: NMR Spectroscopy User Guide

wetgcosy Set up parameters for a WETGCOSY pulse sequence (M)

Applicability: Systems with LC-NMR accessory.

Description: Sets up for a WETGCOSY LC-NMR experiment.

See also: NMR Spectroscopy User Guide

wetghmqcps Set up parameters for a WETGHMQCPS pulse sequence (M)

Applicability: Systems with LC-NMR accessory.

Description: Sets up for a WETHMQCPS LC-NMR experiment.

See also: NMR Spectroscopy User Guide

wetghsqc Set up parameters for a WETGHSQC pulse sequence (M)

Applicability: Systems with LC-NMR accessory.

Syntax: wetghsqc('nucleus')

Description: Sets up for a WETGHSQC LC-NMR experiment.

See also: NMR Spectroscopy User Guide

wetgmqcosy Set up parameters for a WETGHSQC pulse sequence (M)

Applicability: Systems with LC-NMR accessory.

Description: Sets up for a WETGMQCOSY LC-NMR experiment.

See also: NMR Spectroscopy User Guide

wetit Set up and create pulse shapes for Wet1d experiment (M)

Applicability: VnmrJ Walkup

Description: A macro to set up and create pulse shapes for a Wetld experiment. It is based

on suppressing the largest N peaks found in a spectrum.

Related: wetpeaks (P)

wetnoesy Set up parameters for a WETNOESY pulse sequence (M)

Applicability: Systems with LC-NMR accessory.

Description: Sets up for a WETNOESY LC-NMR experiment.

See also: NMR Spectroscopy User Guide.

wetpeaks Number of peaks for wet solvent suppression (P)

Applicability: Walkup

Description: Sets the number of peaks to be suppressed by wet solvent suppression for the

 ${\tt Wetld}$  protocol. The  ${\tt wetit}$  macro suppresses the N tallest peaks found in the scout spectrum, where N is specified by  ${\tt wetpeaks}$ . The parameter is set by the

Number of peaks to suppress menu on the Prescan page.

Values: 1 to 7 for DirectDrive or UnityInova systems; 3 for Mercury systems are the

default values.

Related: Wetld Set up parameters for wet 1H experiment (M)

wetit Set up and create pulse shapes for Wetld experiment (M)

wetpwxcal Set up parameters for a WETPWXCAL pulse sequence (M)

Applicability: Systems with LC-NMR accessory.

Description: Sets up for a WETPWXCAL LC-NMR pulse width calibration.

See also: NMR Spectroscopy User Guide

wettntocsy Set up parameters for a WETTNTOCSY pulse sequence (M)

Applicability: Systems with LC-NMR accessory.

Description: Sets up for a WETTNTOCSY LC-NMR experiment.

See also: NMR Spectroscopy User Guide

wetshape Shape for pwwet pulses (P)

Applicability: Systems with LC-NMR accessory.

Description: Sets the name of the shape used for pwwet pulses (e.g., wetshape='wet').

See also: NMR Spectroscopy User Guide

wexp Specify action when experiment completes (C)

Syntax: wexp(string)

Description: Specifies what action to take when the experiment completes. The wexp

command sets the corresponding parameter wexp. Using the command, rather than setting the parameter value explicitly, notifies the acquisition process that the associated parameter value has changed. Thus, the desired operation can be

effected even if the experiment has already started.

Arguments: string is a string argument containing the command or macro to be executed

when the experiment completes. The string must be enclosed in single quotes. If single quotes are required *within* the text string, place a backslash character

before each of the interior single quotes (\'). Maximum length of the string is 256 characters. To turn off wexp processing, enter wexp (''), where

argument is two single quotes with no space between them.

Examples: wexp('wft(\'all\') calcT1')

wexp('')

See also: NMR Spectroscopy User Guide

Related: wbs Specify action when bs transients accumulate (C)

werr Specify action when error occurs (C)
wexp When experiment completes (P)

wnt Specify action when nt transients accumulate (C)

# wexp When experiment completes (P)

Description: Invokes a single action to occur automatically after the experiment is finished,

which can occur after a single FID or after a number of FIDs in a multi-FID experiment. To specify no wexp processing, set wexp to the null string. If the acquisition has already started, the wexp command must be used to change the wexp parameter. For wexp to execute after an experiment finishes, start the

experiment with the au command.

wexp processing occurs after wnt processing in a single FID experiment, and both can be used. wexp also occurs after wnt during the last FID of a multi-FID experiment. Thus, wnt='wft(\'all\')' wexp='calcT1' and wexp='wft(\'all\') calcT1' transforms each FID in a  $T_1$  experiment as it is performed, and when each of the FIDs has been collected, performs the calculation of the  $T_1$  using a hypothetical macro command calcT1. Notice the use of the backslash to include a single quotation mark inside the string.

Values: Command, macro, or null string (wexp='', where the value is given by two

single quotes with no space between them). If the command or macro uses a file name as an argument, specifying an absolute path is best. Be sure the path is

valid and you have the appropriate write permission.

See also: NMR Spectroscopy User Guide

Related: wexp Specify action when experiment completes (C)

wnt When number of transients (P)

au Submit experiment to acquisition and process data (C)

# wf Width of FID (P)

Description: Width of the FID display. This parameter can be entered in the usual way or

interactively controlled by selecting the sf wf button during a FID display.

Values: 0 to the value of at, in seconds. See also: *NMR Spectroscopy User Guide* 

Related: at Acquisition time (P)

dcon Display noninteractive color intensities map (C)

dconi Interactive 2D data display (C)
df Display a single FID (C)

sf Start of FID (P)

vf Vertical scale of FID (P)

wf1 Width of interferogram in 1st indirectly detected dimension (P)
wf2 Width of interferogram in 2nd indirectly detected dimension (P)

#### wf1 Width of interferogram in 1st indirectly detected dimension (P)

Description: Sets the width of the interferogram display in the first indirectly detected

dimension.

Values: 0 to  $(2 \times ni)/sw1$ , in seconds. See also: *NMR Spectroscopy User Guide* 

Related: ni Number of increments in 1st indirectly detected dimension (P)

Start of interferogram in 1st indirectly detected dimension (P)

Spectral width in 1st indirectly detected dimension (P)

wf Width of FID (P)

# wf2 Width of interferogram in 2nd indirectly detected dimension (P)

Description: Sets the width of the interferogram display in the second indirectly detected

dimension.

Values: 0 to  $(2 \times ni2)/sw2$ , in seconds. See also: *NMR Spectroscopy User Guide* 

Related: ni2 Number of increments in 2nd indirectly detected dimension (P)

Start of interferogram in 2nd indirectly detected dimension (P)

Spectral width in 2nd indirectly detected dimension (P)

wf Width of FID (P)

# wfgtest Waveform generator test (M)

Applicability: Systems with a waveform generator.

Description: Retrieves a parameter set and pulse sequence, and compiles the sequence, in

order to set up an experiment to test the waveform generators.

See also: Waveform Generator Kit Installation

# wft Weight and Fourier transform 1D data (C)

Syntax: (1) wft<(<options,><'nf'><,start><,finish><,step>)>

(2) wft('inverse', exp number, expansion factor)

Description: Performs a Fourier transform on one or more 1D FIDs with weighting applied

to the FID. The command executes a left-shift, zero-order phase rotation, and a frequency shift according to the parameters <code>lsfid</code>, <code>phfid</code>, and <code>lsfrq</code>, respectively, on the time-domain data prior to the weighting and Fourier transformation. The type of Fourier transformation to be performed is determined by <code>proc</code>. wft uses the same arguments as the command <code>ft</code>, and

except for weighting, it functions the same as the ft command.

See also: NMR Spectroscopy User Guide

Related: ft Fourier transform 1D data (C)

1sfidNumber of points to left-shift np FID (P)1sfrqFrequency shift of the fn spectrum in Hz (P)phfidZero-order phasing constant for np FID (P)

Type of processing on np FID (P)

# wft1d Weight and Fourier transform f2 for 2D data (C)

Syntax: (1) wft1d(element\_number)

(2) wft1d<(<options,><coefficients>)>

W

Description: Performs the first Fourier transformation along the dimension defined by sw,

with weighting and matrix transposition. This allows the display of t<sub>1</sub>

interferograms with the dcon and dconi commands.

Except for weighting, wftld functions the same as the ftld command. See

the description of ftld for further information.

Arguments: Same as the arguments to ftld. See the ftld command for details.

See also: NMR Spectroscopy User Guide

Related: dcon Display noninteractive color intensity map (C)

dconi Interactive 2D data display (C)

# wftlda Weight and Fourier transform phase-sensitive data (M)

Values: wft1da<(options)>

Description: Processes 2D FID data as well as 2D planes at particular t<sub>1</sub> or t<sub>2</sub> times from a

3D data set for a pure absorptive display.

wftlda differs from ftlda only in that weighting of the time-domain data is performed prior to the Fourier transform. See the description of ftlda for

further information.

Arguments: Same as arguments to ft2da. See the ft2da command for details.

See also: NMR Spectroscopy User Guide

Related: ftlda Fourier transform phase-sensitive data (M)

Fourier transform phase-sensitive data (M)

wft2da Weight and Fourier transform phase-sensitive data (M)

# wft1dac Combine arrayed 2D FID matrices (M)

Syntax: wft1dac<(<mult1>,<mult2>, ,...<multn>)>

Description: Allows the ready combination of 2D FID matrices within the framework of the

2D Fourier transform program. Weighting is performed. This command requires that the data be acquired either without  $f_1$  quadrature or with  $f_1$  quadrature using the TPPI method. wftldac is used with TOCSY (with

multiple mixing times).

Arguments: mult1, mult2, ..., multn are multiplicative coefficients. The nth

argument is a real number and specifies the multiplicative coefficient for the nth

2D FID matrix.

See also: NMR Spectroscopy User Guide

Related: ftldac Combine arrayed 2D FID matrices (M)

Tocsy Set up parameters for TOCSY pulse sequence (M)

wft2dac Combine arrayed 2D FID matrices (M)

# wft2d Weight and Fourier transform 2D data (C)

Syntax: wft2d<(<options,>coefficients)>

Description: Performs a complete 2D transformation with weighting after 2D data has been

acquired. If the first Fourier transformation has already been done using ftld, wftld, ftlda, or wftlda, then the wftld command performs only the

second transform.

For arrayed 2D experiments, a single array element can be transformed and weighted using the array element number as an argument. Interferograms can

be constructed explicitly using the following coefficient table: wft2d(rr1,ir1,rr2,ir2,...ri1,ii1,ri2,ii2,...).

 ${\tt wft2d}\,(\,{\tt 'ptype'}\,,\,\ldots\,)$  transforms P-type spectra, and

wft2d('ntype',...) transforms N-type spectra. The default is N-type.

wft2d also completes a 2D transform that has been started with wft1d (or related commands such as wftlda). The first transform will not be done again if it has already been performed. For phase-sensitive 2D experiments, the coefficients must be applied as part of the first transform (e.g., with wftlda) since the interferograms are formed at that stage. These coefficients need not be repeated when invoking the subsequent transform: a simple wft2d or ft2d can suffice.

See the ft2d command description for further information.

Arguments: Same as the arguments to ft2d. See the ft2d command for details.

Examples: wft2d(1,0,0,0)

wft2d(2)

wft2d(1,0,1,0,0,1,0,1)

wft2d(.67,0,.33,0,0,.67,0,.33)

See also: NMR Spectroscopy User Guide

Related: dconi Interactive 2D data display (C)

> ft1d Fourier transform along f<sub>2</sub> dimension (C)

Fourier transform "halfway" for pure absorption 2D data (M) ft1da

ft2d Fourier transform 2D data (C)

wft1d Weight and Fourier transform f<sub>2</sub> for 2D data (C)

Weight and FT "halfway" for pure absorption 2D data (M) wft1da Weight and transform for pure absorption 2D data (M) wft2da

#### wft2da Weight and Fourier transform phase-sensitive data (M)

Syntax: wft2da<(options)>

Description: Processes 2D FID data, as well as 2D planes at particular t<sub>1</sub> or t<sub>2</sub> times, from a

3D data set for a pure absorptive display.

wft2da differs from ft2da only in that weighting of the time-domain data is performed prior to the Fourier transform. See the description of ft2da for

further information.

Same as used with ft2da. See the ft2da command for details. Arguments:

See also: NMR Spectroscopy User Guide

Related: ft1da Fourier transform phase-sensitive data (M)

> Fourier transform phase-sensitive data (M) ft2da

wft1da Weight and Fourier transform phase-sensitive data (M)

#### wft2dac Combine arrayed 2D FID matrices (M)

Syntax: wft2dac<(<mult1><,mult2>,...<,multn>)>

Allows the ready combination of 2D FID matrices within the framework of the Description:

> 2D Fourier transform program. Weighting is performed. This command requires that the data be acquired either without  $f_1$  quadrature or with  $f_1$ quadrature using the TPPI method. wft2dac is used with TOCSY (with

multiple mixing times).

mult1, mult2, ..., multn are multiplicative coefficients. The nth Arguments:

argument is a real number and specifies the multiplicative coefficient for the nth

2D FID matrix.

See also: NMR Spectroscopy User Guide

Related: ftldac Combine arrayed 2D FID matrices (M)

ft2dac Combine arrayed 2D FID matrices (M)

Tocsy Set up parameters for TOCSY pulse sequence (M)

wftldac Combine arrayed 2D FID matrices (M)

# wftt3 Process f<sub>3</sub> dimension during 3D acquisition (M)

Description: Allows f<sub>2</sub> processing of 3D data to be performed concurrently with data

acquisition. To invoke this function, set wnt='wftt3' and use au to start the acquisition of the 3D data. When wftt3 detects that all the FIDs comprising a

(t1, t2) block have been acquired, it starts up the ft3d program in

background to process that block of FIDs in f<sub>3</sub>.

The 3D processing information file, created by entering set3dproc within VnmrJ, does not need to contain valid  $f_1$  and  $f_2$  processing information but only valid  $f_3$  processing information. Once the  $f_3$  processing is complete, a new 3D information file can be created for the  $f_1$ - $f_2$  processing stages that contains valid  $f_1$  and  $f_2$  processing information.

The non-standard string parameter path3d can be used to specify the directory into which the  $f_3$  processed 3D data is to be stored. Normally, path3d is absent in the parameter set. If this is the case or if path3d='', the  $f_3$ -processed 3D data is stored in the directory curexp/datadir. path3d can be created by

entering create('path3d','string')
setgroup('path3d','display').

See also: NMR Spectroscopy User Guide

Related: au Submit experiment to acquisition and process data (C)

create Create new parameter in a parameter tree (C)

ft3d Perform a 3D Fourier transform (M,U)

getplane Extract planes from a 3D spectral data set (M)

path3d Path to currently displayed 2D planes from a 3D data set (P) select Select a spectrum or 2D plane without displaying it (C)

set3dproc Set 3D processing (C)

setgroup Set group of a parameter in a tree (C) wnt When number of transients (P)

# which Display which command or macro is used (M)

Syntax: which (name)

Description: Searches VnmrJ libraries and then displays on line 3 which VnmrJ command or

macro with the given name will be executed. For macros, which displays the type of macro (user, local, application, or Varian) and the path to the library.

Arguments: name is the name of a command or macro.

Examples: which('wft')
See also: User Programming

Related: exists Determine if a parameter, file, or macro exists (C)

 $\begin{array}{ll} \textbf{hidecommand} & \text{Execute macro instead of command with same name } (M) \end{array}$ 

# wnt Specify action when nt transients accumulate (C)

Syntax: wnt(string)

Description: Specifies what action to take when nt transients accumulate. The wnt

command sets the corresponding parameter wnt. Using the command, rather than setting the parameter value explicitly, notifies the acquisition process that



the associated parameter value has changed. Thus, the desired operation can be effected even if the experiment has already started.

Arguments: string is a string argument containing the command or macro to be executed

when this event happens. The string must be enclosed in single quotes. If single quotes are required within the text string, place a backslash character before each of the interior single quotes ( $\$ '). Maximum length of the string is 256 characters. To turn off wnt processing, enter wnt (''), where the argument is

two single quotes with no space between them.

Examples: wnt('wft(\'all\')')

wnt('')

See also: NMR Spectroscopy User Guide

Related: nt Number of transients (P)

wbs Specify action when bs transients accumulate (C)

werr Specify action when error occurs (C)
wexp When experiment completes (P)
wnt When number of transients (P)

## wnt When number of transients (P)

Description: Invokes a single action to occur automatically after the FID is finished (ct=nt)

or after each FID in a multi-FID experiment involving an arrayed parameter. The most common processing to occur after an FID is an automatic weighting and Fourier transformation (i.e., wnt='wft'); however, this is normally not

needed because the command qa is the exact equivalent of

wnt='wft(\'acq\')' au (i.e., ga sets the wnt action automatically). To specify no wnt processing, set wnt to the null string. If the acquisition has already been started, the wnt command must be used to change this parameter.

Values: Command, macro, or null string (wnt='', where the value is given by two

single quotes with no space between them).

See also: NMR Spectroscopy User Guide

Related: nt Number of transients (P)

wnt Specify action when nt transients accumulate (C)

# wp Width of plot in directly detected dimension (P)

Description: Sets the width of the displayed or plotted region of the spectrum.

Values: Always stored in Hz, but can be entered in ppm by using the p suffix (e.g.,

wp=6p sets the width of plot to 6 ppm).

See also: NMR Spectroscopy User Guide

Related: wp1 Width of plot in 1st indirectly detected dimension (P)

wp2 Width of plot in 2nd indirectly detected dimension (P)

# wp1 Width of plot in 1st indirectly detected dimension (P)

Description: Analogous to the wp parameter except that wp1 applies to the first indirectly

detected dimension of a multidimensional data set.

See also: NMR Spectroscopy User Guide

Related: wp Width of plot in directly detected dimension (P)

wp2 Width of plot in 2nd indirectly detected dimension (P)

#### wp2 Width of plot in 2nd indirectly detected dimension (P)

Description: Analogous to the wp parameter except that wp2 applies to the second indirectly

detected dimension of a multidimensional data set.

See also: NMR Spectroscopy User Guide

Related: wp Width of plot in directly detected dimension (P)

wp1 Width of plot in 1st indirectly detected dimension (P)

# write Write formatted text to a device (C)

Syntax: (1) write('keywords'><,color|pen>

<,'reverse'>,x,y<,template>) <:height>

- (2) write('alpha'|'printer'|'line3'|'error', template)
  (3) write('reset'|'file'|'fileline', file<, template>)
- (4) write('net', host, port, template) '

Description: Writes text to a graphics screen or plotter in a given format (syntax 1), writes

formatted text to another device (syntax 2), clears a file (syntax 3), or writes to a file (syntax 3). The input to the command comes from arguments in

template, which can be parameters such as n1 or pw.

Arguments: 'keywords' identify the output device ('graphics' | plotter') and the drawing mode ('xor'|'normal'|'newovly'|'ovly'|'ovlyC').

- 'graphics' | 'plotter' is a keyword selecting the output device. The default is 'plotter'. The output selected is passed to subsequent pen, move, or draw commands and remains active until a different mode is specified.
- ''xor', 'normal' is a keyword for the drawing mode when using the 'graphics' output device. The default is 'normal'. In the 'xor' mode, if a line is drawn such that one or more points of the line are in common with a previous 'xor' line, the common points are erased. In the normal mode, the common points remain. The mode selected is passed to subsequent pen, move, and draw commands and remains active until a different mode is specified.
- 'newovly', 'ovly', and 'ovlyC' are keywords that specify an interactive drawing capability that is slightly slower than the 'xor' mode but more consistent in color. 'newovly' clears any previous draws, boxes, and writes made with the 'ovly' modes and draws the figure. 'ovly' draws without clearing so that multi-segment figures can be created. 'ovlyC' clears without drawing.

color is the color of the text on a color display: 'red', 'yellow',
'green', 'cyan', 'blue', 'magenta', and 'white'. The default is
'yellow'.

pen is the plotter pen: 'pen1', 'pen2', etc.

'reverse' is a keyword specifying a sideways orientation of the output.

x and y are coordinates on the screen or plotter, in mm.

template is a string of formatting characters along with arguments to those characters. The format is the same as used with the UNIX printf command (for details, see any basic UNIX manual or enter man printf in UNIX). For example, 'pw = %12.5f' is a template to format the parameter pw as fixed point with a field width of 12 spaces and 5 decimal places. The following format characters are implemented:



```
character
                          %C
integer
                          %d
hexadecimal
                          %h
exponential:
                          %e
fixed point
                          %f
exponential/fixed point
                          용q
octal
                          %0
string
                          %S
write a % character
                          use write(...'%s','%')
```

height returns the height of the characters on the screen or plotter. This is useful for positioning multiple-line displays. See the source code of the macro dtext in the maclib directory for an example of usage.

'file' is a keyword to append data to the file specified. Existing data in the file is not overwritten. By writing repeated 'file' calls, a formatted data file can be created (see the fifth example below). Each write command automatically appends a carriage return (line feed) to the end of the string defined by the template argument. To append data without the automatic line feed, use the 'fileline' keyword instead of 'file'. Also, two backslashes (\\) are interpreted as a new line.

'fileline' is a keyword to append data to the file specified, the same as using the 'file' keyword, but without automatically appending a carriage return (line feed) to the end of the data. Any line feeds desired must be explicitly defined (using \n) by the template argument (see the sixth example below). Furthermore, two backslashes (\\) output a single backslash into the file.

file is the name of the file used with the 'reset', 'file', and 'fileline' keywords.

'net' is a keyword for writing to a network program. The host name and port number must be supplied. The host name may also be an IP address, such as 10.190.x.y. The hostname of the local computer is stored in the instrument parameter. The command serverport may be used to get the port number for the currently executing VnmrJ program.

```
Examples: write('graphics',100,100):$ys
```

```
write('plotter',20,180, 'pw = %12.5f',pw)
write('line3', 'Too many arguments')
write('reset','temp1')
write('file','temp1','%10f %10.1f',n1,pw)
write('fileline','temp1','\nEnd of data\n\n')
serverport:$port
write('net',instrument,$port,'banner(`hello`)')
```

See also: User Programming

Related: dtext Display a text file in the graphics window (M)

serverport Returns the value of the VnmrJ network listening port (C)

<sup>&#</sup>x27;alpha' is a keyword to write text to the alphanumeric screen.

<sup>&#</sup>x27;printer' is a keyword to print text on the printer

<sup>&#</sup>x27;line3' is a keyword to write text as a message on line 3.

<sup>&#</sup>x27;error' is a keyword to write text as an error on line 3 and sound a beep.

<sup>&#</sup>x27;reset' is a keyword to clear the file specified.

# writefid Write numeric text file using a FID element (C)

Syntax: writefid(file<,element\_number>)

Description: Writes a text file using data from the selected FID element. The program writes

two values per line—the first is the value from the X (or real) channel and the second is the value from the Y (or imaginary) channel. writefid writes the raw FID data (i.e., FID data processing based on the parameters phfid,

lsfid, and lsfrq does not occur).

Arguments: file is the name of a text file to store the data.

element\_number is an integer larger than 0 for the number of a FID element.

The default is 1.

See also: NMR Spectroscopy User Guide, User Programming

Related: lsfid Number of complex points to left-shift np FID (P)

lsfrq Frequency shift of fn spectrum in Hz (P)
makefid Make a FID element using numeric text input (C)
phfid Zero-order phasing constant for np FID (P)
writespectrum Write a spectrum to a binary file (C)

# writeparam Write one of more parameters to a file (C)

Syntax: writeparam(file,parlist[,tree]['add' | 'replace')

Description: The writeparam command will write one or more parameters to a specified

file. The first argument is the name of the file. The second argument is a list of the names of the parameters to be written. It is a string parameter and the names can be separated either by a space or a comma. The optional third argument is

the tree from which the parameters are copied.

The variable trees are 'current', 'global'. 'processed' and

'systemglobal'.

An optional final argument is the keyword 'add' or 'replace'. The add keyword will cause the parameters to be appended to the specified file.

If they already exists in the file, their values will be updated. The replace keyword will replace the values in the file with the current values from the tree.

The parameters must exist in both the file and the tree

A special case for the replace option occurs when the parameter list is an empty string. In this case, all the parameters in the file will be updated with the current values in the tree. If the parameter does not exist in the tree, no change will be made for that parameter.

This command may be used to store temporary values. For example, you may want to save wexp, wbs, wnt, etc. in order to run a setup acquisition. When it is done, you want to reset the original values. The fread command can to used to read the persenters back into an appropriate parameter tree.

to read the parameters back into an appropriate parameter tree.

Examples: writeparam(curexp+'/mypar','in')

writes the parameter in into the file mypar in the current experiment directory.

writeparam(curexp+'/mypar','sw ct np','processed')

writes the parameters sw, ct, and np from the processed tree into the file

mypar in the current experiment directory.

## writespectrum write a spectrum to a binary file (C)

Description: Writes out the current spectrum as a binary file. The file has no header

information and is written in the native format (little-endian on Linux; big-

endian on Solaris).

writespectrum scales the data by vs, determines the mode selected, ph, av, or pwr, and writes whatever is displayed by ds. The file is written in the current experiment as specN, where N is the element number.

Examples: Write files spec1, spec2, spec3 ... spec{arraydim} in the current experiment

directory:

```
wft $i=0$ while ($i < arraydim) do $i = $i + 1$ select($i) writespectrum endwhile
```

Write the real and imaginary components if phase mode is selected.

```
ph
\dot{s}i=0
$index=''
while ($i < arraydim) do
     $i = $i + 1
     format($i,0,0):$index
     select($i)
     writespectrum
     mv(curexp+'/spec'+$index, curexp+'/
      spec'+$index+'.re')
     rp = rp + 90
     writespectrum
     mv(curexp+'/spec'+$index, curexp+'/
      spec'+$index+'.im')
     rp = rp - 90
endwhile
```

Related: writefid Write numeric text file using a FID element (C)

## wrtp Command string executed after rtp command (P)

Description: Holds the command string that is executed after an rtp command finishes. It is

mostly used to set frequency-dependent parameter values, such as sw, so that

one parameter set can be used on all spectrometers.

Examples: wrtp='setsw(13p,-2p)'

# wsram Send hardware configuration to acquisition console (C)

Syntax: wsram<:\$success>

Description: Sends new hardware configuration information to the acquisition console when

config is used (e.g., to set lockfreq). wsram (write to static RAM) is not

normally entered directly by the user.

Arguments: success returns 1 if wsram is successful, or 0 otherwise.

See also: *VnmrJ Installation and Administration*.

Related: config Display current configuration and possibly change it (M)

lockfreq Lock frequency (P)

# wshim Conditions when shimming is performed (P)

Description: Specifies when automatic shimming is to be used, according to the method

specified by the parameter method.

Values: 'n' sets that no automatic shimming is performed. Even with wshim set to this

value, the shimming procedure specified by the parameter method can be

activated by using the shim command.

'e' or 'exp' sets that automatic shimming is done before data acquisition.

's' or 'samp' sets that automatic shimming is done only at the beginning of the first experiment, following the change of a sample using the automatic sample changer.

'g' sets that automatic shimming using gradient shimming is done only at the beginning of the first experiment, following the change of a sample using the automatic sample changer. The parameter method is ignored. This option is only available in automation and is not used with the go, ga, or au commands.

'f' or 'fid' set automatic shimming is done prior to the data collection of each new array member in a multi-FID experiment.

'fn', where n is an integer, sets shimming is done prior to data collection of every nth FID (e.g., wshim='f16' shims prior to acquiring FIDs 1, 17, 33, etc.). This method is only relevant to arrayed or 2D experiments.

See also: NMR Spectroscopy User Guide

Related: gf Prepare parameters for FID/spectrum display in acqi (M)

method Autoshim method (P)

## wtfile User-defined weighting in directly detected dimension (P)

Description: Set to name of the file containing the user-written weighting function along the

directly detected dimension. This dimension is referred to as the  $\rm f_2$  dimension in 2D data sets, the  $\rm f_3$  dimension in 3D data sets, etc. The shellscript <code>wtgen</code> is used to compile the user-written weighting module into an executable program. The source file is stored in the directory <code>vnmruser+'/wtlib'</code> with a <code>.c</code> file extension. The executable file is in the same directory and has the same name

as the source file but has no file extension.

Values: file is the name of the executable weighting function or the name of the

weighting function text file.

'' (two single quotes with no space in between) indicates wtfile is inactive

and VnmrJ should not look for a user-written weighting function.

See also: NMR Spectroscopy User Guide; User Programming

Related: wtfile1 User-defined weighting in 1st indirectly detected dimension (P)

wtfile2 User-defined weighting in 2nd indirectly detected dimension (P)

wtgen Compile user-written weighting functions (C,U)

# wtfile1 User-defined weighting in 1st indirectly detected dimension (P)

Description: Set to the name of the file containing the user-written weighting function for the

first indirectly detected dimension. This dimension is often referred to as the  $f_1$  dimension of a multidimensional data set. Otherwise, wtfile1 is analogous

to wtfile.

See also: NMR Spectroscopy User Guide; User Programming

Related: wtfile User-defined weighting in directly detected dimension (P)

wtfile2 User-defined weighting in 2nd indirectly detected dimension (P)

# wtfile2 User-defined weighting in 2nd indirectly detected dimension (P)

Description: Set to the name of the file containing the user-written weighting function along

the second indirectly detected dimension. This dimension is often referred to as the f<sub>2</sub> dimension of a multidimensional data set. wtfile2 can be set with wti on the 2D interferogram data. Otherwise, wtfile2 is analogous to wtfile.



See also: NMR Spectroscopy User Guide; User Programming

Related: wtfile User-defined weighting in directly detected dimension (P)

wtfile1 User-defined weighting in 1st indirectly detected dimension (P)

wti Interactive weighting (C)

#### wtgen Compile user-written weighting functions (M,U)

 $Syntax: \ \ (From \ VnmrJ) \ \text{wtgen} \ (\texttt{file} \texttt{<.c} \texttt{>})$ 

(From UNIX) wtgen file<.c>

# Description: Allows compilation of a user-written weighting function that subsequently can be executed from within VnmrJ. wtqen performs the following functions:

• Checks for the existence of the /vnmr/bin directory and aborts if the

- directory is not found.
- Checks for files usrwt.o and weight.h in the /vnmr/bin directory and aborts if either of these two files cannot be found there.
- Checks for the existence of the user's directory and creates this directory if it does not already exist.
- Establishes in the wtlib directory soft links to usrwt.o and weight.h in the /vnmr/bin directory.
- Compiles the user-written weighting function, which is stored in the wtlib directory, link loads it with usrwt.o, and places the executable program in the same directory; any compilation and/or link loading errors are placed in the file errmsg in wtlib.
- Removes the soft links to usrwt.o and weight.h in the /vnmr/bin directory.

The name of the executable program is the same as that for the source file without a file extension (e.g., testwt.c is the source file for the executable file testwt).

Examples: (From VnmrJ) wtgen('testwt')

(From UNIX) wtgen testwt.c

See also: User Programming

Related: wtfile User-defined weighting for t<sub>2</sub> (P)

wtfile1 User-defined weighting for  $t_1$  (P)

wtfile2 User-defined weighting in ni2 dimension (P)

## wti Interactive weighting (C)

Syntax: wti<(element number)>

Description: Allows weighting parameters to be set interactively for both t2 FIDs and t1

interferograms. *wti* responds appropriately to phfid and lsfid for t<sub>2</sub> FIDs and to phfid1 and lsfid1 for t<sub>1</sub> interferograms. The following parameters can be interactively weighted:

- awc, awc1, and awc2 set the additive weighting constant; added in to the weighting function after the lb and sb (or sbs) contributions but before the gf (or gfs) contributions.
- gf, gf1, and gf2 set the Gaussian apodization constant, in seconds.
- gfs, gfs1, and gfs2 set the Gaussian function shift, in seconds; shifts the origin of the Gaussian function; active only if gf (or gf1) is active.

- 1b, 1b1, and 1b2 set the line broadening factor, in Hz; a positive value gives sensitivity enhancement; a negative value gives resolution enhancement.
- sb, sb1, and sb2 set the sinebell time period, in seconds; a negative value give a sine squared bell.
- sbs, sbs1, and sbs2 set the sinebell shift, in seconds; shifts the origin of the sine bell; active only if sb (or sb1) is active.

These parameters can be typed in or changed with the left mouse button in the proper field. The right mouse button turns off the spectrum for a faster response to changes in the weighting function.

Arguments: element\_number specifies which FID element or interferogram trace is to

be used in adjusting the weighting parameters. The default is the currently

active element or trace.

Examples: wti

wti(3)

See also: NMR Spectroscopy User Guide

Related: lsfid Number of complex points to left-shift np FID (P)

1sfid1 Number of complex points to left-shift ni interferogram (P)

phfid Zero-order phasing constant for np FID (P)

phfid1 Zero-order phasing constant for ni interferogram (P) wtia Interactive weighting for 2D absorptive data (C)

# wtia Interactive weighting for 2D absorptive data (M)

Syntax: wtia<(element\_number)>

Description: Allows weighting parameters to be set interactively for both t<sub>2</sub> FIDs and t<sub>1</sub>

interferograms in 2D absorptive data. Refer to the description of the wti

command for further information.

Arguments: element number specifies which FID element or interferogram trace is to

be used in adjusting the weighting parameters. The default is the currently

active trace.

See also: NMR Spectroscopy User Guide

Related: lsfid Number of complex points to left-shift np FID (P)

1sfid1 Number of complex points to left-shift ni interferogram (P)

phfid Zero-order phasing constant for np FID (P)

wti Interactive weighting (C)

# wtune Specify when to tune (P)

Applicability: Liquids, VnmrJ Walkup, Automation

Description: Specify when automatic probe tuning will happen.

Syntax: wtune = 'value1<value2>...'

Values: 's' - when a new sample is inserted

'e' – before each experiment

'o' – change of operator
'v' – change of solvent

't' - change of temperature

'1 ' - change of high band frequency (tn or dn)

'2' - change of low band frequency (dn or tn)



'n' - do not tune, if 'n' is included in argument list, no tuning will occur.

Examples: wtune = 'st12'

The system will tune when a new sample is inserted (s) or the temperature changes for the current or new sample (t) or there is a change in the high band frequency (tn or dn) (1) or there is a change of low band frequency (dn or tn)

(2).

See also: NMR Spectroscopy User Guide and VnmrJ Walkup

Related: tunemethod Method to use for tuning (P)

protune Macro to start ProTune (M)

wtunedone What to do after ProTune tuning is done (P)

wtunedone What to do after ProTune tuning is done (P)

Description: Specific what to do after ProTune tuning is done. This is a local string parameter

that does not exist by default and must be created to specify a command to be

executed after tuning is finished.

See also: NMR Spectroscopy User Guide and VnmrJ Walkup

Related: protune Macro to start ProTune (M)

**Create** new parameter in a parameter tree (C)

wtune Specify when to tune (P)

wysiwyg Set plot display or full display (P)

Description: Sets whether the window display is the same as the plot ("what you see is what

you get," or WYSIWYG) or is expanded to fill the window. This allows the user to scale the image to the full window, making it easier to view. This parameter

is in the user's global parameter file.

Values: 'y' makes the window picture size depend on the current plotter setting.

Scaling the window does not change the ratio of the picture. This value is the

default display condition.

'n' makes the window display expand, giving a full display.

See also: NMR Spectroscopy User Guide



X-zero position of HP pen plotter or Postscript device (P)

x1 X1 shim gradient (P)
x2y2 X2Y2 shim gradient (P)
x3 X3 shim gradient (P)
x4 X4 shim gradient (P)

xdiag Threshold for excluding diagonal peaks when peak picking (P)

xgate Load time counter (M)

Utility macro for study queue experiment manager (M)

xmaction Perform study queue action (M)

xmactionw Perform study queue action for walkup (M)

xmaddreq Add a required protocol before the main protocol (M)

xmcheckreq Check required protocol name (M)

**xmconvert** Convert a temporarily stored study into a submitted study (M)

xmcopyCopy protocols in a study queue (M)xmdeleteDelete nodes in a study queue (M)xmenablepanelEnable or disable a parameter panel (M)

xmendqEnd a chained study queue (M)xmgetattsGet study queue attributes (M)

xmHprescanSet up and process Proton prescans (M)xminitInitialize an imaging study queue (M)xmlockupMove a study queue node up and lock it (M)

xmmakenode Make a new study queue node (M)

Find next prescan or next experiment in study queue (M)

xmprescan Run prescans in study queue (M)

Recover from error conditions during automation study (M)

xmreadnode
Read attributes from a study queue node (M)
xmrtpar
Retrieve parameters from a study queue node (M)

write enterQ entry for a sample for study queue – automation (M)

wmsara Write sample enterQ entry for study queue– imaging (M)

 $\begin{tabular}{ll} \textbf{xmsatfrq} & Processing for Presat experiment $(M)$ \\ \end{tabular}$ 

Action when study queue node is selected (M)

msetattr

Set an attribute for a study queue node (M)

msetatts

Set an attribute for a study queue node (M)

mshowdata

Show data from a study queue node (M)

xmstartnightq Start the night queue (M)

xmsubmit Submit sample(s) to the study queue (M)

wmtime Update the study queue time (M)

check tune parameter during automation (M)
mwerr
Recover from acquisition error in study queue (M)

xmwexp Processing macro for end of acquisition in study queue (M)

wmwritenode Write study queue node attributes (M)
wmwritesq Write study queue node order (M)

xpol Cross-polarization (P)

X

xpolar1 Set up parameters for XPOLAR1 pulse sequence (M)

xy XY shim gradient (P)
xz XZ shim gradient (P)
xz2 XZ2 shim gradient (P)

x0 X-zero position of HP pen plotter or Postscript device (P)

Applicability: Systems with a Hewlett-Packard pen plotter or a Postscript output device.

Description: Adjusts the x-zero position on the chart. Use hpa to adjust x0 (and y0) to place

the numbers in a pleasing position when filled in on the blank lines. x0 is part

of vnmrsys/global and hence common to all experiments.

Values: Number, in mm.

See also: NMR Spectroscopy User Guide

Related: hpa Plot parameters on special preprinted chart paper (C)

y0 Y-zero position of HP plotter or Postscript device (P)

x1 X1 shim gradient (P)

Description: Holds current setting of the X1 radial shim gradient.

Values: If **shimset** is 1, 2, 8, 10: -2048 to +2047, steps of 1, 0 is no current.

If shimset is 3 to 7, 9: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)

x2y2 X2Y2 shim gradient (P)

Description: Holds current setting of the X2Y2 radial shim gradient.

Values: If shimset is 1, 2, 8, 10: -2048 to +2047, steps of 1, 0 is no current.

If **shimset** is 3 to 7, 9: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)

x3 X3 shim gradient (P)

Description: Holds current setting of the X3 radial shim gradient.

Values: If shimset is 1, 2, 10: -2048 to +2047, steps of 1, 0 is no current.

If shimset is 3 to 7, 9: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)

x4 X4 shim gradient (P)

Description: Holds current setting of the X4 radial shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)



xdiag Threshold for excluding diagonal peaks when peak picking (P)

Description: Used by the 112d program to exclude diagonal peaks when peak picking.

To create the 2D peak picking parameters xdiag and th2d in the current

experiment, enter addpar ('ll2d').

Values: Peaks within xdiag Hz of the diagonal will not be picked by 112d. Setting

xdiag to 0.0 will cause 112d to pick all peaks, including diagonal peaks.

See also: NMR Spectroscopy User Guide

Related: addpar Add selected parameters to the current experiment (M)

Automatic and interactive 2D peak picking (C)
th2d Threshold for integrating peaks in 2D spectra (P)

# xgate Load time counter (M)

Applicability: Systems with a solids module.

Syntax: xgate(counts)

Description: Loads the (12-bit) time counter on the pulse programmer with the specified

number of counts and switches the counter to the external time base (the external trigger). On each trigger, the counter counts one unit down, and the next pulse sequence event starts when the count reaches zero. Often that time count will be just 1 (1.0, as the argument must be a floating point number). If the final pulse is to be performed after a longer delay, two options are available:

• Perform a normal delay, followed by the xgate (1.0) call.

• Calculate how many rotor cycles that delay would be (calculation is typically done based on a parameter srate) and then perform xgate with that calculated number of rotor triggers. Be aware that the only number of rotor cycles that can be counted this way is 4096, because the pulse programmer uses a 12-bit counter). At typical rotor speeds of 5 to 10

kHz, the "counted" delay is limited to 0.8 to 0.4 seconds.

Arguments: counts is the number of counts to load into the time counter. The value must

be a floating point number.

Examples: xgate(5.0)

See also: User Guide: Solid-State NMR; VNMR Pulse Sequences

Related: Spinning rate for magic angle spinning (P)

# xm1 Utility macro for study queue experiment manager (M)

Description: A utility macro for setting study queue attributes and other study queue

operations. Usually called from other macros, and not from the command line.

## xmaction Perform study queue action (M)

Applicability: VnmrJ Walkup, Imaging

Description: Perform an action on an experiment node in the study queue. Usually called

from study queue actions, and not from the command line.

# xmactionw Perform study queue action for walkup (M)

Applicability: VnmrJ Walkup

Description: Perform an action on an experiment node in the study queue. Usually called

from other macros, and not from the command line.

X

xmaddreq Add a required protocol before the main protocol (M)

Applicability: VnmrJ Walkup, Imaging

Description: Add a required protocol before the main protocol, when adding a protocol to the

study queue. Usually called from other macros, and not from the command line.

See also: VnmrJ Walkup, VnmrJ Imaging User's Guide

Related: xmmakenode Make a new study queue node (M)

xmcheckreq Check required protocol name (M)

Applicability: VnmrJ Walkup, Imaging

Description: Check if a required protocol exists in the study queue, and return the full path

filename to data, if data has been acquired. Usually called from plotting macros,

and not from the command line.

See also: VnmrJ Walkup, VnmrJ Imaging User's Guide

Related: cqplot Macro to perform generic 2D plot (M)

plot2D Plot 2D spectra (M)

xmconvert Convert a temporarily stored study into a submitted study (M)

Applicability: VnmrJ Walkup, Imaging

Description: Convert a temporarily stored study into a submitted study. Usually only called

from other macros.

See also: VnmrJ Walkup, VnmrJ Imaging User's Guide

Related: xmsubmit Submit sample(s) to the study queue (M)

xmcopy Copy protocols in a study queue (M)

Applicability: VnmrJ Walkup, Imaging

Description: Copy protocols within a study queue. Usually only called from other macros.

See also: VnmrJ Walkup, VnmrJ Imaging User's Guide

Related: xmaction Perform study queue action (M)

xmactionw Perform study queue action for walkup (M)

xmdelete Delete nodes in a study queue (M)

Applicability: VnmrJ Walkup, Imaging

Description: Delete nodes within a study queue. Usually only called from other macros.

See also: VnmrJ Walkup, VnmrJ Imaging User's Guide

Related: sqfilemenu Study queue file menu commands (M)

xmaction Perform study queue action (M)

xmactionw Perform study queue action for walkup (M)

xmenablepanel Enable or disable a parameter panel (M)

Description: Enable or disable a parameter panel. Usually used to disable the Acquire panel

for Imaging applications. Usually called only from a panel.

xmendq End a chained study queue (M)

Applicability: VnmrJ Walkup



Description: End a chained study queue in the Walkup interface. Usually called by other

macros.

See also: VnmrJ Walkup

Related: xmnext Find next prescan or next experiment in study queue (M)

xmgetatts Get study queue attributes (M)

Applicability: *VnmrJ Walkup*, Imaging Description: Get study queue attributes.

See also: VnmrJ Walkup, VnmrJ Imaging User's Guide

Related: xmaction Perform study queue action (M)

xmHprescan Set up and process Proton prescans (M)

Applicability: VnmrJ Walkup

Description: A macro to set up and process prescans for Proton-type experiments (Proton,

Presat, or Wet1d protocols). Usually called from other macros, and not from the

command line.

See also: VnmrJ Walkup, VnmrJ Imaging User's Guide

Related: Hprescan Proton prescan (P))

Apptype macro for Standard 1D experiments (M)

xminit Initialize an imaging study queue (M)

Applicability: Imaging

Description: Initialize an imaging study queue. Usually called from other macros, and not

from the command line.

See also: VnmrJ Imaging User's Guide

Related: sqfilemenu Study queue file menu commands (M)

xmlockup Move a study queue node up and lock it (M)

Applicability: VnmrJ Walkup, Imaging

Description: A macro to move a study queue node up above other completed nodes in the

study queue, and lock it so it cannot be moved. This is usually done just prior to acquisition. Usually called from other macros, and not from the command line.

See also: VnmrJ Walkup, VnmrJ Imaging User's Guide

Related: acquire Acquire data (M)

xmmakenode Make a new study queue node (M)

Applicability: VnmrJ Walkup, Imaging

Description: Create a new node in the study queue. Usually only called by other macros.

See also: VnmrJ Walkup, VnmrJ Imaging User's Guide
Related: locaction Locator action (M)

xmaddreq Add a required protocol before the main protocol (M)

xmnext Find next prescan or next experiment in study queue (M)

Applicability: VnmrJ Walkup

X

Description: Find the next prescan or next experiment in a study queue. It is used for chaining

prescans and experiments. Usually only called by other macros.

See also: VnmrJ Walkup

Related: acquire Acquire data (M)

startqStart a chained study queue (M)xmprescanRun prescans in study queue (M)

xmwexp Processing macro for end of acquisition in study queue (M)

xmprescan Run prescans in study queue (M)

Applicability: VnmrJ Walkup

Description: Run prescans in a study queue. Usually only called by other macros.

See also: VnmrJ Walkup

Related: cqfindz0 Run an experiment to find the value of z0 (M)

gmapshim Start gradient autoshimming (M)

prescan Study queue prescan (P)

xmnext Find next prescan or next experiment in study queue (M)

xmreact Recover from error conditions during automation study (M)

Applicability: VnmrJ Walkup

Description: A macro to recover from error conditions during a study queue automated

acquisition. Usually only called by other macros.

See also: VnmrJ Walkup

Related: acquire Acquire data (M)

react Recover from error conditions during werr processing (M)

xmreadnode Read attributes from a study queue node (M)

Applicability: VnmrJ Walkup, Imaging

Description: Read attributes from a study queue node. Usually only called by other macros

See also: VnmrJ Walkup, VnmrJ Imaging User's Guide.

Related: xmaction Perform study queue action (M)

xmactionw Perform study queue action for walkup (M)

react Recover from error conditions during werr processing (M)

xmrtpar Retrieve parameters from a study queue node (M)

Applicability: Imaging

Description: Retrieve parameters from a study queue node after its parameters have been

customized. Usually only called by other macros.

See also: VnmrJ Imaging User's Guide

Related: xmmakenode Make a new study queue node (M)

xmselect Action when study queue node is selected (M)

xmsample Write enterQ entry for a sample for study queue – liquids (M)

Applicability: *VnmrJ Walkup*, systems with automation such as sample changer or LC-NMR.

Description: Write the information required for a sample in the study queue when the sample

is submitted. Usually only called by other macros.

See also: VnmrJ Walkup

Related: loc Location of sample in tray (P)

xmsubmit Submit sample(s) to the study queue (M)

xmsara Write enterQ entry for a sample for study queue – imaging (M)

Applicability: Imaging

Description: Halt or resume acquisition in the study queue, especially when using multiple

viewports. Usually only called from interface panels.

xmsatfrq Processing for Presat experiment (M)

Applicability: VnmrJ Walkup

Description: A macro to handle processing steps for the Presat experiment. It is optimized for

use with water. Usually only called from other macros.

See also: VnmrJ Walkup

Related: xmHprescan Set up and process Proton prescans (M)

xmselect Action when study queue node is selected (M)

Applicability: VnmrJ Walkup

Description: A macro to specify the action taken when a study queue node is selected by

double-clicking on it. The action depends on the node status, which is Ready for acquisition, Executing, Completed, etc. The macro also runs the macros associated with selecting a study queue node, and saves the parameters of the

current node before retrieving parameters of the selected node.

See also: VnmrJ Walku

Related: xmaction Perform study queue action (M)

xmactionw Perform study queue action for walkup (M)
xmrtpar Retrieve parameters from a study queue node (M)

xmsetatts Set an attribute for a study queue node (M)

Applicability: VnmrJ Walkup, Imaging

Description: Set an attribute for a study queue node.

See also: VnmrJ Walkup, VnmrJ Imaging User's Guide

Related: xmaction Load colors for graphics window and plotters (M)

xmactionw Location of sample in tray (P)

xmsetattr Set an attribute for a study queue node (M)

Applicability: VnmrJ Walkup, Imaging

Description: Set an attribute for a study queue node.

See also: VnmrJ Walkup, VnmrJ Imaging User's Guide

Related: xmaction Load colors for graphics window and plotters (M)

xmactionw Location of sample in tray (P)

xmshowdata Show data from a study queue node (M)

Applicability: VnmrJ Walkup, Imaging

X

Description: A macro that retrieves data from a completed study queue node. In the Walkup

liquids interface, data is also processed if Process data on drag-and-drop from

locator is selected in the System settings dialog in the Utilities menu.

See also: VnmrJ Walkup, VnmrJ Imaging User's Guide

Related: xmselect Action when study queue node is selected (M)

#### xmstartnightq Start the night queue (M)

Applicability: VnmrJ Walkup

Description: Start the night queue. It also is used to initialize the night queue settings in the

Utilities menu.

Examples: xmstartnightq start the night queue

xmstartnightq('at') initialize the night queue settings.

See also: VnmrJ Walkup, VnmrJ Imaging User's Guide

Related: walkup Walkup automation (M)

#### xmsubmit Submit sample(s) to the study queue (M)

Applicability: VnmrJ Walkup, systems with automation such as sample changer or LC-NMR.

Description: Submit the sample or samples selected in the study queue tray. If the Submit

DayQ button below the study queue area is selected, samples are submitted to the DayQ. If the Submit NightQ button is selected, samples are submitted to the

NightQ.

See also: VnmrJ Walkup

Related: xmsample Write enterQ entry for a sample for study queue – automation (M)

#### xmtime Update the study queue time (M)

Applicability: VnmrJ Walkup, systems with automation such as sample changer or LC-NMR.

Description: Update the study queue time for both DayQ and NightQ. Usually only called

from panels or other macros.

See also: VnmrJ Walkup

Related: sqfilemenu Study queue file menu commands (M)

start a chained study queue (M)

studytime Study time (P)

xmsubmit Submit sample(s) to the study queue (M)

#### xmtune Check tune parameter during automation (M)

Applicability: Automation

Syntax: xmtune

Description: Check tune parameters in the study queue during automation and determine if

tuning will occur. Macro is usually called from within automation and not from

the command line.

See also: NMR Spectroscopy User Guide and VnmrJ Walkup

Related: protune Macro to start ProTune (M)

tunemethod Method to use for tuning (P)
wtune Specify when to tune (P)



xmwerr Recover from acquisition error in study queue (M)

Applicability: VnmrJ Walkup, Imaging

Description: Recover from an acquisition error in a study queue when not running

automation. Usually only called from other macros.

See also: VnmrJ Walkup, VnmrJ Imaging User's Guide

Related: acquire Acquire data (M)

xmreact Recover from error conditions during automation study (M)

xmwexp Processing macro for end of acquisition in study queue (M)

Applicability: VnmrJ Walkup, Imaging

Description: A processing macro; runs at the end of acquisition in the study queue and keeps

track of study queue parameters and settings. Usually only called from other

macros.

See also: VnmrJ Walkup, VnmrJ Imaging User's Guide

Related: acquire Acquire data (M)

xmreact Recover from error conditions during automation study (M)

xmwritenode Write study queue node attributes (M)

Applicability: VnmrJ Walkup, Imaging

Description: Write study queue node attributes. Usually only called from other macros.

See also: VnmrJ Walkup, VnmrJ Imaging User's Guide

Related: xmaction Load colors for graphics window and plotters (M)

xmactionw Location of sample in tray (P)

xmsetattr Set an attribute for a study queue node (M)

xmwritesq Write study queue node order (M)

Applicability: VnmrJ Walkup, Imaging

Description: Write the study queue node order. Usually only called from other macros.

See also: VnmrJ Walkup, VnmrJ Imaging User's Guide

Related: xmaction Load colors for graphics window and plotters (M)

xmactionw Location of sample in tray (P)

xpol Cross-polarization (P)

Applicability: Systems with a solids module.

Description: Selects cross-polarization or direct polarization in solid-state NMR experiments

such as XPOLAR1.

Values: 'n' sets the experiment for direct polarization.

'y' sets the experiment for cross-polarization.

See also: User Guide: Solid-State NMR

Related: xpolar1 Set up parameters for XPOLAR1 pulse sequence (M)

xpolar1 Set up parameters for XPOLAR1 pulse sequence (M)

Applicability: Systems with solids modules.

X

Description: Sets up the solid-state NMR cross-polarization experiment XPOLAR using the

parameters. Otherwise, xpolar1 contains the same functionality as xpolar.

See also: User Guide: Solid-State NMR

Related: hsrotor Display rotor speed for solids operation (P)

rotorsync Rotor synchronization (P)

xy XY shim gradient (P)

Description: Holds current setting of the XY radial shim gradient.

Values: If **shimset** is 1, 2, 8, 10: -2048 to +2047, steps of 1, 0 is no current.

If shimset is 3 to 7, 9: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)

xz XZ shim gradient (P)

Description: Holds current setting of the XZ radial shim gradient.

Values: If shimset is 1, 2, 8, 10: -2048 to +2047, steps of 1, 0 is no current.

If **shimset** is 3 to 7, 9: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)

xz2 XZ2 shim gradient (P)

Description: Holds current setting of XZ2 radial shim gradient.

Values: If shimset is 2, 8: -2048 to +2047, steps of 1, 0 is no current.

If shimset is 3 to 7, 9: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)



Y-zero position of HP pen plotter or Postscript device (P)

y1 Y1 shim gradient (P)
y3 Y3 shim gradient (P)
y4 Y4 shim gradient (P)
yz YZ shim gradient (P)
yz2 YZ2 shim gradient (P)

y0 Y-zero position of HP pen plotter or Postscript device (P)

Applicability: Systems with a Hewlett-Packard pen plotter or a Postscript output device.

Description: Adjusts the y-zero position on the chart. Use hpa to adjust y0 (and x0) to place

numbers in a pleasing position when filled in on the blank lines. yo is part of

vnmrsys/global; therefore, it is common to all experiments.

Values: Number, in mm.

See also: NMR Spectroscopy User Guide

Related: hpa Plot parameters on special preprinted chart paper (C)

X-zero position of HP plotter or Postscript device (P)

y1 Y1 shim gradient (P)

Description: Holds current setting of the Y1 radial shim gradient.

Values: If shimset is 1, 2, 8, 10: -2048 to +2047, steps of 1, 0 is no current.

If **shimset** is 3 to 7, 9: –32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)

y3 Y3 shim gradient (P)

Description: Holds current setting of the Y3 radial shim gradient.

Values: If shimset is 1, 2, 10: -2048 to +2047, steps of 1, 0 is no current.

If shimset is 3 to 7, 9: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)

y4 Y4 shim gradient (P)

Description: Holds current setting of the Y4 radial shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)

yz YZ shim gradient (P)

Description: Holds current setting of the YZ radial shim gradient.

Y

Values: If **shimset** is 1, 2, 8, 10: -2048 to +2047, steps of 1, 0 is no current.

If shimset is 3 to 7, 9: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)

yz2 YZ2 shim gradient (P)

Description: Holds current setting of the YZ2 radial shim gradient.

Values: If shimset is 2, 8: -2048 to +2047, steps of 1, 0 is no current.

If shimset is 3 to 7, 9: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)

# Z

z	Add integral reset point at cursor position (C)
<b>z</b> 0	Z0 field position (P)
z1	Z1 shim gradient (P)
zlc	Z1C shim gradient (P)
<b>z</b> 2	Z2 shim gradient (P)
z2c	Z2C shim gradient (P)
z2x2y2	Z2X2Y2 shim gradient (P)
z2x3	Z2X3 shim gradient (P)
z2xy	Z2XY shim gradient (P)
z2y3	Z2Y3 shim gradient (P)
<b>z</b> 3	Z3 shim gradient (P)
z3c	Z3C shim gradient (P)
z3x	Z3X shim gradient (P)
z3x2y2	Z3X2Y2 shim gradient (P)
z3x3	Z3X3 shim gradient (P)
z3xy	Z3XY shim gradient (P)
z3y	Z3Y shim gradient (P)
z3y3	Z3Y3 shim gradient (P)
<b>z4</b>	Z4 shim gradient (P)
z4c	Z4C shim gradient (P)
z4x	Z4X shim gradient (P)
z4x2y2	Z4X2Y2 shim gradient (P)
z4xy	Z4XY shim gradient (P)
z4y	Z4Y shim gradient (P)
<b>z</b> 5	Z5 shim gradient (P)
z5x	Z5X shim gradient (P)
z5y	Z5Y shim gradient (P)
<b>z</b> 6	Z6 shim gradient (P)
z7	Z7 shim gradient (P)
<b>z</b> 8	Z8 shim gradient (P)
zeroneg	Set all negative intensities of 2D spectra to zero (C)
zoom	Adjust display to given width (M)
zx2y2	ZX2Y2 shim gradient (P)
zx3	ZX3 shim gradient (P)
zxy	ZXY shim gradient (P)
zy3	ZY3 shim gradient (P)

#### z Add integral reset point at cursor position (C)

Syntax: z < (reset1, reset2, ...) >

Description: Resets the integral to zero at the point marked by the displayed cursor. The

command cz removes all such integral resets and it should generally be used

before starting to enter a series of integral zeros (resets). The resets are stored

as frequencies and do not change if fn is changed.

Arguments: reset1, reset2, ... are reset points entered, in either Hz or ppm. The

default is the cursor position). Reset points can be entered in any order.

Examples: z

z(7.5\*sfrq,5\*sfrq,2.5\*sfrq,0.1\*sfrq)

See also: NMR Spectroscopy User Guide

Related: CZ Clear integral reset points (C)

dlni Display list of normalized integrals (C)

ds Display a spectrum (C)

Fourier number in directly detected dimension (P)

nli Find integral values (C)

#### z0 Z0 field position (P)

Description: Holds current setting of the Z0 setting. The value of z0 can be set by su.

lockfreq can be used to find the lock signal or resonance. To use the lock frequency, deactivate z0 by typing the statement z0 = 'n'. To activate z0,

enter z0 = 'y'.

Values: If shimset is 1, 2, 8, 10: -2048 to +2047, steps of 1, 0 is no current.

If shimset is 3 to 7, 9: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: lockfreq Lock frequency (P)

Submit a setup experiment to acquisition (M)

#### z1 Z1 shim gradient (P)

Description: Holds current setting of the Z1 axial shim gradient.

Values: If shimset is 1, 2, 8, 10: -2048 to +2047, steps of 1, 0 is no current.

If **shimset** is 3 to 7, 9: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)

#### z1c Z1C shim gradient (P)

Description: Holds current setting of the Z1C axial shim gradient.

Values: If shimset is 1, 2, 10: -2048 to +2047, steps of 1, 0 is no current.

If **shimset** is 5 or 9: –32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)

#### z2 Z2 shim gradient (P)

Description: Holds current setting of the Z2 axial shim gradient.

Values: If shimset is 1, 2, 8, 10: -2048 to +2047, steps of 1, 0 is no current.

If shimset is 3 to 7, 9: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)

z2c Z2C shim gradient (P)

Description: Holds current setting of the Z2C axial shim gradient.

Values: If  $\mathtt{shimset}$  is 1, 2, 10: -2048 to +2047, steps of 1, 0 is no current.

If **shimset** is 5 or 9: –32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)

z2x2y2 Z2X2Y2 shim gradient (P)

Description: Holds current setting of the Z2X2Y2 radial shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

z2x3 Z2X3 shim gradient (P)

Description: Holds current setting of the Z2X3 radial shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

z2xy Z2XY shim gradient (P)

Description: Holds current setting of the Z2XY radial shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

z2y3 Z2Y3 shim gradient (P)

Description: Holds current setting of the Z2Y3 radial shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

z3 Z3 shim gradient (P)

Description: Holds current setting of the Z3 axial shim gradient.

Values: If shimset is 1, 2, 8, 10: -2048 to +2047, steps of 1, 0 is no current.

If **shimset** is 3 to 7, 9: –32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)

z3c Z3C shim gradient (P)

Description: Holds current setting of the Z3C radial shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

z3x Z3X shim gradient (P)

Description: Holds current setting of the Z3X radial shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

z3x2y2 Z3X2Y2 shim gradient (P)

Description: Holds current setting of the Z3X2Y2 radial shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

z3x3 Z3X3 shim gradient (P)

Description: Holds current setting of the Z2X3 radial shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

z3xy Z3XY shim gradient (P)

Description: Holds current setting of the Z3XY radial shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

z3y Z3Y shim gradient (P)

Description: Holds current setting of the Z3Y radial shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

z3y3 Z3Y3 shim gradient (P)

Description: Holds current setting of the Z3Y3 radial shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

z4 Z4 shim gradient (P)

Description: Holds current setting of the Z4 shim gradient.

Values: If **shimset** is 1, 2, 8, 10: -2048 to +2047, steps of 1, 0 is no current.

If shimset is 3 to 7, 9: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)

z4c Z4C shim gradient (P)

Description: Holds current setting of the Z4C shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

z4x Z4X shim gradient (P)

Description: Holds current setting of the Z4X shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

z4x2y2 Z4X2Y2 shim gradient (P)

Description: Holds current setting of the Z4X2Y2 radial shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

z4xy Z4XY shim gradient (P)

Description: Holds current setting of the Z4XY radial shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

z4y Z4Y shim gradient (P)

Description: Holds current setting of the Z4Y shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

z5 Z5 shim gradient (P)

Description: Holds current setting of the Z5 axial shim gradient.

Values: If shimset is 2, 10: -2048 to +2047, steps of 1, 0 is no current.

If shimset is 3 to 7, 9: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)

z5x Z5X shim gradient (P)

Description: Holds current setting of the Z5X radial shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

z5y Z5Y shim gradient (P)

Description: Holds current setting of the Z5Y radial shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

z6 Z6 shim gradient (P)

Description: Holds current setting of the Z6 axial shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

z7 Z7 shim gradient (P)

Description: Holds current setting of the Z7 axial shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

z8 Z8 shim gradient (P)

Description: Holds current setting of the Z8 shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

zeroneg Set all negative intensities of 2D spectra to zero (C)

Description: Sets to zero all negative intensities of 2D-J spectra.

See also: NMR Spectroscopy User Guide

Related: foldj Fold J-resolved 2D spectrum about f<sub>1</sub>=0 axis (C)

rotate Rotate 2D data (C)

zoom Adjust display to given width (M)

Syntax: zoom(width)

Arguments:

Description: Adjusts the display limits. It is useful in the display of powder patterns after

split has been used. zoom both zooms in and out from the current display. width is the total display width, in Hz. Display limits are set to ±width/2.

See also: NMR Spectroscopy User Guide

Related: split Split the difference between two cursors (M)

zx2y2 ZX2Y2 shim gradient (P)

Description: Holds current setting of the ZX2Y2 shim gradient.

Values: If shimset is 2, 8: -2048 to +2047, steps of 1, 0 is no current.

If shimset is 3 to 7, 9: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)

zx3 ZX3 shim gradient (P)

Description: Holds current setting of the ZX3 shim gradient.

Values: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

zxy ZXY shim gradient (P)

Description: Holds current setting of the ZXY shim gradient.

Values: If shimset is 2, 8: -2048 to +2047, steps of 1, 0 is no current.

If shimset is 3 to 7, 9: -32768 to +32767, steps of 1, 0 is no current.

See also: NMR Spectroscopy User Guide

Related: shimset Type of shim set (P)

zy3 ZY3 shim gradient (P)

Description: Holds current setting of the ZY3 shim gradient.

Values: -32768 to +32767, steps of 1, 0 as no current.

convert compressed data to standard format,

0 1 1	225
Symbols	copy peak picking file to another file, 321
& (ampersand) character, 537	create experiment, 94
· ·	create new parameters, 514
.talk file, 317	create parameters, 393
/(slash) character, 316	. ,
? (question mark) character, 462	create peak picking parameters, 397
@ (at) character, 316	cross-relaxation experiment, 589
	data display, 129
	display a spectrum, 180
Numerics	display FIDs, 141
13C - HSOC	display resolution, 62
13C gHSQC exp, setting up parameters for, 259	display spectra in whitewash mode, 181
<sup>13</sup> C HMQC exp, changing parameters for, 284	draw grid on 2D display, 268
13C HMQCTOXY exp, changing params for, 284	$f_2$ ridges, 230
<sup>13</sup> C HSQC exp, changing parameters for, 288	find and integrate peaks, 318
<sup>13</sup> C HSQCTOXY exp, changing parameters for, 288	first point multiplier, 95, 231
<sup>15</sup> N gHMQC exp, setting up parameters for, 259	first-order phase set to zero, 90
<sup>15</sup> N gHSQC exp, setting up parameters for, 259	
<sup>15</sup> N HMQC exp, changing parameters for, 283, 284	fold J-resolved spectrum, 228
<sup>15</sup> N HMQCTOXY exp, changing params for, 284	Fourier transform 2D data, 236
15N HSQC exp, changing parameters for, 288	Fourier transform arrayed 2D FID data, 236
<sup>15</sup> N HSQCTOXY exp, changing parameters for, 288	general setup, 514
16-bit integer precision, 168	heteronuclear 2D-J, 282
18, 597	heteronuclear chemical shift correlation, 282
	homonuclear J-resolved 2D, 284
18 shim configuration, 597	horizontal axis selection, 591
1st indirectly detected dimension	INADEQUATE pulse sequence, 296
absolute value display mode, 77	incremented delay, 124
additive weighting constant, 78	intensity of spectrum at a point, 350
clear reference line, 114	interactive weighting, 637, 638
cursor difference, 139	interleaving control, 294
cursor position, 94, 111	J-correlation experiment, 589
data display mode, 159	
first-order phase, 328	LC-NMR acquisition parameters, 314
Fourier number, 227	normalization, 370
Gaussian function, 257	number of increments of evolution time, 367
Gaussian shift constant, 258	parameter creation, 49
incremented delay, 124	peak integration threshold, 586
line broadening, 313	peak picking display control, 321
number of increments of evolution time, 367	peak picking parameters, 49
phased spectra display mode, 389, 411	phase selection, 412
power display mode, 455	plot 2D peak picking results, 425
1 1 5 /	plot grid over 2D plot, 422
reference line frequency, 488	plot heteronuclear J-resolved 2D spectra, 422
reference line position, 487	plot homonuclear J-resolved 2D spectra, 423
scale spectral width, 509	plot X,H-correlation 2D spectrum, 424
set frequency referencing, 530	plotter units conversion, 291
set reference line, 491	processing mode for 2D data, 432
sinebell constant, 506	processing parameter group, 145
sinebell shift constant, 507	project 2D data onto axis, 444
spectra width, 574	pseudo-echo weighting parameters, 447
start of plot, 547	rotate 2D data, 494
user-defined weighting, 636	
width of plot, 631	search data set for maximum intensity, 406
zero-order phase, 495	select for processing, 251
200-kHz receiver option, 168	set scaling factor, 509
2D display, showing, 244	sinebell weighting, 543
2D DOSY display	spectra plotting, 417
building up, 227	spectra plotting in whitewash mode, 418
= -:	spectra processing, 442
2D experiments	spectral drift correction, 128
acquire and Fourier transform, 247	stacked spectra display, 189, 191, 193
axis labels, 79	start of chart in second direction, 508
baseline correction, 82	submit to acquisition, 264
color intensity map, 129	symmetrize INADEQUATE data, 228
combine arrayed FID matrices, 236, 240	$t_1$ dimension, $\frac{336}{1}$
control dconi display, 131	1

type of data processing, 441	parameter creation, 49
vertical offset of traces, 612	path to 2D planes, 399
volume value, 297	phase cycling type, 412
weight and Fourier transform, 627	plot peak picking on 2D plane, 425
weight and Fourier transform 2D data, 628	plot series of 3D planes, 429
weight and Fourier transform phase-sensitive	process f <sub>3</sub> dimension, 630
data, 628, 629	processing coefficient file, 515
2D experiments setup, 514	region selective 3D processing, 451
2D spectra, plotting, 427	reset parameters after partial transform, 481
2D spectra, setting negative intensities of, 658	select 2D plane without displaying, 511
<sup>2</sup> H chemical shift, 546	selective 2D processing, 239
2nd evolution dimension	selective transformation, 238
set spectral width, 532	set 3D processing, 515
2nd indirectly detected dimension	spectral dc correction, 548
absolute value display mode, 77	$t_1$ and $t_2$ dimensions, 337
additive weighting constant, 78	terminate 3D FT process, 307
clear reference line, 114	time-domain dc correction, 218
cursor difference, 139	type of data processing, 442
cursor position, 94, 112	weight and FT phase-sensitive data, 628, 629
data display mode, 159	3rd indirectly detected dimension
first-order phase, 329	incremented delay, 125
Fourier number, 227	number of increments of evolution time, 368
Gaussian function, 257	spectral width, 575
Gaussian shift constant, 258	3rd rf channel
incremented delay, 124	create parameters, 224
line broadening, 314	display group of parameters, 146
number of increments of evolution time, 367	display template for parameters, 393
phased spectra display mode, 411	parameter retrieval, 49
power mode processing, 456	4D experiments
reference line frequency, 488	create acquisition parameters, 393
reference line position, 487	incremented delay, 125
right phase, 495	number of increments of evolution time, 368
scale spectral width, 509	parameter creation, 49
set frequency referencing, 531	phase cycling type, 413
set reference line, 492	4th rf channel
sinebell constant, 506	create parameters, 224
sinebell shift constant, 507	display group of parameters, 146
spectral width, 575	5th rf channel
start of plot, 547	create parameters, 225
user-defined weighting, 636	90-degree pulse, 387
width of plot, 632	90-degree pulse width, 454
32-bit integer precision, 168	
3D experiments	
3D plane index selected, 296	$\boldsymbol{A}$
3D plane projection selected, 296	aborting acquisition
3D plane type currently displayed, 418	aborting acquisition
axis labels, 79	with error, 37
create experiment, 94	with no error, 276
create parameters, 393	absolute intensity display mode, 52
display 2D color map of plane from 3D data,	absolute intensity group, 52
172.	absolute-value 2D experiment, 412
display 2D projection plane from 3D data, 173	absolute-value COSY pulse sequence, 478
	absolute-value data display mode, 158, 159
display group of parameters, 146	absolute-value display mode, 76
display next 3D plane, 367	absolute-value MQF COSY parameter set, 264
display previous 3D plane, 437	absolute-value ROESY parameter set, 270
display series of 3D planes, 189	accounting program, 610
extract planes from 3D spectral data, 253	acq_errors file, 44
$f_{3 \text{ ridges}}, \frac{230}{}$	acqaddr parameter, 231
find and integrate peaks on 2D plane, 318	acqbin directory, 513
first point multiplier, 231	acqfil directory, 212, 580
Fourier transform 3D FID into 3D data, 240	Acqueter window, 42
Fourier transform arrayed 3D data sets, 236	
horizontal axis selection, 591	acquisition
incremented delay, 124	abort with error, 37
N-type display, 374	abort with no error, 276
number of increments of evolution time, 367	acquire FID with no processing, 264
	acquisition parameter arrays, 125

action when bs transients accumulate, 621	delete experiment, 98
action when error occurs, 623	interactive mode, 48
action when specified transients accumulate,	subtract FID, 567
630	subtract spectrum, 555
automated proton and carbon, 276	additive weighting constant, 78, 637
automated proton and COSY, 278	allocateWithId procedure, 362
automated proton, carbon, DEPT, 277	AM data conversion, 104
automated proton, carbon, HETCOR, 277	ampersand (&) character, 43, 44
carbon, 89	amplifier band in use, 483
carbon and APT automatically, 91	amplifier mode control, 54
carbon and DEPT automatically, 92	amplifier type, 54
create 2D parameters, 393	AMX data conversion, 104
create 3D acquisition parameters, 393	AnalogPlus digital filter, 136
create 4D acquisition parameters, 393	analyze.inp file, 55, 209, 211
data points to acquire, 63	analyze.list file, 56, 103, 578, 579
data points to be acquired, 373	analyze.out file, 56, 103, 211, 407
date data is acquired, 126	AP Interface Type label, 101
delay before acquisition, 53	application mode, 60
determine if active for experiment, 208	Apptype macro for dosy 2D experiments, 166
display status information, 542	Appropriate and 277
DSP type, 188	APT acquisition, 277
estimate acquisition time, 587	APT spectra
fluorine, 214	plot automatically, 419
hardware values, 521	process automatically, 60
interactive display, 40	arc cosine calculation, 39
LC-NMR 2D parameters, 314	arc sine of number, 62
make equal to time requested, 587	arc tangent calculation, 63
number of scans, 373	arc tangent of two numbers (Y,X), 63
number of transients, 373	argument, type, return identifier for, 595
oversampling factor, 381	arrayed 1D spectra, 562
1 0 ,	
perform Autoshim experiment, 538	processing, 443
perform experiment, 65	processing and plotting, 75
phosphorus, 388	arrayed 2D FID matrices, 628, 629
read hardware values, 468, 540	arrayed experiment
recover from error, 466	control interleaving, 294
resume paused queue, 482	arrayed parameter, returning number of elements in
resume stopped acquisition, 465	an, 544
stop acquisition, 504	arrayed parameters
stopped by temperature interlock, 588	enter as linearly spaced, 60
submit Autolock experiment, 323	order and precedence, 61
submit change sample, Autoshim experiment,	arraying LP parameters, 332
504	assign sysgcoil, 519
submit setup experiment, 566	asynchronous decoupler mode, 154
submit spin setup experiment, 549	attached proton test, 60
	attenuator
time to acquire FID, 63	
trigger signals to wait, 374	coarse type, 91
acquisition bus trap, 46	control, 176
acquisition computer	fine, 216
block size, 86	upper safety limit, 174
resetting, 38	attributes of parameters, 151
Acquisition Controller board, 494	audio filter board, 67
acquisition parameters group, 145	Audio Filter Type label, 67, 101
acquisition queue, resume after pause, 482	audio filters bandwidth, 216
acquisition status, 44	auto lk gradient map generation, 69
acquisition status line, 40	auto.conf file, 202
Acquisition Status window, 43	auto_dir macro, 546
activating current window activity, 306	autocalibration, 39
active parameter, 377	getting with CH3I sample, 66
Active study name (P), 47	routines, 67, 68
activity in current window, 306	setting up with CH#I sample, 66
ADC overflow warning, 45	with autotest sample, getting, 66
Add required protocol before the main protocol, 644	with outstast sample satting up 66
add/subtract experiment, 554	with autotest sample, setting up, 66
	Autogain, see automatic gain
add current FID, 47	
	Autogain, see automatic gain

analysis of DEPT data, 70	Autoshim, see automatic shimming
carbon acquisition, 89	autoshimming, 563
carbon and APT acquisition, 91	gradient, 261
carbon and DEPT acquisition, 92	autotest
fluorine acquisition, 214	application mode, 60
phosphorus acquisition, 388	average value of input, 77
proton acquisition, 274	axis gradients, 102
proton and carbon acquisition, 276	axis labels, 78
proton and COSY acquisition, 278	FID displays and plots, 80
proton, carbon, APT acquisition, 277	units, 79
proton, carbon, DEPT acquisition, 277	diffes, 12
proton, carbon, HETCOR acquisition, 277	
	В
automated gradient map generation macros, 68	В
automatic	background execution, 537
2D normalization, 370	background VNMR processing, 605
2D peak picking, 318	backup current probe file, 39
2D processing, 372	bandpass filter offset for downsampling, 185
analysis of COSY data, 40	bandwidth for shaped pulse, 452
APT spectra processing, 60	bandwidth of audio filters, 216
calibration, 39	bandwidth of digital filter, 184
COSY- and NOESY-type spectra plot, 420	baseline correction, 82, 588
generic processing, 443	linear, 128
heteronuclear J-resolved 2D spectra plot, 422	sensitivity adjustment, 338
homonuclear J-resolved 2D spectra plot, 423	
integral scale adjustment, 300	zero-order, 338
macro execution, 528	baseline flatness, 287
plot APT-type spectra, 419	beeper sound, 83
process FIDs, 443	Bessel filters, 53
spectra plotting, 426	BINOM pulse sequence, 84
vertical scale adjustment, 614, 615	binomial water suppression, 84
X,H-correlation 2D spectrum plot, 424	blanked amplifiers, 493
	block size action, 621
automatic gain	block size storage, 86
enable Autogain, 247, 248	block size transients, 621
errors, 46	boxes
automatic lock	draw on plotter or display, 84
errors, 46	selected by mark command, 85
status, 53	BR24 pulse sequence, 86, 117
submit Autolock experiment to acquisition, 323	Brickwall digital filter, 136
automatic phasing, 58	Bruker data files, 559
optimized, 59	convert to VNMR, 104
zero-order term, 58	read files from 9-track tape, 467
automatic shimming, 635	Bruker data, phasing, 58
create shim method string, 366	bruker.par file, 559
method selection, 353	Butterworth filter, 53, 67
submit Autoshim experiment to acquisition,	button labels, 358
504, 538	button labels, 558
automatic stacking for arrays, 75	
automatic vertical scale adjustment, 614, 615	C
automation data file prefix, 74	C
automation directory	C13.par file, 534
absolute path, 71	calculated spectrum display, 187
check for enter queue, 71	calibration
preparation for run, 69	decoupler pulse, 457
automation directory name, 260	gradient strength, 444
automation mode, 71	shaped pulses, 81
	calibration file, printing, 70
check if active, 69, 208	carbon
automation parameter group, 148	acquisition, 276, 277
automation run	
controlling macro, 70	automated acquisition, 89
enter sample information, 202	plotting, 420, 430
prepare automation directory, 69	process 1D carbon spectra, 89
resume suspended run, 75	carbon decoupler calibration macros, 67
starting, 71	carbon gradient ratio calibration macros, 67
suspend current run, 75	carbon observe calibration macros, 67
autoscaling resumes, 75	carbon-enriched molecules, 277
Autoshim on Z button, 262	Carr-Purcell Meiboom-Gill T2, 108

cartridge tape, 581	complex Fourier transform, 441, 442
center frequencies of nD experiments, 477	complex points to left-shift ni interferogram, 336
center of screen display limits, 93	complex points to left-shift ni2 interferogram, 337
chained acquisition, 71	complex points to left-shift np FID, 336
chained experiments, 97	complex time-domain data points, 331, 332
chained study queue, 562	compressed 2D data conversion, 225
CHAN readout, 593	configuration information, 635
change sample experiment, 95	configuration parameters
changing working directory, 92 channels	display and possibly change, 99
	Configure label, 101
assign frequencies for probe tuning, 593 available for use, 375	conpar file, 100, 113 console hardware status, 293
rf frequencies, 548	Console label, 100
rf generation on each channel, 489	console parameter, 103
set frequency of rf channels, 518	console type, 103
waveform generator on channel, 489	contact time, 103
characters in a string, 316	continuous wave (CW) modulation, 160
chart	contour display
maximum width, 622	display control, 131
maximum width in second direction, 622	contour plot, 128, 404
starting position, 507	display, 168
starting position in second direction, 508	width of plotting area, 621
width, 621	without screen erase, 132
width in second direction, 621	contour plot display, 169
chart paper	conversion units for parameters, 598
preprinted paper for HP plotters, 287	convert a temporarily stored study into a submitted
chemical shift offset frequency, 526	study, 644
chemical shifts list, storing, 165, 405	converting
chemist-style parameters, 86	32-bit data files to VNMR, 559
class C amplifiers, 54, 91, 102	Bruker data, 104
decoupler high-power control, 149	compressed 2D data to standard format, 225
decoupler low-power control, 154	data in table order to linear order, 579
clearing	Hz or ppm to plotter units, 291
experiment text, 115	VXR-style data to VNMR, 108
integral reset points, 117	VXR-style text files to UNIX, 618
cmd parameter, 388	Copy protocols in a study queue, 644
Coarse Attenuator label, 91, 102	copying
coarse attenuator type, 91	experiment data to subfile, 109
coef file, 515, 548	files, 107, 108
coefficient to construct interferogram, 215, 216	local file to remote host, 204
coefficients for digital filtering, 182, 560	one parameter tree to another, 269
Cold Probes, 115	peak file to another file, 321
COLOC sequence, 424	remote file to local host, 202
color intensity map, 295	system macro to become user macro, 344
display, 128	user macro files, 342
without screen erase, 132	CORBA client, 115
color selection for drawing, 406	CORBA server, 115
colors for plotting, 98, 526	corrected difference between successive spectra, 219
combining arrayed 2D FID matrices, 236, 240	cosine value, 107
comm port for sample changer, 100	cosine-squared window function, 555
command execution, 204	cost accounting, 610
commands	COSY
dbsetup remove, 127	acquisition, 278
dbupdate, 127	automatic analysis and plot, 390
display which command or macro is used, 630	automatic analysis of data, 40
edit online description, 349	phase-sensitive, 107
online description, 348	plotting, 420
rename, 283	COSY-like correlation spectra, 228
comparing shim sets, 150	CPMGT2 pulse sequence, 108
compiling	create protocols macro, 557
user PSG object library, 448	Create study queue parameters for imaging, 557
user pulse sequences, 513	Create study queue parameters for liquids, 110
user-written weighting functions, 637	creating
compiling pulse sequence, 513	FID display parameters, 221
completed FIDs in experiment, 93	LC-NMR parameters, 396
completed transients, 115	parameters in a parameter tree, 112

UNIX directory, 357	plotting, 422
cross-polarization, 649	starting point, 601
cross-relaxation, 589	decoupler
CryoBay Monitor software, 115	adjust tip-angle resolution time, 157
cubic curve fitting, 56, 210	decoupling sequence, 183, 184
curecc file, 519	field strength, 155
curpar file, 113	field strength calculation, 275
current experiment	fine power attenuator, 176
correct parameter characteristics, 224	frequency, 142
determine if acquisition active, 208	frequency offset array, 510
current FID data block, 94	frequency offset control, 164
current gradient coil, 249 Current study queue node id, 46	high-power control, 149 linear modulator power, 177
current window, 116	low-power mode, 154
current working directory, 454	mode during status periods, 154, 155
current-type parameter tree, 113	modulation frequency, 101, 155, 156
cursor	modulation mode, 160, 161
adjust tau2 to start of acquisition, 592	nucleus lookup, 161
difference of two frequency cursors, 138, 139	power level with linear amplifier, 174
difference of two time-domain cursors, 139	power to switchable probe caution, 174, 175,
mode, 350	176
move cursor to center spectrum, 93	proton decoupler pulse calibration, 434
move cursor to nearest line, 368	pulse calibration, 457
move spectral window according to cursors,	pulse length, 433
359	pulse sequence diagram, 173
reset integral to zero at cursor, 653	set frequency to cursor position, 509
set decoupler frequency to cursor position, 509,	tip-angle resolution, 178
510	used as transmitter, 124
split difference of two cursors, 553	used for pulsing, 160
state in df, ds, or dconi programs, 114	WALTZ decoupling present, 620
cursor position, 111, 112	decoupler 2 parameter values
time domain, 113	set from probe file, 517
curve fitting, 55, 209	decoupler modulation frequency, 156
cutoff point for VT regulation, 617	decoupler parameter values
CW amplifier mode, 54	set from probe file, 517
cycle phase, 108	def file, 164
cycled BR24 pulse sequence, 117	Default button, 137
cycled MREV8 pulse sequence, 117	default directory for Files menu system, 136
CYCLENOE sequence, 117	defaultdomain file, 518, 525 defaultrouter file, 518, 525
	Define excitation band, 401, 512
D	Define excitation band, 401, 512  Define excitation band for solvent suppression, 401
_	delay
D2PUL pulse sequence, 124	first, 124
DAC, converting gauss/cm value to, 224	incremented delay for pulse sequence, 124, 125
data conversion to linear order, 579	overhead delay between FIDs, 123
data display mode, 158, 159	preacquisition, 53, 390
data file display in current experiment, 132	wait for another trigger, 197
data point, determining size of a, 58 data points to be acquired, 373	wait to acquire a spectrum, 197
• •	delay-type parameter, 112
data processing type on FID, 440, 441, 442 data set conversion from VXR-style to VNMR, 104,	deleting
108	experiments, 137
data station system configuration, 575	file, parameter, or FID directory, 137
data truncation limit, 116	files, 492
data.fdf file, 241, 242	spectra from analysis, 138
database for VnmrJ, 127	user macro, 137
datadir directory, 630	DEPT
date of data acquisition, 126	acquisition, 277
dbsetup remove command, 127	analysis and plot, 391
dbupdate command, 127	automated complete analysis, 70
dc correction, 218	automatic analysis and spectrum editing, 51
dc offsets removed from FIDs, 132	plotting data, 421
dconi.out file, 350	spectra array processing, 140
decay curves, 409	dept.out file, 51
deconvolution, 218, 223	DEPTGL pulse sequence, 139
display numerical results, 540	destroying

parameters, 140	start of plot, 547
parameters of a group, 140	user-defined weighting, 636
devicenames file, 527	width of plot, 631
devicetable file, 527	zero-order phase, 494
dg window, 582	directories
dg2 parameter, 147	change working directory, 92
Dgcstecosy pulse sequence, 147	create new UNIX directory, 357
Dgcstehmqc pulse sequence, 147	default for Files menu system, 136
DgcsteSL pulse sequence, 147	delete, 137
dglc parameter group, 148	display current working directory, 454
Dgroup of a parameter, 517	get information about files, 251
diagonal parameter arrays, 61	list files, 150
diagonal peaks threshold during peak picking, 643	list files in directory, 316, 335
dialog box, 103	move directory, 363, 479
dialog box from a macro, 149	path to current experiment, 116
dialog, starting a, 164, 165, 202	remote VXR-style system, 201
dialoglib directory, 164	remove empty directories, 492
difference between cursors in Hz, 138	remove from experiment, 97
difference between cursors in seconds, 139	rename directory, 363, 479
difference NOE experiment, 117	stored queue experiments, 71
diffusion analysis, 210	user's manual directory, 349
add to current display, 211	user's private VNMR files, 601
add to current plot, 407	VNMR system, 576
display, 210	disk file errors, 46
diffusion constant, 409	display
calculation, 409	acquisition information, 43
diffusion experiment analysis, 56	lock level, 42
diffusion gradient level, 250	spinner speed, 42
Diffusion Ordered Spectroscopy (DOSY), 166	temperature, 42
digital filter type, 136	Display integral values, 170
digital filtering	display limits
bandwidth, 184, 560	set for full screen, 244
bandwidth for oversampling, 379	set for full screen with room for traces, 244
coefficients for oversampling, 379	set for left half of screen, 315
create downsampling parameters, 395	set for right half of screen, 490
create parameters for oversampling, 398	display mode for plotter, 430
downsampling factor, 168	Display normalized integral values, 171
file of FIR digital filter coefficients, 222	display parameters
inline type, 188	create 3D display parameters, 393
number of coefficients, 182, 560	full recall of set, 231
parameter creation, 49	move between experiments, 352
digital lock display, 470	recall set, 464
digital resolution measurement, 178	save as a set, 503
digitally filtered FIDs, 150, 561	set to full spectrum, 214
dimension of experiment, 61, 89	display parameters group, 146
dimensionality of experiment, 251	display templates for 3rd rf channel parameters, 393
direct polarization, 649	display templates for pulse sequence, 533
directly detected dimension	displaying
absolute value display mode, 76	2D color map of 3D plane, 172
additive weighting constant, 78	2D data interactively, 129
data display mode, 158	2D spectra in whitewash mode, 181
first-order phase, 328	2D spectra in whitewash mode with no screen
Fourier number, 227	erase, 181
Gaussian function, 257	3D parameter group, 146
Gaussian shift constant, 257	3D plane projection, 173
line broadening, 313	3D planes, 189
phase angle display mode, 389	3rd/4th rf channel parameter group, 146
phased spectra display mode, 410	acquisition information, 43
power display mode, 455	acquisition parameter group, 145, 148
reference line frequency, 487	acquisition status information, 542
reference line position, 487	acquisition time, 212
scale spectral width, 508	add another diffusion analysis, 211, 407
set reference line, 491	arrayed acquisition parameters, 125
sinebell constant, 506	automation parameters, 148
sinebell shift constant, 506	color intensity map, 128
spectral width, 525, 574	contour plot, 128
, ,	1 7 -

contour plot with screen erase, 169	erase, 194
contour plots, 168	stacked spectra with no screen erase, 195
create 2D parameters, 393	strings in text window, 200
current working directory, 454	system macro file, 344
data file in current experiment, 132	temperature, 42
dialog box from a macro, 149	text file for current experiment, 585
display parameter group, 146	text file in graphics window, 196
error messages, 203	text files, 91
Ethernet address, 200	time of acquisition, 212
experiment library, 212	user macro files, 342
experiment time, 212	which command or macro is used, 630
exponential curves, 210	width adjustment, 658
FID, 141, 142	done codes, 44
FID as connected dots, 167	DOSY (Diffusion Ordered Spectroscopy)
FID file in current experiment, 132	experiment, 166
FIDs in whitewash mode, 145	dosyfit program, 166
FIDs of 2D experiment, 141	dosyfrq, 166
files in directory, 150	dosygamma, 167
formatted text, 632	double-precision data acquisition, 168
full window display, 639	double-precision VNMR FID data, 99
grid on 2D display, 268	double-quantum filtered COSY pulse sequence, 177
horizontal LC axis, 126	downsampling
inset spectrum, 298	bandpass filter offset, 185
integral amplitudes, 170	bandwidth of digital filtering, 184
integral with a spectrum, 180	creating parameters, 395
integrals at reset points, 152	digital filter coefficients, 182
LC-NMR parameters, 148	factor, 168
limNET nodes, 164	inline type, 188
line frequencies above threshold, 153	parameter creation, 49
lock level, 42	setting parameters, 358
log file for experiment, 212	dpiv, 170
message on acquisition status line, 40	dpivn, 171
next 3D plane, 367	dprofile, 172
normalized integral amplitudes, 170	DQCOSY, 177
normalized integrals, 153	DQCOSY experiment, 177
parameter value, 462	drawing a line between points, 177
parameters and their attributes, 151	drift correction
peak frequencies, 169	2D spectra traces, 128
phase file in current experiment, 133	activity flag, 128
plot is same as display, 639	calculation, 128
plotted contours, 168	cancel, 92
polynomial curves, 210	group, 128
previous 3D plane, 437	ds, 130
processing parameter group, 145, 148	ds.out file, 350
pulse sequence diagram, 173	DSP parameter creation, 49
pulse sequence generation errors, 448	DSP type (see digital filtering), 188
recalculated simulated spectrum, 187	DSP, type of, 189
remote VXR-style directory, 201	dummy scans, 559
scale under spectrum or FID, 182	Dynamic Angle Spinning (DAS), 126
set for center of screen, 93	dynamic binding, 514
set full screen with room for traces, 244	
set limits for full screen, 244	E
shim method string, 185	L
shim parameter group, 148	echo command (UNIX), 200
spectra in whitewash mode, 196	eddy current
spectrum, 180	testing, 268
spin simulation parameter arrays, 151	eddy current compensation
spin simulation parameter group, 147	off, 200
spinner speed, 42	on, 200
stacked FIDs, 143, 144, 145	editing
stacked spectra automatically 101	files, 200
stacked spectra automatically, 191 stacked spectra automatically with no screen	macros, 343
erase, 192	menu file, 353
stacked spectra horizontally, 192	parameter file with user-selected editor, 394
stacked spectra horizontally with no screen	parameter file with vi editor, 394
stacked spectra nortzontarry with no screen	UNIX text files, 606

user macro, 345	display phase file, 133
ejection of sample, 199, 201	edit text file, 586
elements, returning number of, 544	experiment library display, 212
elliptical filters, 53, 67	fit data to lineshapes, 223
Email address, 201	get text from data file, 255
End a study queue, 556	join existing experiment, 303, 304
enter.conf file, 202	make FID element using numeric text input,
enterexp file, 202	347
enumerated values, 112	move display parameters between experiments,
remove from a variable, 517	352
Ernst angle pulse calculation, 203	move FIDs between experiments, 354
errmsg file, 637	move parameters between experiments, 359
error codes, 44	nucleus selection, 534
error conditions recovery, 466	number of completed FIDs, 93
error during acquisition, 45, 623	parameters for basic experiment, 534
error handling control, 295	pulse sequence setup, 265
error message display, 203	recalculate number of transients, 587
errors in pulse sequence generation, 448	remove inactive lock and join experiment, 599
Ethernet	remove old files and directories, 97
address display, 200	replace text file, 585
disconnect host computer, 525	resume a stopped acquisition, 465
host computer connection, 518	retrieve FIDs from a file, 495
evolution dimension	retrieve parameters from file, 496
set spectral width, 532	retrieve shim coil settings, 496
evolution time increments, 367, 368	save FIDs, 569
excitation pulse, 387	save parameters, 572
Execute a protocol from the locator, 328	save text to a data file, 453
Execute processing macro, 205	select 1D experiment for processing, 250
executing VNMR command, 204	select 2D experiment for processing, 251
exiting from VNMR, 208, 611	set up $T_{I experiment}$ , 167
exp5 (add/subtract experiment), 48, 98 experiment data retrieval, 497	setup macros 225
experiment data retrieval, 497 experiment directory path, 116	setup macros, 225 shim values to use, 322
experiment numbers list, 349	shimming conditions, 635
experiment numbers list, 349 experiment parameters, restoring, 305	solvent selection, 534
experiment text file	stop acquisition, 504
append string, 64	string parameters for storage, 365
clear text, 115	submit Autolock experiment to acquisition, 32.
experiment time display, 212	submit Autoshim experiment to acquisition,
experimental frequency of transition, 98	538
experimental lines, assigning transitions, 62	submit change sample, Autoshim to acquisition.
experiments	504
abort acquisition with no error, 276	submit setup experiment to acquisition, 566
acquire and Fourier transform, 247	submit to acquisition, 65, 264
acquisition time estimate, 587	text file display, 585
action when bs transients accumulate, 621	expfit.out file, 407
action when error occurs, 623	exponential analysis, 578, 579
action when experiment completes, 625, 626	exponential curve, 209
action when nt transients accumulate, 630	exponential curve fitting, 56, 103, 210
add parameters, 49	exponential curves display, 210
add parameters for FID display, 221	exponential curves plot, 407
append string to text file, 64	exponential value of number, 208
calculate dimension, 89	exponential weighting, 313, 314
clear text file, 115	external time base, 643
completed transients, 115	extract entries in VXR-style directory, 136
correct parameter characteristics, 224	extrapolated dispersion mode, 219
correct parameter limits and step sizes, 396	
create workspace, 94	
delete an experiment, 137	F
determine if acquisition active, 208	F1 linear prediction parameters, setting, 525
dimension, 61	f <sub>1</sub> scaling factor for 2D multipulse sequences, 509
dimensionality, 251	$f_1$ , $f_2$ display, 374
display acquisition time, 212	f <sub>2</sub> ridges, 95
display data file, 132	f <sub>3</sub> processing of 3D data, 630
display FID file, 132	FDF files, 241, 242, 570
display log file, 212	FDM program, running, 217
	P B,

FID block, move, 354	TPPI 2D Fourier transformation, 239
FID block, reverse, 483	type of data processing, 440
FID button, 264	vertical position, 612
FID data, move, 355	vertical position of imaginary FID, 613
FID data, reverse, 486	vertical scale, 606
fid file, 580	weight and Fourier transform 1D FIDs, 627
FID file, memory map open, 356	weighting interactively, 637
FID trace, move, 356	width of FID display, 626
FID trace, reverse, 488	write numeric text file using a FID element, 634
FID, memory map close, 355	zero-order phasing constant, 413
fid.orig file, 580	field position, 654
FIDs	FIFO loop size, 221
absolute-value mode data display, 159	Fifo Loop Size label, 101, 221
acquisition time, 63 action after FID finishes, 631	FIFO underflow error, 46 files
action after last FID, 626	append data to file, 633
add to add/subtract experiment, 47	automation data file name prefix, 74
arrayed 2D FID matrices, 628, 629	Bruker data files for conversion, 559
automatic processing, 443	check for existence, 205
axis label units, 80	clear a file, 632
combine arrayed 2D FID matrices, 236, 240	delete, 137
complex points to left-shift np FID, 336	delete one or more files, 492
compress double-precision FID data, 99	display experiment library, 212
copy FIDs into exp5 as array, 261	display in text window, 91
create display parameters, 49, 221	edit with user-selectable editor, 200
current data block, 94	file name extension information, 252
cursor difference, 139	find number of files in directory, 251
delete FID directory, 137	find words and lines in text file, 326
digitally filtered FID, 561	get text from data file, 255
display as connected dots, 167	handle interactively, 221
display FID files in current experiment, 132	lines or records in file, 373
display FID of 2D experiment, 141	links to files, 316
display scale, 182	list files in directory, 150, 316, 335
display single FID, 141, 142	load parameters from file into a tree, 231
display stacked FIDs, 143, 144, 145	make a copy, 107
display whitewashed FIDs, 145	make FID files using numeric text input, 347
filtered, 560	making a copy, 108
first point multiplier, 230	move a file, 479
Fourier transform 1D FIDs, 233	move file, 363
Fourier transform 2D data, 236	plot files, 611
Fourier transform 3D FID into 3D data, 240 hypercomplex 2D Fourier transform, 239	plot text file, 430
imaginary part display, 142	print or plot to a file, 438 print text files, 451, 611
interactive display, 40	put text file into another file, 453
interleave FIDs during processing, 294	read 32-bit data files into VNMR, 559
left-shift FID to time-domain cursor, 588	read Bruker data files from tape, 467
make FID element using numeric text input,	remove old files from experiment, 97
347	rename a file, 479
maximum point in an FID, 220	rename file, 363
move FIDs between experiments, 354	retrieve FIDs from file, 495
noise level measurement, 371	retrieve parameters from file, 496
number of completed FIDs, 93	retrieve shim coil settings from file, 496
overhead delay between, 123	return information from files display, 222
plot a scale under a FID, 447	save FIDs in experiment, 569
plot in whitewash mode, 408	save parameters from experiment, 572
plot one or more FIDs, 421	save parameters from tree to a file, 232
prepare parameters for acqi display, 256	save shim coil settings to a file, 573
pulse breakthrough effects, 493	text files display in graphics window, 196
remove dc offsets, 132	transfer file from remote source, 202
retrieve from a file, 495	transfer files to remote destination, 204
retrieve from experiment subfile, 497	write formatted text to a file, 632
save FID data in FDF format, 570	Files menu system
save in current experiment, 569	default directory, 136
solvent subtraction, 395 start of FID display, 536	filter bandwidth, 216 filter delays, 53
subtract FID from add/subtract experiment, 567	filter diagonalization method (FDM), 217
saddaet 112 from add/saddaet experiment, 307	mar diagonanzadon mediod (1 Divi), 417

filtlib directory, 222	2nd evolution dimension, 531
Find an xml menu, 223	evolution dimension, 530
Find gzlvl1/gzwin button, 271	proton spectra, 134, 525, 529
Find gzwin button, 271	frequency referencing, see reference line
fine attenuator, 591	frequency scale dimension adjustment, 508
fine attenuator configuration, 216	frequency shift of fn spectrum, 337
fine attenuator control, 176	frequency shift of fn1 spectrum, 337
Fine Attenuator label, 102, 176, 216, 591	frequency shift of fn2 spectrum, 338
fine power attenuator, 176	frequency synthesizer latching, 313
finite impulse response (FIR) coefficients, 222	frequency synthesizer overrange, 380
first delay in pulse sequence, 124	frequency synthesizer value, 452
first pulse width 387	frequency-independent phase, 412
first pulse width, 387 first-order baseline correction, 588	frequency-independent term, 58
first-order phase, 328, 329	frequency-shifted quadrature detection, 232 frequency-type parameter, 112
make zero, 90	ftr3d call name, 307
first-order phase correction, 126	full display, 639
first-point multiplier, 95	full screen display limits, 244
fitspec.data file, 223	full-band amplifier, 54
fitspec.inpar file, 223, 519	full-width at half-height (FWHH), 318
fitspec.outpar file, 223, 519, 540	7,7
flag-type parameter, 112	
flashe command, 225	G
flip time, 387, 454	gain of receiver, 247
FLIPFLOP pulse sequence, 226	gap between lines in spectrum, 248
flow encoding gradient level, 257	GARP decoupling sequence, 155, 157, 160
fluorine	gating time for receiver, 115, 493, 494
automated acquisition, 214	gauss/cm, converting to DAC value, 224
process 1D spectra, 215	Gaussian apodization constant, 637
fm-fm mode decoupling, 155	Gaussian fraction, 187
fm-fm modulation (swept-square wave), 160	Gaussian fraction for lineshape, 519
folding-in problem, 216	Gaussian function, 257
foreground processing, 605	Gaussian function shift, 637
formatted text writing to a device, 632	Gaussian lineshape, 223
formatting real number as a string, 229	Gaussian shift constant, 257, 258
Fourier number, 227	Gaussian time constant, see Gaussian function
Fourier number scaled value of an integral, 297	Gaussian window function, 248
Fourier number scaled volume of a peak, 297 Fourier transform	gcoil parameter, 249
1D data, 233, 627	GCU (gradient compensation unit), 267
2D data, 234, 236, 627	Gemini systems
3D FID into 3D data, 240	convert data to VNMR, 104
phase-sensitive data, 235, 239	convert files to UNIX format, 618
processing mode for 2D data, 432	decompose files to UNIX files, 136
fourth decoupler	list contents of directory, 201
adjust tip angle resolution time, 158	read tape, 581
decoupler mode, 155	general setup for 2D experiments, 514
decoupling sequence, 184	generalized curve fitting to data, 209 generic automatic processing, 443
frequency, 143	Gilson Liquid Handler window, 260
frequency offset control, 165	gilson.conf file, 202
modulation frequency, 156	global file, 113, 114
modulation mode, 161	update after VNMR install, 600
nucleus lookup, 162	global parameter tree
power level with linear amplifier, 175	save parameters, 505
tip angle resolution, 179	global-type parameter tree, 113
fp.out file, 103, 138, 230, 578, 579	gmapz pulse sequence, 263
frequency limits of region, 254	gmapz.par file, 263
frequency of a line, 252	grad_cw_coef parameter, 409
frequency of decoupler, 142, 143	grad_p_coef parameter, 409
frequency of lock, 324	grad_pl array, 410
frequency of rf channels, 518	gradient
frequency offset far decoupler, 510	coil, 249
frequency offset for decoupler, 164, 165 frequency offset for observe transmitter, 590	gradient autoshimming, 261
Frequency Overrange label, 101, 380	gradient calibration constant, 249, 519
frequency referencing	gradient calibration pulse sequence, 444
maquement referenceing	gradient coil configuration, 575

gradient coil configuration file, 102	HETCORCP1 pulse sequence, 282
gradient coil updating, 600	HETCORPS pulse sequence, 282
gradient COSY pulse sequence, 250	hetero2d, 282
gradient HSQCTOXY, 259	heteronuclear 2D-J experiment, 282
gradient level trim, 270	heteronuclear chemical shift correlation, 282
gradient map generation, 68, 69	heteronuclear J-resolved 2D spectra, 422
gradient map generation, automatic, 68	heteronuclear multiple-quantum coherence, 284
gradient shape, 266	Hewlett-Packard plotters 287 642 651
gradient shimming display menu, 262	Hewlett-Packard plotters, 287, 642, 651 hiding a command, 283
map shims, 262	High Power Amplifier Enable, 283
pulsed field gradient strength, 270	high-power pulse widths, calibrating, 67
set parameters, 262	Hilbert transform algorithm, 219
spectral width percentage, 271	HMBC sequence, 424
start acquisition, 261	HMQC phase-sensitive PFG pulse sequence, 259
start gradient autoshimming, 261	HMQC pulse sequence, 424
z-axis shims number, 270	HMQCR pulse sequence, 284
Gradient Shimming System menu, 262	HMQC-TOCSY 3D pulse sequence, 284
gradient step size, 266	HOM2DJ pulse sequence, 284
gradient strength	homo2d, 285
maximum value, 264	HOMODEC experiment, changing parameters for,
gradient strength for each axis, 270	285
gradient strengths calibration for PGE, 408	homodecoupling control, 285
gradient total limit, 270	homonuclear correlation, 107, 589
gradients for X, Y, and Z axes, 267	homonuclear J-resolved 2D experiment, 284
Gradients label, 101	homonuclear J-resolved 2D spectra, 423
gradtables directory, 520	homorof1, 285
graphics window	homorof2, 286
display message with large characters, 81	homorof3, 286
display taxt file 106	homospoil, 287
display text file, 196 draw box, 84	pulse length, 289 pulses, 287
write formatted text to screen, 632	horizontal LC axis, 126, 399
Graphics Window colors, 516	horizontal offset, 284
graphics window, dividing into rows and columns,	horizontal projection of trace, 130
520	horizontally stacked spectra, 192
GraphOn terminal window clearing, 97	host computer
gray scale contrast adjustment, 268	serial port connection to changer, 545
gray scale display adjustment, 268	host computer connection to Ethernet, 518
gray scale image plot, 295	host computer disconnect from Ethernet, 525
grid lines over 2D plot, 422	host disk errors, 46
grid on a 2D display, 268	hostname.le0 file, 518
gripper abort, 45	hosts.3D file, 241
Group A parameters, 147	Hoult setting for final pulse times, 287
group of parameters in tree, 520	HSQC experiment, 288
	HSQC pulse sequence, 259
11	hsqcHT experiment, 288
Н	HSQC-TOCSY 3D pulse sequence, 289
H1.par file, 534	HSQCTOXY, gradient, 259
Hadamard	htbw1, 289
tocsy experiment, 590	htfrq1, 290
Hadamard transform processing, 441	hypercomplex points to left-shift interferogram, 336
Hadamard waveforms, 290	
hardware Ethernet address display, 200	1
hardware shimming	•
list of shims, 281	I1 and I2 values, 203
hardware shims, 467, 515	identifier, return for argument type, 595
hardware status of console, 293	idle mode for amplifiers, 54
hardware values in acquisition system, 521	IF Frequency label, 101
hardware Z1 shimming, 281	imaginary part of FID, 142
HCCHTOCSY sequence, 277	imaging
height of peak, 230	application mode, 60
HETCOP acquisition 277	attenuator, 92
HETCOR experiment changing parameters for 282	macros and menus, 60
HETCOR pulse sequence 282 424	Imaging Gradient Coil label, 102 imconi macro, 295
HETCOR pulse sequence, 282, 424	inicom macro, 473

inactive parameter, 377	J
INADEQUATE data about 2-quantum axis, 228	J-correlation experiment, 589
INADEQUATE pulse sequence, 296	Join a work space from the locator, 163
Incredible Natural Abundance Double-Quantum	joining
Transfer Experiment, 296	an existing experiment, 303, 304
incremented delay for pulse sequence, 124, 125	joint arrays, 61
index of experimental frequency of transition, 98	J-resolved 2D spectrum, 228
indirectly detected axis, 79	jump-and-return sequence, 305
INEPT pulse sequence, 296	JUMPRET sequence, 305
info directory, 515	,
info_# file, 409	
Initialize liquids study queue, 110	K
inline DSP, 188 Insensitive Nuclei Enhanced by Polarization	keyboard entries record, 473
Transfer, 296	keyboard focus to input window, 228
inserting a sample, 293, 298	keyboard input into variables, 296
inset spectrum, 298	kinetics analysis, 56, 210, 308, 309, 407
integer-type parameter, 113	kineties analysis, 30, 210, 300, 300, 107
integral	
display, 180	L
display mode, 299	
integral value, 297	label a stacked spectra display, 194
largest value in region, 298	lastlk file, 312
normalization scale, 297	latching capabilities of frequency synthesizer, 313 Latching label, 101, 313
offset, 299	LC axis, 126, 399
regions, 287	LC-NMR
reset points, 653	2D acquisition parameters, 314
scale, 299	create parameters, 396
set value, 523	create pseudo-2D dataset, 261
integral amplitudes display, 170	delay for trigger, 197
integral amplitudes plot, 416	display horizontal LC axis, 126
integral scale adjustment, 300	display LC-NMR parameters, 148
integrals	general 2D experiment setup, 315
clear reset points, 117	set up parameters for LC-NMR sequences, 315
data truncation limit, 116	set up pulse sequence for LC-NMR run, 314
display list 153	set up scout run, 531
display list, 152 find integral values, 368	TOCSY sequence, 315
reset point amplitudes, 316	least-squares curve fitting, 55, 209
reset point frequencies, 317	left half of screen display limits, 315
integration, 1D spectrum, 299	left-shift FID to time-domain cursor, 588
intensity of spectrum at a point, 349	left-shift ni interferogram, 336
intensity threshold, 563	left-shift ni2 interferogram, 337
interactive acquisition display, 40	left-shift np FID, 336
interactive phasing, 180	leg relay control, 316 Ifs (low-frequency suppression) option, 395
interactive UNIX shell, 538	limits for scales in regression, 508
interferogram coefficients, 215, 216	limits of parameter in a tree, 523
interferograms	limNET nodes database, 164
first-point multiplier, 231	line amplitudes list, 321
start of display, 536	line assignments for spin simulation, 96, 151
type of data processing, 441, 442	line broadening, 313, 314
weighting interactively, 637	line broadening factor, 638
width of display, 627	line drawing between points, 177
zero-order phasing constant, 414	line drawing capability, 358, 406
interlock to control lock level and spin speed, 295 Internet address, 42, 43, 44	line frequencies, 321, 544
inverse cosine calculation, 39	line frequencies and intensities
inverse Fourier transform, 233, 234	display list, 153
inverse sine, 62	find values, 369
inverse tangent, 63	line list plotting, 425
isreal, 300	line listing intensity and frequency, 252
isstring, 301	line narrowing sequence, 361
iterated parameters list, 302	linear amplifiers, 55, 102
iterations in an iterative simulation, 368	decoupler power level, 174, 175 power level, 590
•	linear curve fitting, 56
	linear fitting to data, 210

linear modulator power, 177, 591	lpanalyz.out.# file, 333, 334
linear monotonic order data, 579	
linear prediction	N.A.
algorithm, 329, 330	M
algorithm data extension, 332, 333	maclib directory, 84, 112, 137, 344, 473
arraying parameters, 332	Macro to create protocols, 110
calculation start point, 525, 564, 565	Macro to perform generic 2D plot, 110
coefficients to calculate, 331	Macro to save study queue parameters, 111
create parameters, 397 data extension, 330, 331	Macro to set weighting functions from a panel, 111
data extension, 550, 551 data extension start point, 564	macros
multiple operations, 332	activated by VNMR bootup, 84
number of data points, 331, 332	automatic execution, 528
output spectrum, 335	before experiment starts, 247
parameter creation, 49, 50	change action of abort command, 38
print output, 333, 334	check for existence, 205 copy system macro to become user macro, 344
printout, 333	copy user macro file, 342
type of data processing, 441, 442	create without text editor, 112
linearly spaced array values, 60	delete user macro, 137
line-narrowing multiple-pulse, 86	display dialog box, 149
lines of text, look up from a text file, 326	display system macro, 344
lineshape modification, 519	display user macro in text window, 342
linewidth for spin simulation, 544	display which macro is used, 630
linewidth measurement, 178	edit online description, 349
Load experiment from protocol, 556	edit user macro with vi editor, 345
Load experiment from protocol (M), 109	edit with macro editor, 343
load time counter, 643	hide command with same name, 283
local file transfer to remote host, 204 local host name display, 200	keyboard entries, 473
local oscillator (L.O.), 325	list system macros, 345
location of sample in tray, 323	list user macro file names, 343
location to start a line, 358	load macro into memory, 343
Locator action, 323	name of invoking macro, 342
lock	name storage for macros, 365
acquisition time constant, 323	online description, 348 real-value storage parameters, 465
automatic control, 53	remove macro from memory, 453
automatic phase adjustment, 323	remove system macro, 345
capture, 53	remove user macro from directory, 344
digital lock display, 470	restore normal abort function, 38
frequency, 524	return values to calling macro, 482
gain value, 325	terminate calling macro, 37
interactive, 40	terminate execution, 482
lock frequency adjustment, 324	magic angle spinning, see MAS
lock parameters setup, 524	magnet leg relay control, 316
loop time constant, 325	magnetization recovery, 124
phase value, 325	Make Shimmap button, 262
power value, 325 read current lock level, 470	makeuser command, 600
remove inactive lock, 599	manual directory, 349
solvent selection, 546	map shims, 262
solvent used, 312	MARK button, 350, 351
time constant, 325	mark output, 601
lock file, 605	mark1d.out file, 62, 350, 550, 601
lock frequency	mark2d.out file, 85, 350
track changes, 317	MAS cross-polarization spin-lock contact time, 103 MAS spinning speed, 558
Lock Frequency label, 101, 324	Max. Narrowband Width label, 168
lock level display, 42	Max. Spectral Width label, 101
lock level interlock, 295	Maximum DMF label, 101
log file, 212	maximum frequency of any transition, 544
log file for experiment, 212	Maximum gradient DAC value, 266
logarithm of a number, 321	maximum gradient strength for each axis, 270
login macro, 84	maximum of two spectra, 554
loop size of fifo, 221	maximum parameter value array, 397
Lorentzian lineshape, 223, 519	maximum transients accumulated, 363
low-band amplifier, 54	mean of the data in regression.inp file, 433
low-pass digital filter, 395	measured line frequencies, 544

measured line frequencies array, 550	type of data processing, 442
memory buffers, write to disk, 226	zero-order phasing constant, 414
memory increased by removing macros, 453	node files, 203, 204
memory map FID file, close, 355	nodes file, 203, 204
memory map open FID file, 356	NOE experiment, 117
memory usage statistics, 362	NOESY
Menu On button, 353	parameter set, 264
menulib directory, 222, 353	plotting spectra, 420
menus	NOESY 1D experiment, 371
button command string, 362	NOESY experiment, 371
change status of menu system, 353	noise level estimate, 254
edit menu with vi editor, 353	noise level in spectrum, 372
label for button, 358	noise level of FID, 371
menu displayed by Return button, 313	noise modulation, 160
return currently active menu, 366	noise multiplier, 372
select menu without activation, 366	normalized integral amplitudes, 170
message	normalized integral amplitudes plot, 416
confirm using mouse, 103	normalized integrals display list, 153
display with large characters, 81	normalized intensity mode, 370
messages from send2Vnmr, 317	nt array, 410
method string, 563	N-type display, 374
microimaging	nucleus for decoupler, 161, 162
gradient calibration constant, 519	nucleus for observe transmitter, 588
refocusing pulse shape, 455	nucleus selection, 534
shape of excitation pulse, 388	nucleus to add to probe file, 49
minimum frequency of any transition, 545	nuctables directory, 161
minimum intensity threshold, 563	number of increments of evolution time, 367, 368
minimum of two spectra, 554	Number of RF Channels label, 101, 375
minimum parameter value array, 398	Nyquist frequency, 294
MLEV-16 decoupling sequence, 155, 157, 160	
mode for n-dimensional data display, 591	
modulation frequency of decoupler, 155, 156	0
modulation mode for decoupler, 160, 161	object library for PSG, 448
mouse	observe nucleus transmitter frequency, 537
confirming a message, 103	offset
moving	horizontal, 284
files, 363, 479	integral, 299
parameters between experiments, 359	vertical, 612
spectral window according to cursors, 359	offset frequency
transmitter offset, 359	calculate for nucleus and ppm, 526
MQCOSY pulse sequence, 360	online description of command or macro, 348
MREV8 pulse sequence, 117, 361	edit description, 349
multidimensional data display mode, 591	open reel tape, 581
multihost processing, 242, 243	
multiple receivers	
	Operating system account owner, 382 Operator name, 378
add transformed data files with weighting, 493	Operating system account owner, 382
number of receivers, 374	Operating system account owner, 382 Operator name, 378
number of receivers, 374 set filter bandwidth, 361	Operating system account owner, 382 Operator name, 378 oph real-time variable, 108 order of parameter array, 61
number of receivers, 374 set filter bandwidth, 361 set gain, 362	Operating system account owner, 382 Operator name, 378 oph real-time variable, 108
number of receivers, 374 set filter bandwidth, 361 set gain, 362 weighting for different receivers, 466	Operating system account owner, 382 Operator name, 378 oph real-time variable, 108 order of parameter array, 61 out.c file, 106
number of receivers, 374 set filter bandwidth, 361 set gain, 362 weighting for different receivers, 466 multiple-pulse line narrowing, 86, 361	Operating system account owner, 382 Operator name, 378 oph real-time variable, 108 order of parameter array, 61 out.c file, 106 overhead delay between FIDs, 123
number of receivers, 374 set filter bandwidth, 361 set gain, 362 weighting for different receivers, 466 multiple-pulse line narrowing, 86, 361 multiple-quantum filtered COSY, 360, 589	Operating system account owner, 382 Operator name, 378 oph real-time variable, 108 order of parameter array, 61 out.c file, 106 overhead delay between FIDs, 123 overrange of frequency synthesizer, 380
number of receivers, 374 set filter bandwidth, 361 set gain, 362 weighting for different receivers, 466 multiple-pulse line narrowing, 86, 361 multiple-quantum filtered COSY, 360, 589 multipulse experiments	Operating system account owner, 382 Operator name, 378 oph real-time variable, 108 order of parameter array, 61 out.c file, 106 overhead delay between FIDs, 123 overrange of frequency synthesizer, 380 oversampling
number of receivers, 374 set filter bandwidth, 361 set gain, 362 weighting for different receivers, 466 multiple-pulse line narrowing, 86, 361 multiple-quantum filtered COSY, 360, 589 multipulse experiments f <sub>1</sub> scaling factor, 509	Operating system account owner, 382 Operator name, 378 oph real-time variable, 108 order of parameter array, 61 out.c file, 106 overhead delay between FIDs, 123 overrange of frequency synthesizer, 380 oversampling bandwidth, 379
number of receivers, 374 set filter bandwidth, 361 set gain, 362 weighting for different receivers, 466 multiple-pulse line narrowing, 86, 361 multiple-quantum filtered COSY, 360, 589 multipulse experiments	Operating system account owner, 382 Operator name, 378 oph real-time variable, 108 order of parameter array, 61 out.c file, 106 overhead delay between FIDs, 123 overrange of frequency synthesizer, 380 oversampling bandwidth, 379 factor for acquisition, 381
number of receivers, 374 set filter bandwidth, 361 set gain, 362 weighting for different receivers, 466 multiple-pulse line narrowing, 86, 361 multiple-quantum filtered COSY, 360, 589 multipulse experiments f <sub>1</sub> scaling factor, 509	Operating system account owner, 382 Operator name, 378 oph real-time variable, 108 order of parameter array, 61 out.c file, 106 overhead delay between FIDs, 123 overrange of frequency synthesizer, 380 oversampling bandwidth, 379 factor for acquisition, 381 filter type, 380
number of receivers, 374 set filter bandwidth, 361 set gain, 362 weighting for different receivers, 466 multiple-pulse line narrowing, 86, 361 multiple-quantum filtered COSY, 360, 589 multipulse experiments f <sub>1</sub> scaling factor, 509 scaling factor, 508	Operating system account owner, 382 Operator name, 378 oph real-time variable, 108 order of parameter array, 61 out.c file, 106 overhead delay between FIDs, 123 overrange of frequency synthesizer, 380 oversampling bandwidth, 379 factor for acquisition, 381 filter type, 380 number of coefficients, 379 parameter creation, 49, 398 setting parameters, 358
number of receivers, 374 set filter bandwidth, 361 set gain, 362 weighting for different receivers, 466 multiple-pulse line narrowing, 86, 361 multiple-quantum filtered COSY, 360, 589 multipulse experiments f <sub>1</sub> scaling factor, 509	Operating system account owner, 382 Operator name, 378 oph real-time variable, 108 order of parameter array, 61 out.c file, 106 overhead delay between FIDs, 123 overrange of frequency synthesizer, 380 oversampling bandwidth, 379 factor for acquisition, 381 filter type, 380 number of coefficients, 379 parameter creation, 49, 398
number of receivers, 374 set filter bandwidth, 361 set gain, 362 weighting for different receivers, 466 multiple-pulse line narrowing, 86, 361 multiple-quantum filtered COSY, 360, 589 multipulse experiments f <sub>1</sub> scaling factor, 509 scaling factor, 508  N name of pulse sequence, 513	Operating system account owner, 382 Operator name, 378 oph real-time variable, 108 order of parameter array, 61 out.c file, 106 overhead delay between FIDs, 123 overrange of frequency synthesizer, 380 oversampling bandwidth, 379 factor for acquisition, 381 filter type, 380 number of coefficients, 379 parameter creation, 49, 398 setting parameters, 358 Oxford shim supply, 539 Oxford VT controller, 351
number of receivers, 374 set filter bandwidth, 361 set gain, 362 weighting for different receivers, 466 multiple-pulse line narrowing, 86, 361 multiple-quantum filtered COSY, 360, 589 multipulse experiments f <sub>1</sub> scaling factor, 509 scaling factor, 508  N name of pulse sequence, 513 name storage for macros, 365	Operating system account owner, 382 Operator name, 378 oph real-time variable, 108 order of parameter array, 61 out.c file, 106 overhead delay between FIDs, 123 overrange of frequency synthesizer, 380 oversampling bandwidth, 379 factor for acquisition, 381 filter type, 380 number of coefficients, 379 parameter creation, 49, 398 setting parameters, 358 Oxford shim supply, 539
number of receivers, 374 set filter bandwidth, 361 set gain, 362 weighting for different receivers, 466 multiple-pulse line narrowing, 86, 361 multiple-quantum filtered COSY, 360, 589 multipulse experiments f <sub>1</sub> scaling factor, 509 scaling factor, 508  N name of pulse sequence, 513 name storage for macros, 365 natural logarithm of number, 321	Operating system account owner, 382 Operator name, 378 oph real-time variable, 108 order of parameter array, 61 out.c file, 106 overhead delay between FIDs, 123 overrange of frequency synthesizer, 380 oversampling bandwidth, 379 factor for acquisition, 381 filter type, 380 number of coefficients, 379 parameter creation, 49, 398 setting parameters, 358 Oxford shim supply, 539 Oxford VT controller, 351
number of receivers, 374 set filter bandwidth, 361 set gain, 362 weighting for different receivers, 466 multiple-pulse line narrowing, 86, 361 multiple-quantum filtered COSY, 360, 589 multipulse experiments f <sub>1</sub> scaling factor, 509 scaling factor, 508  N name of pulse sequence, 513 name storage for macros, 365 natural logarithm of number, 321 negative intensities, setting 2D, 658	Operating system account owner, 382 Operator name, 378 oph real-time variable, 108 order of parameter array, 61 out.c file, 106 overhead delay between FIDs, 123 overrange of frequency synthesizer, 380 oversampling bandwidth, 379 factor for acquisition, 381 filter type, 380 number of coefficients, 379 parameter creation, 49, 398 setting parameters, 358 Oxford shim supply, 539 Oxford VT controller, 351 Oxford-Sorenson VT controller, 351
number of receivers, 374 set filter bandwidth, 361 set gain, 362 weighting for different receivers, 466 multiple-pulse line narrowing, 86, 361 multiple-quantum filtered COSY, 360, 589 multipulse experiments f <sub>1</sub> scaling factor, 509 scaling factor, 508  N  name of pulse sequence, 513 name storage for macros, 365 natural logarithm of number, 321 negative intensities, setting 2D, 658 ni interferogram	Operating system account owner, 382 Operator name, 378 oph real-time variable, 108 order of parameter array, 61 out.c file, 106 overhead delay between FIDs, 123 overrange of frequency synthesizer, 380 oversampling bandwidth, 379 factor for acquisition, 381 filter type, 380 number of coefficients, 379 parameter creation, 49, 398 setting parameters, 358 Oxford shim supply, 539 Oxford VT controller, 351
number of receivers, 374 set filter bandwidth, 361 set gain, 362 weighting for different receivers, 466 multiple-pulse line narrowing, 86, 361 multiple-quantum filtered COSY, 360, 589 multipulse experiments f <sub>1</sub> scaling factor, 509 scaling factor, 508  N  name of pulse sequence, 513 name storage for macros, 365 natural logarithm of number, 321 negative intensities, setting 2D, 658 ni interferogram number of complex point to left shift, 336	Operating system account owner, 382 Operator name, 378 oph real-time variable, 108 order of parameter array, 61 out.c file, 106 overhead delay between FIDs, 123 overrange of frequency synthesizer, 380 oversampling bandwidth, 379 factor for acquisition, 381 filter type, 380 number of coefficients, 379 parameter creation, 49, 398 setting parameters, 358 Oxford shim supply, 539 Oxford VT controller, 351 Oxford-Sorenson VT controller, 351
number of receivers, 374 set filter bandwidth, 361 set gain, 362 weighting for different receivers, 466 multiple-pulse line narrowing, 86, 361 multiple-quantum filtered COSY, 360, 589 multipulse experiments f <sub>1</sub> scaling factor, 509 scaling factor, 508  N  name of pulse sequence, 513 name storage for macros, 365 natural logarithm of number, 321 negative intensities, setting 2D, 658 ni interferogram	Operating system account owner, 382 Operator name, 378 oph real-time variable, 108 order of parameter array, 61 out.c file, 106 overhead delay between FIDs, 123 overrange of frequency synthesizer, 380 oversampling bandwidth, 379 factor for acquisition, 381 filter type, 380 number of coefficients, 379 parameter creation, 49, 398 setting parameters, 358 Oxford shim supply, 539 Oxford VT controller, 351 Oxford-Sorenson VT controller, 351

parameter directory	display from tree with attributes, 151
delete, 137	display parameters group, 146
parameter list	display templates for third rf channel, 393
parameter names and values, 392	display values in text window, 200
plotting, 434	displaying value, 462
parameter panel	downsampling, 395
enable or disable, 644	edit parameter and its attributes, 394
parameter sets	full recall of display parameters, 231
correct saved parameter sets, 599	full spectrum display, 214
file name of retrieved set, 221	get value, 256
update all sets in directory, 599	gradient shimming, 262
parameter tree	limits of parameter in tree, 523
copy parameters of group, 269	linearly spaced steps, 61
create new parameter, 112	list to be iterated, 302
destroy parameters of a group, 140	lock parameters setup, 524
display parameters with attributes, 151	make parameter active, 377
limits of parameter, 523	maximum values, 397
load parameters from file into a tree, 231	minimum values, 398
make parameter active, 377 make parameter inactive, 377	move display parameters between experiments
prune extra parameters, 446	move display parameters between experiments, 352
remove a parameter, 140	plot list automatically, 57
set Dgroup of a parameter, 517	plot inst automatically, 57 plot on special chart paper, 287
set group of a parameter, 520	prepare for acqi, 256
set values of string parameter, 517	print all, 57
systemglobal-type tree, 100	protection mode, 527
types of trees, 113	prune parameters from tree, 446
value of parameter, 535	pseudo-echo weighting, 447
write parameters to file, 232	read from file and load into tree, 231
parameters	recall display parameter set, 464
3rd rf/3D parameter group, 146	reset after partial 3D FT, 481
4th rf channel parameter display group, 146	resolution enhancement, 481
acquisition/processing group, 145	restoring current experiment, 305
add for FID display, 221	retrieve from experiment subfile, 497
add parameter to probe file, 50	retrieve from file, 496
add to current experiment, 49	retrieve individual parameters from file, 497
adjust values from setup macros, 225	retrieve parameter from probe file, 253
adjusting, 305	save display parameters as set, 503
adjusting plot, 304	save from experiment, 572
arrayed for acquisition, 125	save from tree to file, 232
arraying order and precedence, 61	save parameters from global tree, 505
automation parameter group, 148	set group of parameter in tree, 520
basic experiment setup, 534	set up for pulse sequences, 448
boxed for plotting, 86	set up standard two-pulse sequence, 504
change type, 533	shims parameter group, 148
check existence, 205	sine window function, 542
chemist-style, 86	sinebell weighting, 543 sine-squared window function parameter
convert to PGE, 408	
copy between trees, 269	values, 543
correct limits and step sizes, 396	spin simulation parameter arrays, 151 spin simulation parameter group, 147
correct parameter characteristics, 224 create 2D parameters, 393	spin system parameters to iterate, 296
create 3D parameters, 393	step size values, 398
create 4D acquisition parameters, 393	system configuration, 99
create for fourth channel, 224, 225	test state of parameter, 377
create for linear prediction, 397	turn off active parameter, 377
create for third rf channel, 224	types of values, 112
create LC-NMR parameters, 396	unit conversion, 598
create new parameter in tree, 112	update after new VNMR install, 600
create oversampling parameters, 398	value of parameter in tree, 535
create parameters for 2D peak picking, 397	VAST experiment parameter setup, 604
create solvent subtractions parameters, 395	version of parameter set, 399
customize parameter sets, 601	parlib file, 448
destroy a parameter, 140	paths
destroy parameters of a group, 140	2D planes from a 3D data set, 399
display control, 57	current working directory, 454

VNMR system directory, 576	264
VNMR user directory, 601	absolute-value ROESY pulse sequence, 270
Pbox	amplifiers on/off control, 407
add parameter definition to pbox.inp file, 404	eddy current testing, 268
assign Pbox calibration data, 457	gradient calibration constant, 519
convert to Pbox default units, 404	HMQC phase-sensitive pulse sequence, 259
converts to default units, 404	HMQC pulse sequence, 258
create Pbox shape file, 109	HSQC pulse sequence, 259
create shape definition, 459	NOESY parameter set, 264
define excitation band, 401, 512	selective excitation pulse sequence, 512
define excitation band for solvent suppression,	sequence for PFG testing, 388
401	TNNOESY pulse sequence, 269
display interactive modulation pattern, 185	pge file, 408
display interactive pulse shape, 185	PGE pulse sequence
display last generated pulse shape, 184	calibrate gradient strengths, 408, 410
display modulation pattern, 184	extract data, 409
display pulse shape, 184	parameter conversion, 408
extract dmf value, 401	plot results, 409
extract dres value, 402	print results, 409
extract fine power level, 403	processing of data, 409
extract name of last shape, 402	phantom for gradient calibration, 519
extract power level, 402	phase angle display mode, 158, 389 phase correction applied to interferogram, 126
extract pulse length, 402	11 0 ,
generate a single-band shapefile, 458 open shape definition file, 378	phase cycling type, 412, 413 phase file
plot modulation pattern, 449	display in experiment, 133
plot modulation pattern, 449 plot pulse excitation profile, 435	phase of first pulse, 412
plot pulse shape, 449	phase of peaks, 230, 412
plot the last created pulse shape, 449	phase parameters
print pulse header, 435	automatic calculation of, 58
reset temporary pbox/Vnmr variables, 404	phase-correction angles, 328, 494, 495
simulate Bloch profile for a shaped pulse, 458	phased data display mode, 158
write a wave into file, 453	phased spectra display mode, 389, 410, 411
write wave definition string, 535	phase-sensitive 2D transformation, 328
pbox	phase-sensitive COSY pulse sequence, 107
write wave definition string, 535, 537	phase-sensitive data, 235, 239, 628, 629
pbox shape file, 109	phasing
pcss.outpar storage file, 165, 405	automatic, 58
peak frequencies display, 169	control update region, 413
peak frequencies plot, 434	phosphorus
peak frequencies threshold, 586	acquisition, 388
peak height or phase measurement, 230	processing, 388
peak heights comparison, 52	spectrum plotting, 430
peak noise, 254	pi/3 shifted sinebell squared window function, 414
peak number, 315	pi/4 shifted sinebell squared window function, 415
peak picking, 318	pl2dj macro, 426
diagonal peak threshold, 643	planes
parameters creation, 397	extract from 3D spectral data, 253
plot results, 425	Plot 2D arguments, 430
peak printout threshold, 586	Plot Design, joining, 303, 304
peak search range of data points, 373	plot parameters
peak truncation in spectra plot, 417	adjusting, 304
peak width of solvent resonances, 545	plot queue
peak, selecting, 568	show jobs in queue, 541
peaks.bin file, 321	stop jobs and remove from queue, 307
peak-to-peak noise, 372	plots, 409
maximum number to use 352	plotter
maximum number to use, 352 on HP plotter, 526	characteristics, 527 device setup, 428
* '	* /
selection for drawing, 406 Perform gradient shimming utility functions (M),	display mode, 430 Hewlett-Packard, 642
110	maximum number of pens, 352, 526
Performa I, II, III, 102	maximum width of plotting area, 622
Performa modules, 267	plot contours, 404
PFG	reinitializing, 307
absolute-value MQF COSY pulse sequence,	resolution of points drawn, 435

show plot queue, 541	polarization transfer experiments, 139
stopping plot jobs, 307	polynomial curve, 209
submit plot and change plotter page, 391	polynomial curves display, 210
write formatted text to plotter, 632	polynomial curves plot, 407
plotter units	polynomial fitting of baseline, 82
converted from Hz or ppm, 291	Postscript printer, 642
Plotters color, 516	powder pattern
plotting	finding the center, 553
2D contour plots for 3D planes, 429	power data display mode, 158
2D displayed resolution, 62	power level for decoupler with deuterium decoupler,
2D peak picking results, 425	175
2D spectra in whitewash mode, 418	power level for decoupler with linear amplifier, 174,
arrayed 1D spectra, 419	175
ATP-type spectra, 419	power level of transmitter, 590
axis label units, 79	power spectra display mode, 455, 456
boxed parameters, 86	power, setting, 527
carbon spectrum, 420	ppm calculations, 475
color assignments, 98, 526	ppm of solvent resonances, 545
contour plot with colors, 404	preacquisition delay, 390
contours display, 168, 169	precedence of parameter array, 61
COSY data set automatically, 390	presat 1H experiment, 437
COSY spectra, 420	Presat experiment processing, 647
deconvolution analysis, 422	print queue
DEPT analysis, 391	show jobs in queue, 541
DEPT data, 421	stop print jobs and remove from queue, 308
display same as plot, 639	printcap entry, 611
draw box, 84	printer
exponential curves, 407	device setup, 438
FIDs, 421	linewidth resolution, 435
FIDs in whitewash mode, 408	maximum width of chart, 622
files, 611	resolution in dots/mm, 435
formatted text, 632	send text to printer, 438
grid on 2D plot, 422	start print operation, 438
heteronuclear J-resolved 2D spectra, 422	stopping print jobs, 308
homonuclear J-resolved 2D spectra, 423	type, 611
horizontal LC axis, 399	write formatted text on printer, 633
limit to center of page, 93	printing
line list, 425	color assignments, 98, 526
NOESY spectra, 420	parameters, 57
non-arrayed 1D spectra, 426	PGE calculated results, 409
noninteractive gray scale image, 295	probe file after autocalibration, 39
parameter list, 392, 434	starting, 438
parameter list on special paper, 287	text file, 451
parameters automatically, 57	text files, 611
peak frequencies over spectrum, 434	printing area, see chart
PGE calculated results, 409	probe
phosphorus spectrum, 429	phase glitch removal, 226
plotter characteristics, 527	tuning, 363, 461, 581
polynomial curves, 407	tuning frequencies, 593
proton spectrum, 422	type, 439
pulse sequence, 436	probe directory, create new, 51
scale below spectrum or FID, 447	probe file, 39, 49, 526
set full page plot with room for traces, 244	add parameter, 50
set limits for full page plot, 244	retrieve parameter, 253
spectra, 417	set decoupler parameter values, 517
spectra automatically, 426	update, 600
spectra in whitewash mode, 432	probe file, copying, 70
spectral expansion, 52	probe file, create new, 51
start of plotting position, 507	probe file, make copy, 70
start of plotting position in second direction,	probe gcal calibration macros, 68
508	probe protection control, 440
text file, 430	probe, copying, 70
X,H-correlation 2D spectrum, 424	probe, editing, 440
plotting area, see chart	probeConnect, 439
plotting scaling factor, 291	procdat file, 515
point-by-point, 554	processed-type parameter tree, 113

processing	phase glitch removal, 226
1D carbon spectra, 89	pulse breakthrough effects, 493
2D spectra, 442	pulse interval time, 456
3D data processing information, 515	pulse length of decoupler, 433
arrayed 1D spectra, 443, 562	pulse power for shaped pulse, 452
create 2D parameters, 393	pulse sequence
create 3D processing parameters, 393	compiling, 513
DEPT spectra array, 140	display diagram, 173
FIDs automatically, 443	initiate compilation, 513
fluorine 1D, 215	label for screen, 449
generic automatic, 443	name to be used, 513
interleave FIDs, 294	phase-sensitive COSY, 107
phosphorus 1D spectra, 388	plotting a picture of a sequence, 436
proton 1D, 275	set up parameters, 448
select 1D experiment for processing, 250	setup macro, 265
selected 2D experiment, 251	Pulse Sequence Controller board, 494
simple 1D spectra, 441	pulse sequence generation, see PSG
solvent subtraction events, 395	pulse sequences
processing mode for 2D data, 432	display templates, 533
processing on FID, 440	pulse width in degrees, 387, 454
processing on the interferogram, 441, 442	pulse width length, 454
processing parameters group, 145	pulse width of first pulse, 387
procpar file, 99, 113, 599	pulse width optimum value, 203
procpar3d file, 515	pulsed field gradient strength, 270
procpar3d parameter set, 367, 515	pulse-type parameter, 112
programmable pulse modulation, 160	pulsewidth, setting, 527
project 2D data onto axis, 444	pure absorptive display, 239
projection plane, 173	pwwet pulse width, 625
protection mode of parameter, 527	pwx1 parameter, 457
proton	
acquisition, 276, 277	
automatic acquisition, 274	Q
spectra processing, 275	quadratic fitting to data, 56, 210
spectra vertical scale adjustment, 615	quadrature detection
spectrum plotting, 422, 430	frequency shifted, 232
proton acquisition, 277, 278	quadrupole echo pulse sequence, 559
proton chemical shifts spectrum	quadrupole echo puise sequence, 339 question mark (?) notation, 462
calculating, 165	question mark (1) notation, 402
calculating and showing, 405	
reducing to a list, 165	R
proton decoupler	N.
pulse calibration, 434	real Fourier transform, 441, 442
proton decoupler calibrations, 68	real number formatted into string, 229
proton frequency configuration, 275	real number, returning square root of a, 558
Proton Frequency label, 100, 275	real numbers, truncating, 592
proton gradient ratio calibration macros, 67	real variable
proton observe calibration macros, 69	create real variable without value, 471
proton parameter set, getting, 66	real-time digital filter, 380
Proton prescan, 287	real-time DSP (digital filtering), 188
protune macro, 445	real-type parameter, 112
ProTune shell script, 446	real-value storage for macros, 465
protunegui, 446	receiver
pseudo-2D, 604	channel imbalance, 108
pseudo-2D dataset, 261	gain, 247
pseudo-echo weighting, 447	gating time, 115, 493, 494
psg directory, 513	overflow warning, 45
PSG errors, 448	receiver option, 200-kHz, 168
PSG message, 37	recon_all, 471
PSG object library compilation, 448	recording current window activity, 306
psg.error file, 448	recording keyboard entries, 473
psglib directory, 513	Recover from acquisition error in study queue, 64
PTS frequency synthesizer, 101, 452	Recover from error conditions during automation
P-type diagonal, 228	study, 646
P-type double-quantum axis, 228	REDOR1 pulse sequence, 474
pulse amplifier	reference deconvolution, 218
mode, 54	reference frequency

position, 476, 477	rotating 2D data, 494
reference line	rotating frame NOE experiment, 589
clear referencing, 114	rotating frame Overhauser experiment, 493
frequency, 487, 488	rotating frame Overhauser experiment, 1D, 493
position, 487	rotational echo double-resonance, 474
reference frequency, 475	rotor speed display, 289
set line, 491, 492	rotor synchronization, 289
reference peak, see reference line	configuration parameter, 494
reference spectrum to TMS, 588	spinning rate, 558
refocusing pulse shape, 455	Rotor Synchronization label, 101, 289, 494
regions	RS-232 cable, 45
divide spectrum into regions, 478	Run an experiment to find the value of z0 (M), 110
find tallest peak, 405	Run prepare acquisition macro, 436
frequency limits of specified region, 254	running FDM program, 217
in spectrum, 375	running i Divi program, 217
plot expansions, 52	
selection, 287	S
region-selective 3D processing, 451	
regression analysis data input, 490	s2pul3rf parameter set, 393
regression mode, 55, 56	sample
regression mode curve fitting, 209	change for acquisition, 95
regression scale limits, 508	ejection from probe, 199, 201
	insert in probe, 293, 298
regression.inp file, 211, 433, 490	location of samples in tray, 323
RELAY-COSY pulse sequence, 478	spin rate, 549
RELAYH pulse sequence, 478	submit change sample experiment to
release procedure, 362	acquisition, 504
remote file transfer to local host, 202	temperature, 390, 582
remote machine name, 42, 43, 44	sample changer
removing	automation mode active, 69, 71
dc offsets, 132	automation run preparation, 69
directories, 492	change sample experiment, 95
files, 492	comm port, 100
user macro, 344	controlling, 70
renaming a command, 283	controlling macro for automation, 70
renaming files, 363, 479	errors, 45
reset points for integrals, 117	last lock solvent used, 312
Reset study queue parameters, 111	resume suspended automation run, 75
resetting acquisition computer, 38	serial port connection, 545
resolution enhancement function, 313, 314	starting automation run, 71
resolution enhancement parameters, 481	status window, 562
resolution equalization, 62	suspend automation run, 75
resolution on printers and plotters, 435	tray size, 592
restorenuctable, 482	Sample Changer label, 100, 592
Retrieve a parameter set from the locator, 163	Sample Changer Serial Port label, 545
Retrieve a shimset set from the locator, 163	sample information for automation run, 202
Retrieve and process fid data from the locato, 163	Sample Management System serial port connection,
retrieving	545
FIDs, 495	sample tray size, 100
parameters, 496	sample day size, 100 sampleinfo file, 70
Return button selection of menu, 313	save study parameters for imaging, 558
rf band in use, 483	saving
rf channel selection, 484	data, 569
rf channel type, 485	digitally filtered FIDs, 150
rf channels available, 375	display parameters, 503
rf channels frequencies, 518, 548	experiment data to subfile, 573
rf generation type, 489	,
rf pulse shape analysis, 452	FID data in FDF format, 570
rf waveform generator, 489	FIDs in current experiment, 569
ridges in FID display, 230	files using a base name, 505
right half of screen display limits, 490	parameters from current experiment, 572
right phase parameter, 495	parameters to file, 232
right phase-correction angles, 494	shim coil settings, 573
ROESY experiment, 493	text file into another file, 453
ROESY parameter set, 270	scale below spectrum or FID, 182, 447
Roesy1D, 493	scale limits in regression, 508
root-mean-square noise, 254, 372	scale spectral width, 508, 509
100t Incan square noise, 257, 572	scaling constant, 363

scaling factor for multipulse experiments, 508	shim method string creation, 366
scaling factor for plots, 291	shim method string display, 185
scaling factors, 78	shim parameters group, 148
scan in progress, 116	shim set type, 538
scout run, 531	shim supply, 539
screen display set for center, 93	shim values comparison, 150
SCSI errors, 46	shim values used, 322
second decoupler	shimmap calculations, 271
acquisition parameters, 146	shimmethods directory, 185, 353
adjust tip-angle resolution time, 157	shimming
decoupler mode, 154	automatic shimming conditions, 635
decoupling sequence, 183	Autoshim method, 353
fine power attenuator, 176	errors, 46
frequency, 142	interactive, 40
frequency offset array, 510	Z1 hardware, 281
frequency offset control, 164	shims
linear modulator power, 177	list of, for hardware shimming, 281
modulation frequency, 156	read all shims, 467
modulation mode, 160	set all shims, 515
	*
nucleus lookup, 162	Shimset label, 101, 538
power level with linear amplifier, 174	sidechain assignments, 277
pulse sequence diagram, 173	signal-to-noise ratio, 545, 583, 584
set frequency to cursor position, 509	estimate, 254
tip-angle resolution, 179	maximum, 186
second delay, 124	measurement, 186
selected widths, setting, 533	sine value of angle, 542
Selective 1D experiments Apptype macro, 511	sine window function values, 542
selective excitation experiment, continuing, 568	sinebell constant, 506
selective excitation pulse sequence, 512	sinebell shift, 638
selective frequencies, setting, 533	sinebell shift constant, 506, 507
selective inversion, setting up, 533	sinebell squared window function, 414, 415
selexHT, 512	sinebell time period, 638
send command to VNMR, 513	sinebell weighting parameters selection, 543
Seq label on screen, 449, 513	sinebell-squared window function, 558
seqfil file, 174	sine-squared window function, 543
seqgenmake file, 513	SIS (12 bit) gradients, 102
seqlib directory, 212, 513	skyline projection, 444
serial port connection, 545	software preparation date, 483
serverport, 514	software revision level, 483
Set colors for Graphics Window, 516	solids
set colors for Plotter, 516	adjust tau2 to current cursor position, 592
Set Default button, 137	cross polarization spin-lock experiments, 56
Set initial state for multiple viewports, 612	cross-polarization spin-lock analysis, 210
Set parameter bits for use with protocols, 518	echo pulse sequence, 559
Set Params button, 544	f <sub>1</sub> spectral width scaling factor, 509
sethtfrq1, 521	first pulse phase, 412
settune, 533	MREV8 multiple-pulse experiment, 361
setup experiment, 566	multiple-pulse line narrowing, 117
setup macros, 225	rotor speed display, 289
sf wf button, 536, 626	rotor synchronization module, 494
shape of an excitation pulse, 387	scaling factor for multipulse experiments, 508
shape of refocusing pulse, 455	solid-state echo pulse sequence, 559
shaped observe excitation sequence, 537	solid-state HETCOR sequence, 282
shaped pulse analysis, 452	spinning speed for MAS, 558
shaped pulse calibration, 81, 452	VT controller in use, 351
shapeinfo file, 81, 452	solvent resonances ppm and peak width, 545
shapelib directory, 81, 183, 184, 452, 457	solvent selection, 534
shared amplifier type, 54	solvent subtraction
shell on UNIX, 537, 538	create parameters, 395
Shifted Laminar Pulses (SLP), 314	filter bandwidth for filtered FID, 560
shifted sinebell squared window function, 414, 415	order of polynomial to fit digital filtered FID,
shim coil settings, 471	561
retrieve from file, 496	parameter creation, 50
save to file, 573	solvent suppression, 237
shim gradient, 642, 650, 651, 652, 654, 655, 656,	number of peaks for wet, 625
657. 658	solvent table information, 546

solvents file, 534, 546	proton spectrum plotting, 422
solvent-suppressed region, 560	reduce spectral width to minimum, 357
sparse FID data points, 167	reference line frequency, 487
Specify action when experiment is done	reference line position, 487
commnad, 622	reference to TMS, 588
parameter, 622	rotate 2D spectrum, 494
spectra	select spectrum without displaying, 511
3D dc correction, 548	signal-to-noise estimate, 254
absolute value display mode, 76	signal-to-noise test, 584
add spectrum to add/subtract experiment, 547	solvent-suppressed region, 560
APT plotting, 419	spectral integral display and plot, 299
automatic 1D integrate, 299	stacked spectra display, 189, 191, 192, 194,
automatic phase, 58	195
automatic phase adjustment, 58	subtract spectrum from add/subtract
center cursor, 93	experiment, 555
data truncation limit, 116	threshold for integrating 2D peaks, 586
deconvolution, 223	total width to be acquired, 574
delete spectra from $T_{1 \text{ or } T2 \text{ analysis}}$ , 138	update region during phasing, 413
display calculated spectrum, 187	vertical offset in stacked display, 612 vertical position, 612
display single spectrum, 180	1 ,
display single spectrum, 180	vertical scale, 613 whitewash mode display, 196
divide spectrum into regions, 478 drift correction calculation, 128	spectra, taking maximum of two, 554
drift correction carethation, 128	spectral expansion automatic plot, 52
extract planes from 3D spectral data, 253	spectral subtraction, 555
find gap in spectrum, 248	spectral width, 574, 575
find peak heights or phases, 230	percentage for gradient shimming, 271
find tallest peak in region, 405	reduce to minimum, 357
fold COSY-like correlation spectra, 228	set for given spectral window, 532
fold J-resolved 2D spectrum, 228	set in 2nd evolution dimension, 532
frequency shift of spectrum, 337, 338	set in evolution dimension, 532
frequency-independent phase, 412	spectrometer proton frequency, 275
full display, 214	spectrometer system configuration, 575
horizontal offset of each spectrum, 284	spectrum, plotting on side, 428
inset spectrum display, 298	spectrum, plotting on top, 428
integral amplitudes display, 170	spectrum, plotting on top and side, 429
integral amplitudes plot, 416	spin rate of sample, 549
integral regions, 287	spin setup experiment, 549
intensity of a spectrum at a point, 349	spin simulation
interactive display, 41	clear line assignments, 96
move cursor to center, 93	deconvolution start point, 601
move spectral window according to cursors,	display group of parameters, 147
359	index of experimental frequency, 98
noise limit, 372 normalized integral amplitudes display, 170	intensity threshold, 563
normalized integral amplitudes plot, 416	linewidth, 544 maximum frequency of any transition, 544
normalized intensity mode, 370 number of regions, 375	measured line frequencies, 544 measured line frequencies array, 550
offset of integral, 299	minimum frequency of any transition, 545
peak frequencies display, 169	number of iterations, 368
peak height comparison, 52	parameter arrays, 151
peak search range of data points, 373	parameters to be iterated, 302
phase adjustment, 58	perform spin simulation, 552
phase angle display mode, 389	set parameters to iterate, 296
phased display mode, 389, 410, 411	spin system entry, 554
phosphorus processing, 388	transition amplitude, 97
phosphorus spectrum plot, 429	transition frequency, 98
plot a scale under a spectrum, 447	transition number, 96
plot arrayed 1D, 419	using deconvolution as input, 223
plot COSY automatically, 420	vertical scale, 573
plot NOESY automatically, 420	spin speed interlock, 295
plot one or more spectra, 417	spin system
plot peak frequencies, 434	enter values for spin simulation, 554
plot spectra in whitewash mode, 432	restoring to before last run, 597
power spectra display mode, 455, 456	spini.inpar file, 597
processing simple 1D, 441	spini.la file, 151, 152

spini.savela file, 597	run prescans, 646
spin-lock contact time, 103	show data from node, 647
Spinner Control window, 551	submit sample(s) to, 648
spinner errors, 45	update the time, 648
spinner speed display, 42	write node attributes, 649
spinning speed for MAS, 558	write node order, 649
spins.list file, 552	Study queue file menu commands, 556
spins.outdata file, 187	Study queue mode, 556
Spinsight data, 496	Study queue prescan, 437
spinsys directory, 296	study queue, reset parameters for imaging, 557
spline fitting of baseline, 82	Study status, 566
split difference between two cursors, 553	Study time, 566
spoiler gradient level, 269	subfile, 109
square root, returning, 558	substring selected from a string, 567
square wave mode decoupling, 155, 160	subtracting zero-frequency components, 395
SSECHO pulse sequence, 559	sum of input, 78
SSECHO1 pulse sequence, 559	sum of squares of input, 78
stacked display width, 621	sum/difference spectrum, 48
stacked FIDs, 143, 144, 145	summing projection, 444
stacked plot of 2D spectra, 181, 418	sum-to-memory error, 46
stacked spectra display, 189, 191, 192, 194, 195	Sun display, clear window, 97
stacked spectra horizontal offset, 284	swept-tune graphical tool, 363, 461, 581
stacking control, 562	switchable probe caution, 149, 174, 175
stacking mode, 561	synchronous decoupler mode, 154
standard 2-pulse sequence, 387, 504	Synthesizer label, 101, 452
standard application mode, 60	synthesizer value, 452
standard deviation of input, 77	syshelppath global parameter, 60
start of FID display, 536	sysmaclibpath global parameter, 60
start of interferogram, 536	sysmenulipath global parameter, 60
start of plot, 547	system configuration parameters, 99
Start the night queue, 648	system console type, 103
starting	system macros
VNMR directly, 608	copy system macro to become user macro, 344
starting Plot Designer, 303	display in text window, 344
static binding, 514	list system macro names, 345
stdpar directory, 534	online description, 348
steady state pulses, 559	remove system macro, 345
Step Size label, 102, 590	system type configuration, 575
step size parameter value array, 398	System Type label, 100, 103, 575
stopping acquisition, 504	systemglobal directory, 113
stored data, style, 104	systemglobal parameter tree, 100
stored queue, 71	systemglobal-type parameter tree, 113
streaming tape, 581	
strength of pulsed field gradient, 270	_
string	Τ
format for output, 229	$T_1$ analysis, 210
length in characters, 316	delete spectra, 138
select substring, 567	plot curves, 407
text window display, 200	set up parameters, 167
string parameter values, 517	$T_1$ analysis, $\frac{56}{}$
string variable creation, 563	$T_I$ analysis, 578
string-type parameter, 112	t <sub>1</sub> dimension, 336
Study identification, 565	$T_2$ analysis, 210
Study parameters, 565	delete spectra, 138
study queue	plot curves, 407
delete nodes, 644	$T_2$ analysis, $\frac{56}{}$
experiment manager utility macro, 643	$T_2$ analysis, $\frac{579}{7}$
find next prescan or next experiment, 645	t <sub>2</sub> dimension in, 337
get attributes, 645	tablib directory, 579
initialize imaging study, 645	tangent value of angle, 580
make new node, 645	tapes
move node up and lock it, 645	device selection, 581
perform action, 643	display contents, 580
perform action for walkup, 643	rewind tape, 581
read node attributes, 646	taps in digital filter, 560
retrieve parameters, 646	tau2 adjustment, 592

Tcl script, 582	TNROESY sequence, 589
tdelta parameter, 250	TNTOCSY sequence, 589
tdiff parameter, 250	TOCSY experiment, 589
temperature calculation curve, 583	tocsy experiment, Hadamard encolded, 590
Temperature Control window, 582	TOCSY pulse sequence, 236, 240, 628, 629
temperature display, 42	tof parameter, 317
temperature interlock, 587	total correlation (TOCSY) experiment, 589
temperature of sample, 582	traces
temperature regulation, 618	find maximum intensity, 406
terminating	select trace without displaying, 511
abort function in macro, 38	transients completed, 115
calling macro, 37	transients setpoint action, 630, 631
Test acquire mode, 583	transients to be acquired, 373
testing signal-to-noise of spectrum, 584	transition amplitude, 97
text file	transition calculation, 62
display for current experiment, 585	transition frequency, 98, 544
edit file, 586	transition number calculation, 96
edit with vi text editor, 606	· · · · · · · · · · · · · · · · · · ·
	transitions frequency, 545
editor, 200, 606	transmitter
graphics window display, 196	fine power, 591
plotting, 430	frequency of observe nucleus, 537
print text files, 451, 611	frequency offset for observe transmitter, 263,
put into another file, 453	590
search for words and lines, 326	linear modulator power, 591
write file using a FID element, 634	local oscillator (L.O.) gate, 325
text output sent to printer, 438	move transmitter offset, 359
text window	nucleus of observe transmitter, 588
display files, 91	positioning, 590
display status, 585	power level with linear amps, 590
display strings and parameter values, 200	pulse sequence diagram, 173
display user macro, 342	transmitter frequency, 142, 143
list files, 150	transmitter frequency, 142
tflow parameter, 257	tray size on sample changer, 592
third decoupler	trigger signals to wait before acquisition, 374
adjust tip-angle resolution time, 157	trim gradient level, 270
decoupler mode, 155	triple-quantum filtered 2D MAS experiment, 126
decoupling sequence, 183	TROESY pulse sequence, 592
fine power attenuator, 176	truncating real numbers, 592
frequency, 142	truncation limit, 116
frequency offset array, 510	TUNE INTERFACE unit, 593
frequency offset control, 165	tunemethod, 594
linear modulator power, 177	tuning the probe, 593
modulation frequency, 156	Type of Amplifier label, 54, 102
modulation mode, 161	type of Ampinier label, 54, 102
nucleus lookup, 162	
	Type of RF label, 101, 485, 489
power level with linear amplifier, 175	
set frequency to cursor position, 510	U
tip-angle resolution, 179	U
threshold for integrating peaks in 2D spectra, 586	U+ H1 Only decouplers, 485
threshold for peak printout, 586	Ultra 8 shim configuration, 597
threshold for printout of peak frequencies, 586	Ultra•nmr shim system, 471, 539
time constant for lock, 325	ultra18, 597
time constant for lock acquisition, 323	ultra8, 597
time counter, 643	unit conversion for parameters, 598
time-domain cursor position, 113, 592	UNIX shell startup, 537, 538
time-domain cursors, 139, 588	unlocked experiment, 599
time-domain dc correction, 218	unshifted cosine-squared window function, 555
time-domain solvent subtraction, 395	unshifted Gaussian window function, 248
tip-angle resolution for decoupler, 178, 179	unshifted sinebell-squared window function, 558
tip-angle resolution time for decoupler, 157, 158	update VnmrJ database, 127
TMS reference, 588	updating
TNCOSYPS sequence, 589	gradient coil, 600
TNDQCOSY sequence, 589	
TNMQCOSY sequence, 589	updating revision global file and parameters, 600
TNNOESY parameter set, 269	Upper Limit label, 102, 590
TNNOESY sequence, 589	Use Console Data button, 100
* *	user macros

copy file, 342	VT Controller label, 101, 617
copy system macro to become user macro, 344	VT controller type, 617
create without text editor, 112	VT cutoff point, 617
delete, 137	VT errors, 45
display in text window, 342	VT FAILURE message, 618
edit with vi text editor, 345	VT regulation light, 587
library, 112	VT system in use, 351
list user macro file names, 343	VT wait time, 618
remove user macro from directory, 344	VXR-style directory
user-defined weighting, 636	decompose to UNIX files, 136
users currently on the system, 620	VXR-style systems
user-selectable editor, 343	convert data to VNMR, 104, 108
user-supplied modulation, 160	decompose files to UNIX files, 136
user-written weighting functions, 637	read tapes, 580
usrwt.o file, 637	remote directory display, 201
	VXR-style text files
	conversion to UNIX format, 618
V	conversion to CTAIL format, CTC
value of parameter in a tree, 256	
variable temperature, see VT	W
	w.command (LINIX) 620
Varian shim supply, 539 VAST accessory, 604	w command (UNIX), 620
	walkup automation, 620
VAST data analysis, 98	WALTZ decoupling present, 620
VAST experiments, setting up initial parameters for,	WALTZ decoupling sequence, 155
604	WALTZ-16 modulation, 160
VAST microtiter plate, 98	warning error codes, 45
version of parameter set, 399	water suppression, 84, 305, 589
vertical offset, 612	waveform generator, 489, 537
vertical position, 612	decoupling, 178
FID, 612	pulse interval time, 456
imaginary FID, 613	test, 627
vertical projection of trace, 130	waveform generator decoupling, 179
vertical scale adjustment, 614, 615	Waveform Generator label, 102, 489
vertical scale adjustment, automatic, 614	weight and Fourier transform
vertical scale for 2D displays, 614	1D data, 627
vertical scale for projections and traces, 616	2D data, 628
vertical scale for simulated spectrum, 573	along f <sub>2 for 2D data</sub> , 627
vertical scale for spectrum, 613	phase-sensitive data, 628, 629
vertical scale of FID, 606	weight.h file, 637
vi command (UNIX), 200, 606	weighting
vi text editor, 345, 353, 606, 608	constant, 78
Viewport layout, 306	interactive weighting, 637
viewport, set state to three default viewports, 613	interactive weighting for 2D absorptive data.
viewports, set new, 613	638
VNMR	user defined, 636
background processing, 605	weighting function compilation, 637
exiting, 208	wet 1H experiment, 624
exiting from system, 611	WETDQCOSY pulse sequence, 624
start VNMR application directly, 608	WETGCOSY pulse sequence, 624
style of stored data, 108	WETGHMQCPS pulse sequence, 624
system directory, 576	WETGHSQC pulse sequence, 624
updating parameters and global file after install,	WETNOESY pulse sequence, 625
600	WETTWEST pulse sequence, 625 WETPWXCAL pulse sequence, 625
user directory, 601	WETTNTOCSY pulse sequence, 625
	·
vnmr_textedit file, 394	WFG (waveform generator), 267
vnmr_vi file, 394	WFG + GCU gradients, 102
vnmraddr parameter, 231	what you see is what you get, 639
vnmreditor variable, 200, 394	whitewash mode, 408, 418, 432
VnmrJ, 609	whitewash mode display, 145, 181, 196
software preparation date, 483	who is using system, 620
software revision level, 483	wideline systems data precision, 168
VnmrJ database, 127	width of chart, 621
VnmrJ database, updating, 127	maximum width, 622
VnmrJ network listening port, 514	maximum width in second direction, 622
vnmrsystem variable, 576	second direction, 621
vnmruser variable, 601	width of FID, 626

width of interferogram, 627	Z3C shim gradient, 655
width of plot, 631, 632	Z3X shim gradient, 655
wildcard characters, 492	Z3X3 shim gradient, 656
window	Z3XY shim gradient, 656
current, 116	Z3Y shim gradient, 656
window activity, 306	Z3Y3 shim gradient, 656
windows	Z4 shim gradient, 656
clearing a window, 97	Z4C shim gradient, 656
word lookup in text file, 326	Z4X shim gradient, 656
Work space	Z4X2Y2 shim gradient, 657
labels for all viewport buttons, 305	Z4XY shim gradient, 657
numbers of all viewports, 303	Z4Y shim gradient, 657
numbers of the current viewports, 305	Z5 shim gradient, 657
1 /	
workspace for VNMR experiment, 94	Z5X shim gradient, 657
write out memory buffers, 226	Z5Y shim gradient, 657
Write sample enterQ entry for study queue	Z6 shim gradient, 657
automation, 646	Z7 shim gradient, 657
imaging, 647	Z8 shim gradient, 658
wtlib directory, 637	z-axis shims used for gradient shimming, 270
wtune, 638	zero-filling, 227, 482
	zeroing phase, 115
34	zero-order baseline correction, 338
X	zero-order phase, 494, 495
X Axis, Y Axis, Z Axis label, 102, 267	zero-order phasing constant, 413, 414, 495
X,H-correlation 2D spectrum, 424	zero-order term automatic phase, 58
X1 shim gradient, 642	zfs (zero-frequency suppression) option, 395
	ZX2Y2 shim gradient, 658
X2Y2 shim gradient, 642	ZX3 shim gradient, 658
X3 shim gradient, 642	ZXY shim gradient, 658
X4 shim gradient, 642	ZY3 shim gradient, 658
XL systems	<i>g</i> ,
convert data to VNMR, 104	
convert files to UNIX format, 618	
decompose files to UNIX files, 136	
list contents of directory, 201	
read tape, 581	
xmtune, 648	
XPOLAR1 pulse sequence, 649	
XY shim gradient, 650	
XY32 decoupling sequence, 155, 160	
XZ shim gradient, 650	
XZ2 shim gradient, 650	
x-zero position of Hewlett-Packard plotter, 642	
Υ	
V1 shim gradient 651	
Y1 shim gradient, 651	
Y3 shim gradient, 651	
Y4 shim gradient, 651	
YZ shim gradient, 651	
YZ2 shim gradient, 652	
y-zero position of Hewlett-Packard plotter, 651	
Z	
z0 calibration, automatic, 66, 68	
Z0 field position, 654	
Z0, automatic adjustment, 68	
Z1 shim gradient, 654	
Z1C shim gradient, 654	
Z2 shim gradient, 654	
Z2C shim gradient, 655	
Z2X2Y2 shim gradient, 655, 656	
Z2X3 shim gradient, 655	
Z2XY shim gradient, 655	
Z3 shim gradient, 655	