

MAY 20 1998

SAND--98-0907

# SANDIA REPORT

SAND98-0907 • UC-705

Unlimited Release

Printed May 1998

Supersedes SAND96-2836,  
dated December 1996

## xdamp Version 3: An IDL<sup>®</sup>-based Data and Image Manipulation Program

RECEIVED

JUN 02 1998

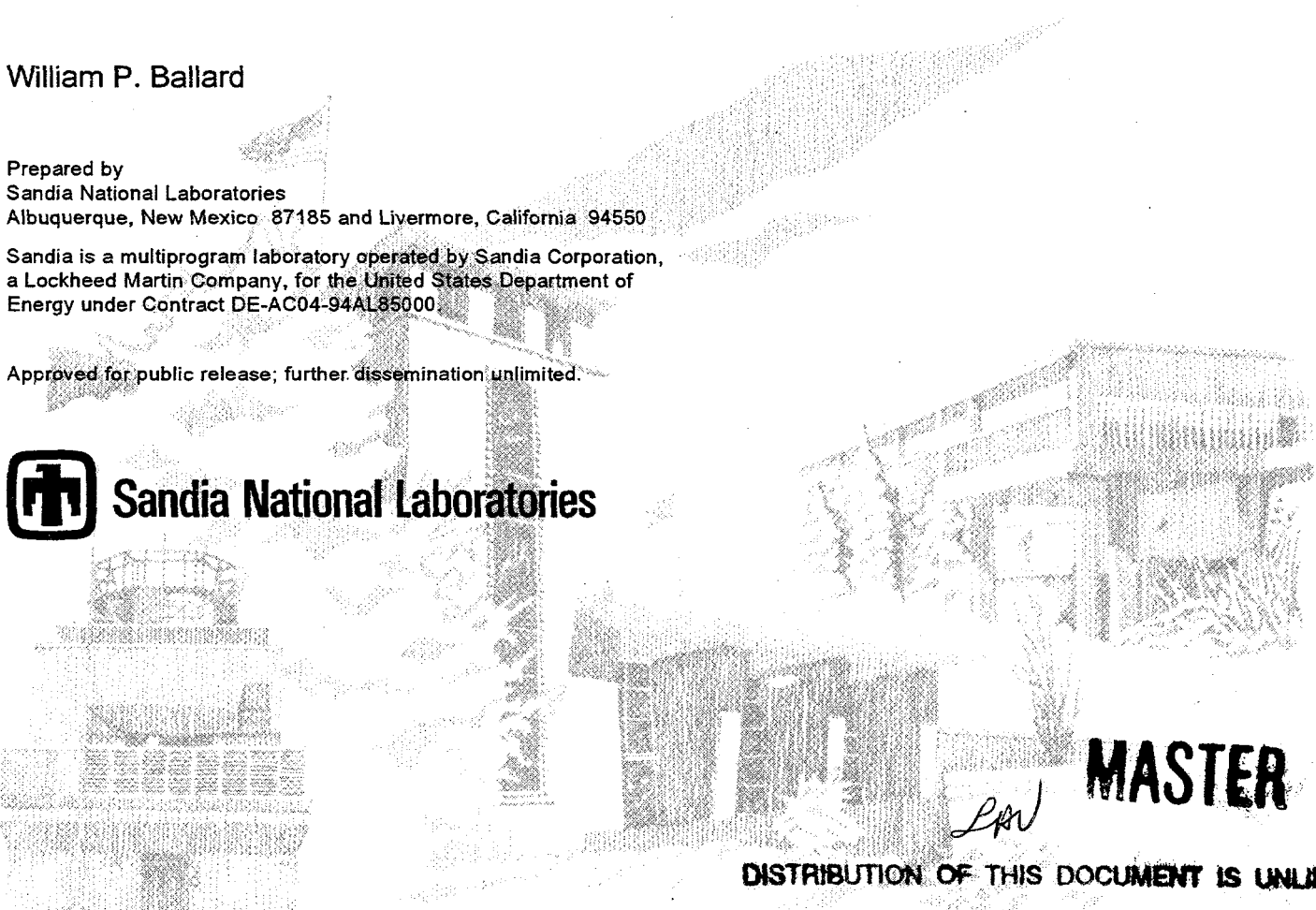
OSTI

William P. Ballard

Prepared by  
Sandia National Laboratories  
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation,  
a Lockheed Martin Company, for the United States Department of  
Energy under Contract DE-AC04-94AL85000

Approved for public release; further dissemination unlimited.



MASTER

*LPW*

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

**NOTICE:** This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from  
Office of Scientific and Technical Information  
P.O. Box 62  
Oak Ridge, TN 37831

Prices available from (615) 576-8401, FTS 626-8401

Available to the public from  
National Technical Information Service  
U.S. Department of Commerce  
5285 Port Royal Rd  
Springfield, VA 22161

NTIS price codes  
Printed copy: A03  
Microfiche copy: A01



## **DISCLAIMER**

**Portions of this document may be illegible electronic image products. Images are produced from the best available original document.**

SAND98-0907  
Unlimited Release  
Printed May 1998

Distribution  
Category UC-705

Supersedes SAND96-2836  
Printed December 1996

# *xdamp* Version 3: An IDL<sup>®</sup>-based Data and Image Manipulation Program

**William P. Ballard**

Simulation Technology Research Department  
Sandia National Laboratories  
P.O. Box 5800  
Albuquerque NM 87185-1179

## **Abstract**

The original DAMP (Data Manipulation Program) was written by Mark Hedemann of Sandia National Laboratories and used the CA-DISSPLA<sup>™</sup> (available from Computer Associates International, Inc., Garden City, NY) graphics package as its engine. It was used to plot, modify, and otherwise manipulate the one-dimensional data waveforms (data vs. time) from a wide variety of accelerators. With the waning of CA-DISSPLA and the increasing popularity of Unix<sup>®</sup>-based workstations, a replacement was needed. This package uses the IDL<sup>®</sup> software, available from Research Systems Incorporated in Boulder, Colorado, as the engine, and creates a set of widgets to manipulate the data in a manner similar to the original DAMP and earlier versions of *xdamp*. IDL is currently supported on a wide variety of Unix platforms such as IBM<sup>®</sup> workstations, Hewlett Packard workstations, SUN<sup>®</sup> workstations, Microsoft<sup>®</sup> Windows<sup>™</sup> computers, Macintosh<sup>®</sup> computers and Digital Equipment Corporation VMS<sup>®</sup> and Alpha<sup>®</sup> systems. Thus, *xdamp* is portable across many platforms. We have verified operation, albeit with some minor IDL bugs, on personal computers using Windows 95 and Windows NT; IBM Unix platforms; DEC Alpha and VMS systems; HP 9000/700 series workstations; and Macintosh computers, both regular and PowerPC<sup>™</sup> versions. Version 3 adds the capability to manipulate images to the original *xdamp* capabilities.

THIS PAGE INTENTIONALLY LEFT BLANK

---

*Table of Contents*


---

List of Figures .....	9
List of Tables.....	9
Introduction.....	11
PHILOSOPHY.....	11
INSTALLATION .....	12
RESTRICTIONS .....	13
GETTING STARTED.....	13
INITIALIZATION FILE.....	13
FINISHING UP .....	15
VERSION 3 CHANGES .....	15
Screen Layout .....	17
File Menu .....	19
FILE.OPEN.....	19
FILE.CLOSE .....	20
FILE.APPEND.....	20
FILE.GET DATA .....	20
FILE.IMPORT IMAGE.GIF.....	21
FILE.IMPORT IMAGE.HDF .....	21
FILE.IMPORT IMAGE.JPG.....	21
FILE.IMPORT IMAGE.PERKIN ELMER .....	21
FILE.IMPORT IMAGE.PICT.....	21
FILE.IMPORT IMAGE.TECPLOT .....	22
FILE.IMPORT IMAGE.TIFF .....	22
FILE.EXPORT IMAGE.EPS .....	22
FILE.EXPORT IMAGE.GIF .....	22
FILE.EXPORT IMAGE.JPEG.....	22
FILE.EXPORT IMAGE.PICT .....	23
FILE.EXPORT IMAGE.TIFF.....	23
FILE.SAVE .....	23
FILE.SAVE AS .....	23
FILE.SAVE ASCII.....	23
FILE.SET FILTER.....	24
FILE.SET DATA TYPE.....	24
FILE.EXIT .....	24
Edit Menu.....	25
EDIT.STORE .....	25
EDIT.NEW .....	25
EDIT.COPY .....	25
EDIT.SORT.....	25
EDIT.RENAME .....	26

EDIT.SELECT .....	26
EDIT.DELETE.....	26
EDIT.KEEP .....	26
EDIT.RESTORE .....	26
EDIT.NOTES .....	26
EDIT.FILE NOTES .....	27
EDIT.TITLE.....	27
Print Menu .....	29
PRINT.SCREEN .....	29
PRINT.ALL.....	29
PRINT.SELECTED .....	29
PRINT.SELECTED .....	29
PRINT.SUMMARY .....	29
PRINT.HEADER .....	29
PRINT.NOTES.....	30
PRINT.AUDIT TRAIL .....	30
PRINT.FILE NOTES .....	30
PRINT.PRINTER TYPE.....	30
PRINT.PRINTER SETUP.....	30
PRINT.QUEUE OUTPUT .....	30
PRINT.QUEUE NAME .....	30
PRINT.PRINTER RESET.....	31
Options Window .....	33
CURSOR.....	33
AUTO PLOT .....	34
Clear Stack After Operations .....	34
MAKE INITIALIZATION FILE .....	34
PS .....	35
CPS.....	35
QUEUE NAME.....	35
GRAPHIC AREA SIZE.....	35
PLOT #/PAGE.....	35
SHOT DATE .....	35
TODAYS DATE .....	35
PRETTY.....	35
SYMBOLS.....	35
MONOCHROME.....	36
COLOR .....	36
LINE THICKNESS.....	36
GRID TYPE.....	36
PICTURE .....	36
WIRE .....	36

SURFACE.....	36
CONTOUR.....	36
FILL CONTOURS.....	36
CONTOUR LINES.....	36
NUMBER OF CONTOURS.....	36
HARDWARE FONT.....	37
VECTOR FONT.....	37
XFONT.....	37
GRAPHICS FONT SIZE.....	37
MAX/MIN.....	37
FWHM.....	37
RISE/FALL.....	37
BEST GUESS.....	38
PEAK DOWN.....	38
ENDS IN.....	38
COMPARE BASELINE FIXED.....	38
COMPARE BASELINE VARIABLE.....	38
FILE FILTER.....	38
DATA FORMAT.....	39
LINEAR.....	39
LOG.....	39
ROUNDED.....	39
EXACT.....	39
EXTENDED.....	39
SUPPRESS 0.....	39
MINOR TICKS AUTO.....	40
MINOR TICKS OFF.....	40
Tools Menu.....	41
TOOLS.MACRO.....	41
TOOLS.MACRO.Single File.....	42
TOOLS.MACRO.Sequence of Files.....	43
TOOLS.SHOW REGISTERS.....	43
TOOLS.SHOW LIMITS.....	43
TOOLS.SHOW NOTES.....	43
TOOLS.SHOW FILE NOTES.....	44
TOOLS.SHOW HEADER.....	44
TOOLS.XFONT.....	44
TOOLS.COLOR TABLE.....	44
TOOLS.PALETTE.....	44
Help Window.....	45
Operators.....	47
OPERATOR OVERLOADING.....	47

ADD .....	49
SUBTRACT .....	49
MULTIPLY .....	49
DIVIDE .....	50
ABS .....	50
AVERAGE .....	50
EXPONENTS .....	50
POWER .....	50
INVERSE .....	50
SQRT .....	51
EXP .....	51
LN .....	51
PWR10 .....	51
LOG10 .....	51
UNITS .....	51
SCALE AXIS .....	51
CEILING .....	51
FLOOR .....	51
WINDOW .....	51
SHIFT .....	52
BASELINE .....	52
COMPARE .....	52
NORMALIZE .....	52
INTEGRATE .....	53
DIFFERENTIATE .....	53
TRANSFORMS .....	53
FFT .....	54
IFFT .....	54
CONVOLVE .....	54
CROSS CORRELATE .....	54
MIRROR .....	54
ROTATE .....	55
MIRROR X=0 .....	55
MIRROR Y=0 .....	55
MIRROR X=Y .....	55
MIRROR X=-Y .....	55
UNWRAP .....	55
FILTERS .....	55
ERODE&DILATE .....	57
LEE FILTER .....	57
BOXCAR SMOOTH .....	57
HISTOGRAM EQUALIZE .....	57
ROBERTS EDGE .....	58

SOBEL EDGE .....	58
SPECIAL .....	59
EDGE ALIGN .....	59
PEAK ALIGN .....	60
Y at X.....	60
X at Y.....	60
CONCATENATE.....	60
INTERPOLATE.....	60
SPECTRUM .....	61
CALORIMETER.....	61
CABLE COMPENSATE .....	61
DEDROOP.....	62
DEFIDU.....	62
GENERATE COMPENSATOR .....	62
LINE PROFILE .....	63
RHO PROFILE .....	63
X-Y PROFILE .....	64
CENTER.....	64
FILM2484.....	64
USER1 .....	64
USER2 .....	64
USER3 .....	64
USER4 .....	64
Speed Buttons .....	65
WAVEFORM/IMAGE.....	65
ENTER.....	65
CLEAR .....	65
SELECT.....	65
ALL.....	65
PLOT.....	66
ZOOM.....	66
PRINT SCREEN.....	65
CURSOR .....	66
H BARS .....	66
V BARS .....	66
Waveform/Image Buttons .....	69
Problems and Troubleshooting .....	71
Macintosh Specific Issues .....	73
Windows Specific Issues .....	74
HDF Data File Organization .....	75
HDF read and write IDL procedures .....	77
READ HDF FILE PROCEDURE.....	77

WRITE HDF FILE PROCEDURE .....82  
References .....87  
List of Distribution Files .....88

---

*List of Figures*

---

Figure 1	xdamp screen layout in waveform mode .....	17
Figure 2	xdamp screen layout in image mode.....	18
Figure 3	Windows file selection widget .....	19
Figure 4	Options Window Layout.....	33
Figure 5	EXPONENTS subwindow.....	50
Figure 6	Waveform TRANSFORMS subwindow .....	53
Figure 7	Image TRANSFORMS window .....	55
Figure 8	Waveform FILTERS input window .....	56
Figure 9	Image FILTERS Window.....	57
Figure 10	Waveform SPECIAL selection window.....	59
Figure 11	Image SPECIAL selection window .....	63

---

*List of Tables*

---

Table 1	xdamp.ini Initialization File Commands .....	13
Table 2	xdamp command translations to xdamp version 3.....	15
Table 3	Operations that can use the cursor .....	34
Table 4	Sample Macro to calculate power-weighted mean voltage...41	
Table 5	Automatic Register Names .....	47
Table 6	Register Names and Contents from COMPARE Operation .52	

THIS PAGE INTENTIONALLY LEFT BLANK

---

## Introduction

---

### PHILOSOPHY

*xdamp* is licensed software. To obtain a license, contact William P. Ballard at [wballa@sandia.gov](mailto:wballa@sandia.gov). If you are licensed, you will receive automatic notification of updates, available at the restricted web site <http://stl.sandia.gov/www/xdamp.html>.

*xdamp* relies, in as many places as possible, on IDL maintained routines. This choice should allow *xdamp* to age gracefully as Research Systems, Inc. will do most of the updating of the operations that actually modify the internal data. Nearly all of the procedures are precompiled when *xdamp* is started because the IDL macro programming language is interpreted at run time. Precompiling causes *xdamp* to pause initially while nearly everything is compiled, but results in faster execution times when manipulating data.

*xdamp* loads all the data into a set of working waveform arrays and image matrices and then closes the data file. It creates a button for each waveform/image and all manipulations can be done by pushing waveform/image buttons, followed by operator buttons. Two stacks are maintained, one for images and one for waveforms. Thus, you can switch modes easily and change back to the same state after manipulating data in the other mode without losing the waveform or image selections. You can choose whether the stacks are maintained between operations or are automatically cleared.

The original contents of the file are unchanged until you actively overwrite the original file. When attempting to overwrite a file, you are asked if you are sure you want to do this. However, *xdamp* minimizes how often it asks for confirmation of actions.

*xdamp* saves data using the Hierarchical Data Format (HDF) developed by the National Center for Supercomputing Applications (NCSA). IDL provides all the requisite HDF file handling utilities for all of the computing platforms of interest. Full libraries of FORTRAN and C routines are available from NCSA at <http://hdf.ncsa.uiuc.edu/>. The contents of the graphics area may be written to a file in a number of standard graphics formats such as EPS, JPEG, GIF, TIFF, and PICT. Waveforms and images may also be written out in an ASCII format for interchange with other software packages. However, these ASCII files are very large compared to the HDF files.

Whenever an operation is performed, the results are immediately and automatically plotted out (if the automatic plot feature is on). Also, the relevant pulse parameters (maximum, minimum, pulse width, rise time, and fall time) are calculated. When plotting images, only the image on the top of the stack is plotted.

*xdamp* is designed to be self documenting. There are areas in the data file for notes relating to the overall data set and notes for each individual waveform/image. *xdamp* keeps an audit trail of the operations performed on each waveform/image that change its contents. The audit trail and the individual notes are displayed whenever a single waveform or image is graphed.

Automatic execution of command (macro) files is supported. This feature allows the user to create new procedures from combinations of existing operations. These files can be nested up to 10 deep to create complicated operations from a sequence of simple macros.

*xdamp* automatically creates a journal file named **xdamp.jnl** in the current directory. The journal file can be copied and edited to easily create a macro file. Whenever a “summary” operation is performed, a spreadsheet file called **xdamp.ss** is created. This contains space delimited information with the name, the maximum and minimum values, the rise and fall times, as well as the full-width-at-half-maximum pulse widths for all selected waveforms or all waveforms if none are selected.

There is a powerful and flexible initialization file called **xdamp.ini** that can be used to customize each session according to the desires of the particular user. The maximum size and number of waveforms and images can be specified in this file.

When performing any operation with an inherent order, the ENTER button must be used to select one of the waveforms or images. Usually this defines the waveform/image to be overwritten; for instance, when performing an addition of two waveforms. In some instances the ENTER button is used to define the reference waveform (COMPARE, aligning waveforms, and when generating cable compensators). In the case of inverse fast Fourier transforms (IFFTs), the ENTER button is used to designate the real portion of the waveform to be transformed.

The limits on the number of possible waveforms/images and their maximum sizes are relatively arbitrary. These limits can be seen by using the TOOLS.SHOW LIMITS menu selection. The system manager can increase the defaults at will (in the **XIsys\_ini.pro** module in the **XIsys\_sp.pro** file) but the user has independent control through his individual **xdamp.ini** initialization file. Extremely long waveforms are possible but may create difficulties on computers with limited memory. We recommend that you keep the maximum number and maximum length of both waveforms and images close to what your problem actually requires. *xdamp* automatically allocates the required internal working data space so you do not need to consider this when sizing the number of arrays or images.

Some of the operators create new waveforms or images, if space permits. If space does not permit, the original waveform/image is *overwritten* unless more than one waveform/image is generated, in which case the operation fails with an error message. To conserve waveform/image space, if an operation that creates new waveforms is repeated, and would duplicate some waveform/image names, the old waveforms are automatically overwritten.

## INSTALLATION

The *xdamp* source code is provided in a tar file named **xdamp.tar** or a zip file **xdamp.zip**. It should be installed in the directory **\$IDL\_PATH/lib/xdamp** using the command “tar -xvf **xdamp.tar**” or an appropriate PKUNZIP utility. Be certain the directory has public read permission and that all files in the directory also have this permission. Almost all system-specific information is contained in the file **XIsys\_sp.pro**. The system manager must edit the **XIsys\_ini.pro** module in this file and set the default print queue name. However, VMS systems use the **xdamp\$PRINTER** logical for this and don't need editing, just definition of the logical. The default limits for the number of waveforms (maxarrays), their maximum length (maxsize), the number of images (maximages), their maximum size (nxmax, nymax), and the default graphics area (xgsize, ygsize) can be changed in the **XIsys\_ini** module in this file also. Finally, some customizing of the waveform/image button area is possible in this file.

**RESTRICTIONS**

*xdamp* must use IDL version 5.0 or later, for Macintosh, Unix and PC-based systems. *xdamp* requires a three-button mouse (most seem to work) to operate on windows computers.

**GETTING STARTED**

The IDL environment variables must be defined as described in the IDL installation instructions for the specific computer environment. The *xdamp* files must be installed in the (publicly accessible) \$IDL\_PATH/lib/*xdamp* directory. Then the user can access *xdamp* simply by initiating IDL (type *idl* at the prompt or click on the icon) and then typing *xdamp* at the first IDL prompt. All subsequent inputs are funneled through the main *xdamp* window. If you install *xdamp* elsewhere, the IDL software will not automatically find *xdamp*.

For Unix systems, you can control the background color of the IDL widgets and the text font used for these widgets (the graphics font is completely independent). The following lines (with sample colors and font) may be inserted in your Xdefaults file. Be careful, if you choose too large a font, the graphics area will not be next to the controls and will be too small; if the screen does not appear similar to Fig. 1, then reduce the font size. I have had good success using a bold font for the widget font. Alternatively, reduce the graphics area size using the GRAPHIC AREA=XSIZExYSIZE in the initialization file (see next topic). You can also reserve a block of color indices so that other applications won't destroy the color maps with the last entry in the list below.

```
Idl*background: LightSteelBlue
```

```
Idl*foreground: Black
```

```
Idl*fontList: ROMB12
```

```
Idl.colors: 64 (the period, rather than asterisk, is correct here)
```

For PC systems, the *wfont* command in the *xdamp.ini* file allows control of the default fonts for buttons and the default appearance of IDL in general. Macintosh computers have a similar capability through the File Preferences menu.

**INITIALIZATION FILE**

There is an optional initialization file that may reside in the subdirectory from which you will execute *xdamp*. Its name must be *xdamp.ini*. Different versions may be in different subdirectories. The following commands, one per line, may be in this file in any order and will define the options preferences. The commands are not case sensitive, although the argument may be (for instance, the file filter). Further information on these controls may be found in the Options section (page 33). A sample *xdamp.ini* file resides in \$IDL\_PATH/lib/*xdamp*. Custom versions can easily be generated with the Write Initialization File button under the Options command.

**TABLE 1. *xdamp.ini* Initialization File Commands**

Controls this feature	Default	Alternates
Maximum number of waveforms	Maxarrays=	Default in XIsys_sp.pro or use <i>xdamp.ini</i> file
Maximum length of waveforms	Maxsize=	Default in XIsys_sp.pro or use <i>xdamp.ini</i> file

TABLE 1. xdamp.ini Initialization File Commands

Controls this feature	Default	Alternates
Maximum number of images	Maximages=	Default in XIsys_sp.pro or use xdamp.ini file
Maximum size of images	Maximagesize=1024x1024	Default in XIsys_sp.pro or use xdamp.ini file
Graphic area dimensions in pixels	Graphic Area=800x750	Default in XIsys_sp.pro or use xdamp.ini file
Cursor or prompt inputs	Cursor On	Cursor Off
Automatic plotting	Autoplot On	Autoplot Off
Clear Stack After Operation	Clear Stack Off	Clear Stack On
Print device	PS	CPS
Unix print queue	Queue Name=	Default in XIsys_sp.pro or use xdamp.ini file
Plots/page	One/page	Two/page, Four/page
Date on plots	Today's Date	Shot Date
Pretty plot style	Pretty Off	Pretty On
Plot with symbols	Symbols Off	Symbols On
Screen plot colors	Monochrome	Color
Line thickness	Line thick=1.0	Number >0.2, <5.0
Grid appearance	Grid Off	Grid On, Grid Zero
Image plot style	Picture	Wire, Surface, Contour
Contour fill	Contour fill off	Contour Fill on
Contour lines	Contour line off	Contour line on
Number of contours	10	Any number
Windows font (PCs)	Wfont=SYSTEM	Any valid windows font such as Wfont=ARIAL*BOLD*14
Graphics font type	Hardware Font	Vector Font
Graphics Hardware Font <sup>a</sup>	Gfont=FONT	Font must be fully qualified <sup>b</sup>
Vector font size multiplier <sup>c</sup>	Font size = 1.0	Number >0.2, <5.0
Display max/min	Max/min On	Max/min Off
Display FWHM	FWHM On	FWHM Off
Display rise/fall	Rise/fall On	Rise/fall Off
Pulse parameter method	Best Guess	Peak Down, Ends In
Compare baseline fixed or variable during curve fit	Compare baseline variable	Compare baseline fixed
Default file filter	File filter=*.hdf	Any wildcard string
File data encoding for read	DAMP	DAMP, SICDAS, EXCEL, ASCII, USER

TABLE 1. *xdamp.ini* Initialization File Commands

Controls this feature	Default	Alternates
To read and save raw DAMP waveforms (:U)	DAMPU	DAMPNOU
X-axis	Xlinear	Xlog
X-axis endpoints	Xrounded	Xexact, Xextended
X-axis zero in graphics area	Xzero On	Xzero Off
X-axis minor tick marks	Xminor Auto	Xminor Off
Y-axis	Ylinear	Ylog
Y-axis endpoints	Yrounded	Yexact, Yextended
Y-axis zero in graphics area	Yzero On	Yzero Off
Y-axis minor tick marks	Yminor Auto	Yminor Off
Z-axis	Zlinear	Zlog
Z-axis endpoints	Zrounded	Zexact, Zextended
Z-axis zero in graphics area	Zzero On	Zzero Off
Z-axis minor tick marks	Zminor Auto	Zminor Off

a. Valid only if Hardware Font selected

b. Example: -adobe-helvetica-medium-r-normal--14-140-75-75-p-77\*

c. Valid only if Vector Font selected

You must be exceedingly careful when setting the graphics hardware font to choose a legal definition. If in doubt, leave this blank and use the *xfont* utility in the OPTIONS window to find a legal font name and then include it in the initialization file.

## FINISHING UP

To leave *xdamp*, select FILE and then EXIT from the *xdamp* menu. This action returns you to the IDL prompt. Then type EXIT to return to the system prompt. On a Macintosh, *DO NOT* use the FILE.QUIT on the screen menu bar because this will not properly close the journal file or any other open files.

## VERSION 3 CHANGES

Every feature of prior *xdamp* versions has been preserved in Version 3. If you do not want image capability at all, simply set `maximages=0` and the image processing features of *xdamp* will be disabled. However, a number of *xdamp* commands were renamed to make it more obvious how they would operate on images. Therefore, old macro files will usually not work. The following table shows the replacement commands for old *xdamp* commands. Commands that are in grayed boxes are not grandfathered to still work in automatic mode in version 3.

TABLE 2. *xdamp* command translations to *xdamp* version 3

old <i>xdamp</i> command	Version 3 <i>xdamp</i> command
YUNITS	UNITS
TRUNCATE BEFORE	WINDOW
TRUNCATE AFTER	WINDOW

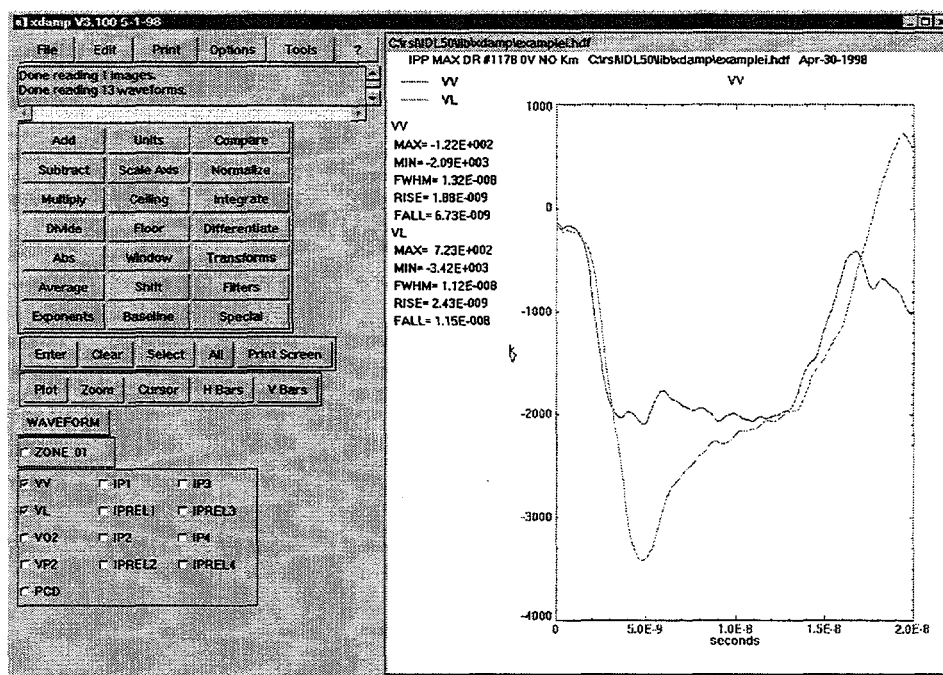
TABLE 2. *xdamp* command translations to *xdamp* version 3

old <i>xdamp</i> command	Version 3 <i>xdamp</i> command
TIMESHIFT	SHIFT
TIMEALIGN EDGE ALIGN	SPECIAL EDGE ALIGN
TIMEALIGN PEAK ALIGN	SPECIAL PEAK ALIGN
TIMEBASE	SCALE AXIS
CABLES CABLE COMPENSATE	SPECIAL CABLE COMPENSATE
CABLES DEDROOP	SPECIAL DEDROOP
CABLES DEFIDU	SPECIAL DEFIDU
CABLES GENERATE COMPENSATOR	SPECIAL GENERATE COMPENSATOR
MISCELLANY INTERPOLATE	SPECIAL INTERPOLATE
MISCELLANY CONCATENATE	SPECIAL CONCATENATE
MISCELLANY SPECTRUM	SPECIAL SPECTRUM
MISCELLANY MIRROR	TRANSFORM MIRROR
MISCELLANY CALORIMETER	SPECIAL CALORIMETER

## Screen Layout

The *xdamp* screen (Fig. 1) is designed so that input from a virtual terminal is never needed. The *xdamp* title bar displays the current version number and the date of that version. The left portion of the screen contains all the menus and buttons needed. The right portion of the screen contains the graphics region and a label area for the active file name.

FIGURE 1. *xdamp* screen layout in waveform mode

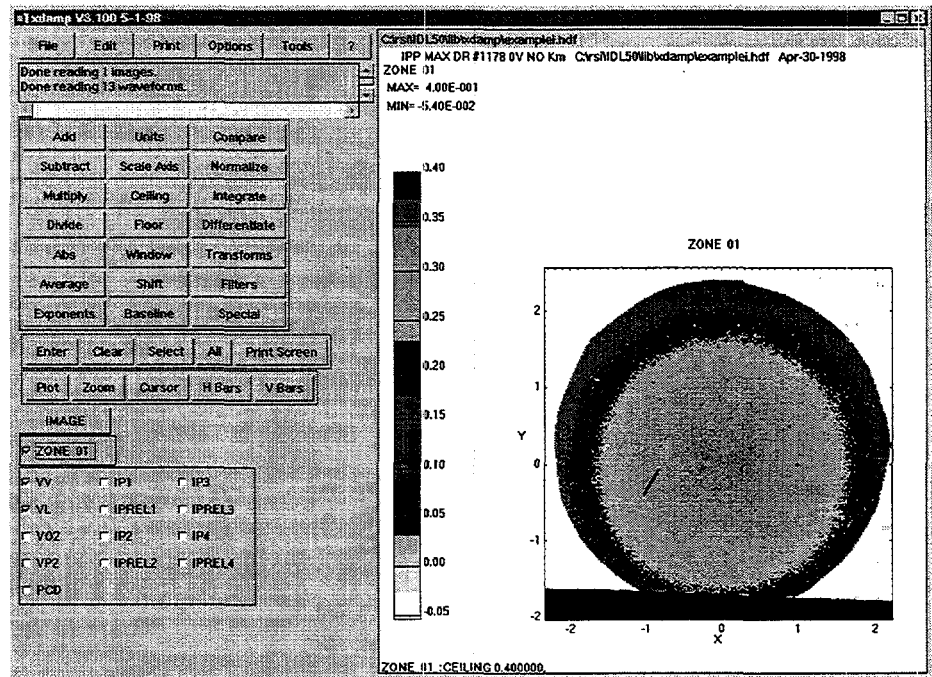


Across the top of the left portion of the *xdamp* screen is a series of “menu” buttons, not all of which actually lead to menus. The FILE, EDIT, PRINT, TOOL, and ? buttons lead to menus but the OPTIONS button pops a window with available option selections.

The space immediately below the menu bar contains a scrollable message area in which *xdamp* places output. Informational messages, general messages, error messages, and cursor values are shown in this area. For the sake of sanity, *xdamp* does not beep at you to alert you that a new message is present.

Below the message area is the operator box. Each button causes the operation described to be performed. In a few cases, such as the TRANSFORM button, subsets of further buttons appear. Only one operator button can be pressed at a time.

Below the operator area is the speed button area where buttons used extremely frequently are placed. In most cases, the same operation executed by pressing one of these buttons is available from one of the menus.

FIGURE 2. *xdamp* screen layout in image mode

Below the speed button area is the image/waveform mode selection button. Pressing this button toggles between plotting images or waveforms. Alternatively, simply selecting or deselecting an image or waveform will change modes.

Below the image/waveform mode button are the image buttons and the waveform buttons. One button is present for each of the waveforms. Multiple images/waveforms can be selected simply by pressing multiple buttons. Pressing an already set button deselects that waveform. Buttons may also be selected using wildcard strings by using the EDIT.SELECT menu selection or the Select speed button.

The following sections describe each of these actions in detail, beginning with the menus and proceeding down the left side of the screen. For the command equivalent, inputs that you choose are shown enclosed by the brackets  $\langle \rangle$ . *The punctuation in the command equivalent is critical.* Also, although the command equivalents are shown with varying cases, the actual commands are converted to uppercase so the commands are case insensitive except for waveform/image names and file names. The notation FILE.OPEN means the OPEN subcommand from the FILE menu and is also the command equivalent.

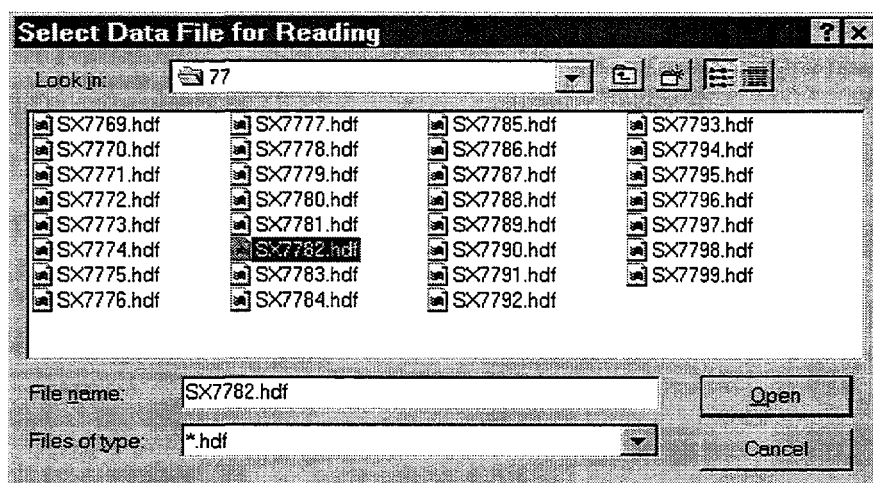
## File Menu

### FILE.OPEN

The FILE.OPEN menu selection pops a file selection widget (Fig. 3) to allow you to choose the data file to open. The default extension is defined by the case sensitive file filter defined in the initialization file or by the FILE.SET FILTER menu selection. It may be temporarily edited within the file selection widget. The widget allows you to move relatively freely between various directory tree structures. *Beware, only single mouse clicks are needed to move around on Unix systems.*

You should set the data storage type using the FILE.DATA TYPE menu item or the equivalent in the OPTIONS widget before trying to open a file. This command *automatically* discerns if the data file was written in HDF, and reads in the data appropriately. If records longer than the current maximum allowed are present, then the data are averaged over sets of points to reduce the record size. In automatic execution mode, you will *not* be queried about whether to save a currently open file when opening a new file.

FIGURE 3. Windows file selection widget



DAMP files must be in the latest variable record length format. When opening or saving a DAMP file, it is assumed that the header file exists with the same name but with extension “.list”. If this is not true, you are prompted for the header filename with the file selection widget. If no header exists, the program proceeds with a blank header area.

SICDAS data files from the SPHINX accelerator are in ASCII format but have the header information in the file and use a format that saves the starting and ending times of the waveforms, rather than the entire time data set.

ASCII waveform input data are in the format: name, horizontal axis units, vertical axis units, number of points and a set of x-y pairs, delimited by white space. The ASCII data need not be in order or have equally spaced points, *xdamp* will sort and then resample the waveform using the smallest time interval. ASCII image data are in the format: name, x-axis units, y-axis units, z-axis units, number of x-points, number of y-points, and then x, y, z data sets. Resampling is not supported for images. If you are in auto-

matic execution mode and use a ? instead of a full file name, you will be prompted for the desired file name. You can modify the prompt by placing your query string after the ?.

**Command Equivalent**

FILE.OPEN

<full path + file name> or ?

**FILE.CLOSE**

The FILE.CLOSE menu selection will close the active data set. If any modifications to the original data have been made, then you will be asked if you wish to save the data set before the images/waveforms are cleared out. If you answer yes, then the SAVE AS sub-menu will appear to obtain a file name for saving the data. If you have already saved the file, then the NO in the command equivalent is not needed.

**Command Equivalent**

FILE.CLOSE

NO

**FILE.APPEND**

The FILE.APPEND menu selection also uses the file selection widget to choose a file to be appended to the currently opened file. Each waveform/image in the appended file will have the shot number from the file name (if any) prefixed to the waveform/image name. If there is no shot number, then one or more plus signs (+) are prefixed to the waveform/image name. The number of these is determined by the number of append operations performed. The addition of the shot number or plus signs to each file name makes certain that the names are unique. When saving a file after an append operation, the default file name is still that of the initial file opened. If you are in automatic execution mode and use a ? instead of a full file name, you will be prompted for the desired file name. You can modify the prompt by placing your query string after the ?.

**Command Equivalent**

FILE.APPEND

<full file name> or ?

**FILE.GET DATA**

The FILE.GET DATA menu selection uses the file selection widget to choose a data file. This file is opened and a widget with buttons for each waveform/image is presented. You choose which images/waveforms (which are automatically plotted) you want to append to the currently opened data set and then press the DONE button. The waveform/image names will either have the shot number from the file name prefixed to the waveform/image names or a number of plus signs (+) that depends on the number of files opened. These images/waveforms are added to the current data set. Do not use this command if the target file has only one waveform, instead use FILE.APPEND. The command equivalent is shown below for adding two waveforms "WVFM1" and "WVFM2" to the data set. If you are in automatic execution mode and use a ? instead of a full file name, you will be prompted for the desired file name. You can modify the prompt by placing your query string after the ?.

**Command Equivalent**

FILE.GET DATA

<full file name> or ?

WAVEFORM.WVFM1

WAVEFORM.WVFM2

DONE

**FILE.IMPORT IMAGE.GIF**

The FILE.IMPORT IMAGE.GIF menu selection pops a file selection widget to allow you to choose a file containing an image in GIF format. The image is translated and read into the database. If the image is too large for the allocated size, then it is resampled by choosing every second or third pixel and a warning is displayed. If you are in automatic execution mode and use a ? instead of a full file name, you will be prompted for the desired file name. You can modify the prompt by placing your query string after the ?.

**Command Equivalent**

---

**FILE.IMPORT IMAGE.GIF**

&lt;full file name&gt; or ?

**FILE.IMPORT IMAGE.HDF**

The FILE.IMPORT IMAGE.HDF menu selection pops a file selection widget to allow you to choose a file containing an image in HDF Scientific Data Set (Spyglass) format. The image is translated and read into the database. If the image is too large for the allocated size, then it is resampled by choosing every second or third pixel and a warning is displayed. If you are in automatic execution mode and use a ? instead of a full file name, you will be prompted for the desired file name. You can modify the prompt by placing your query string after the ?.

**Command Equivalent**

---

**FILE.IMPORT IMAGE.HDF**

&lt;full file name&gt; or ?

**FILE.IMPORT IMAGE.JPG**

The FILE.IMPORT IMAGE.JPG menu selection pops a file selection widget to allow you to choose a file containing an image in JPEG format. The image is translated and read into the database. If the image is too large for the allocated size, then it is resampled by choosing every second or third pixel and a warning is displayed. If you are in automatic execution mode and use a ? instead of a full file name, you will be prompted for the desired file name. You can modify the prompt by placing your query string after the ?.

**Command Equivalent**

---

**FILE.IMPORT IMAGE.JPG**

&lt;full file name&gt; or ?

**FILE.IMPORT  
IMAGE.PERKIN ELMER**

The FILE.IMPORT IMAGE.PERKIN ELMER menu selection pops a file selection widget to allow you to choose a file containing an image in Perkin Elmer image scanner format. The image is translated and read into the database. If the image is too large for the allocated size, then it is resampled by choosing every second or third pixel and a warning is displayed. If you are in automatic execution mode and use a ? instead of a full file name, you will be prompted for the desired file name. You can modify the prompt by placing your query string after the ?.

**Command Equivalent**

---

**FILE.IMPORT IMAGE.PERKIN ELMER**

&lt;full file name&gt; or ?

**FILE.IMPORT IMAGE.PICT**

The FILE.IMPORT IMAGE.PICT menu selection pops a file selection widget to allow you to choose a file containing an image in PICT format. The image is translated and read into the database. If the image is too large for the allocated size, then it is resampled by choosing every second or third pixel and a warning is displayed. If you are in automatic execution mode and use a ? instead of a full file name, you will be prompted

for the desired file name. You can modify the prompt by placing your query string after the ?.

**Command Equivalent**

FILE.IMPORT IMAGE.PICT

<full file name> or ?

**FILE.IMPORT  
IMAGE.TECPLOT**

The FILE.IMPORT IMAGE.TECPLOT menu selection pops a file selection widget to allow you to choose a file containing an image in Tecplot™ (available from Amtec Engineering) binary format. The image is translated and read into the database. If the image is too large for the allocated size, then it is resampled by choosing every second or third pixel and a warning is displayed. If you are in automatic execution mode and use a ? instead of a full file name, you will be prompted for the desired file name. You can modify the prompt by placing your query string after the ?.

**Command Equivalent**

FILE.IMPORT IMAGE.TECPLOT

<full file name> or ?

**FILE.IMPORT IMAGE.TIFF**

The FILE.IMPORT IMAGE.TIFF menu selection pops a file selection widget to allow you to choose a file containing an image in TIFF format. The image is translated and read into the database. If the image is too large for the allocated size, then it is resampled by choosing every second or third pixel and a warning is displayed. If you are in automatic execution mode and use a ? instead of a full file name, you will be prompted for the desired file name. You can modify the prompt by placing your query string after the ?.

**Command Equivalent**

FILE.IMPORT IMAGE.TIFF

<full file name> or ?

**FILE.EXPORT IMAGE.EPS**

The FILE.EXPORT IMAGE.EPS menu selection pops a file selection widget to allow you to choose a file name for saving the current plot area as an encapsulated postscript (EPS) data file. The default file extension is **.eps**. If you are in automatic execution mode and use a ? instead of a full file name, you will be prompted for the desired file name. You can modify the prompt by placing your query string after the ?.

**Command Equivalent**

FILE.EXPORT IMAGE.EPS

<full file name> or ?

**FILE.EXPORT IMAGE.GIF**

The FILE.EXPORT IMAGE.GIF menu selection pops a file selection widget to allow you to choose a file name for saving the current plot area as a GIF data file. The default file extension is **.gif**. If you are in automatic execution mode and use a ? instead of a full file name, you will be prompted for the desired file name. You can modify the prompt by placing your query string after the ?.

**Command Equivalent**

FILE.EXPORT IMAGE.GIF

<full file name> or ?

**FILE.EXPORT IMAGE.JPEG**

The FILE.EXPORT IMAGE.JPEG menu selection pops a file selection widget to allow you to choose a file name for saving the current plot area as a JPEG data file. The default

file extension is **.jpg**. If you are in automatic execution mode and use a ? instead of a full file name, you will be prompted for the desired file name. You can modify the prompt by placing your query string after the ?.

**Command Equivalent**

---

FILE.EXPORT IMAGE.JPEG  
<full file name> or ?

**FILE.EXPORT IMAGE.PICT**

The FILE.EXPORT IMAGE.PICT menu selection pops a file selection widget to allow you to choose a file name for saving the current plot area as a PICT data file. The default file extension is **.pic**. If you are in automatic execution mode and use a ? instead of a full file name, you will be prompted for the desired file name. You can modify the prompt by placing your query string after the ?.

**Command Equivalent**

---

FILE.EXPORT IMAGE.PICT  
<full file name> or ?

**FILE.EXPORT IMAGE.TIFF**

The FILE.EXPORT IMAGE.TIFF menu selection pops a file selection widget to allow you to choose a file name for saving the current plot area as a TIFF data file. The default file extension is **.tif**. If you are in automatic execution mode and use a ? instead of a full file name, you will be prompted for the desired file name. You can modify the prompt by placing your query string after the ?.

**Command Equivalent**

---

FILE.EXPORT IMAGE.TIFF  
<full file name> or ?

**FILE.SAVE**

The FILE.SAVE menu selection overwrites the original data file with the current contents of the waveforms and images using the Hierarchical Data Format (HDF). If you have performed an append operation, the name of the first file opened is used. You are asked if you are certain that you want to overwrite the original file before continuing with this operation *unless* you are in automatic execution mode.

**Command Equivalent**

---

FILE.SAVE

**FILE.SAVE AS**

The FILE.SAVE AS menu selection prompts for a new file name via the file selection widget so you can alter the directory, file name, or file extension before actually saving the waveforms and images using Hierarchical Data Format (HDF). The default file extension for a save as operation is **.hdf**. If the new and old file names are identical, you are asked if you are certain that you want to overwrite the original file before continuing with this operation. If you are in automatic execution mode and use a ? instead of a full file name, you will be prompted for the desired file name. You can modify the prompt by placing your query string after the ?.

**Command Equivalent**

---

FILE.SAVE AS  
<full file name> or ?

**FILE.SAVE ASCII**

The FILE.SAVE ASCII menu selection saves the selected waveforms or images in an ASCII file suitable for editing and use by many external graphics packages. If more than one waveform/image is selected, then all selected waveforms are output in a single file.

For waveforms, the output data format is the name, the horizontal axis units label, the vertical axis units label, the number of points, and then a series of x-y pairs separated by white space. For images, the output data format is the name, x-axis units label, the y-axis units label, the z-axis units label, the number of x points, the number of y points, and then (x, y, z) data tuples separated by white space. If no images/waveforms are selected, then all images or waveforms, depending on the current mode, are written to the output file. The output file name is chosen by the file selection widget with a default extension of `.asc`. If you are in automatic execution mode and use a `?` instead of a full file name, you will be prompted for the desired file name. You can modify the prompt by placing your query string after the `?`.

**Command Equivalent**

---

FILE.SAVE ASCII  
<full file name> or ?

**FILE.SET FILTER**

The FILE.SET FILTER menu selection asks for the new case sensitive file filter to be applied in future OPEN, SAVE, and SAVE AS operations. The file filter can be very sophisticated and use multiple wildcard characters (\*) or simply specify the desired file extension. Example formats are `*.hdf` or `*77*.hdf` or `saturn_023*.dat`. Filters can also be set through the OPTIONS widget.

**Command Equivalent**

---

FILE.SET FILTER  
<filter string>

**FILE.SET DATA TYPE**

The FILE.SET DATA TYPE menu selection pops a widget to set the format the data to be read were stored in. This has no effect on the output data formats. Possible responses are DAMP, SICDAS, EXCEL (comma delimited), USER, or ASCII formats as well as a CANCEL button if you don't want to change the current selection. File save operations always use the HDF format, no matter what data type is set. The USER format is distributed in the `XI_user.pro` procedure. The user may define this subroutine if desired. The data type can also be set through the OPTIONS widget.

**Command Equivalent**

---

FILE.SET DATA TYPE  
DAMP or SICDAS or EXCEL or USER or ASCII

**FILE.EXIT**

The FILE.EXIT menu selection exits *xdamp*. Prior to exiting, a FILE.CLOSE operation is performed, allowing you to save the current data set if you desire. Then, if any data remains in the print file, you are asked if you wish to spool it to the printer before exiting. This operation returns you to the IDL prompt, not the operating system. In command mode, the following sequence assumes you do not wish to save the active file. Also, an additional Yes or No may be needed if there is a print file open. The print question is asked before the question about saving the file.

**Command Equivalent**

---

FILE.EXIT  
YES or NO

---

## *Edit Menu*

---

In this context, all editing is done on images, waveforms, or a set of special storage registers. There are two methods to create new waveforms or images, NEW and COPY. These are very similar in end effect. There is no method to create a new register; you must use the predefined registers #R0 through #R9. Be aware that *xdamp* is case sensitive for waveform/image names so the names VV, Vv, vV, and vv are unique.

### EDIT.STORE

The EDIT.STORE menu selection allows you to store a number in one of the predefined registers. A constant or the contents of another register can be stored in these registers.

**Command Equivalent**

---

EDIT.STORE  
<value or #register>  
<0 through 9>

### EDIT.NEW

The EDIT.NEW menu selection creates a new waveform or image depending on the current mode. It prompts for the desired name of the new waveform/image. This name must be different from all other names of the same category in the current data base. All other data about the waveform/image is assumed to be blank until some operation targeting this waveform/image is performed. Then, the new waveform/image takes its properties (number of points, time span) from the other waveform/image in the operation.

**Command Equivalent**

---

EDIT.NEW  
<waveform or image name>

### EDIT.COPY

The EDIT.COPY menu selection copies the last waveform/image selected to a new waveform/image. You are prompted for the new waveform/image name. If the new name is identical to an existing one, you will be asked if you are certain you wish to overwrite the old waveform/image. Should you answer affirmatively, the copy command will proceed, writing over the old data with the selected waveform or image. All other data are identical to the old waveform/image.

**Command Equivalent**

---

EDIT.COPY  
<waveform/image name>

### EDIT.SORT

The EDIT.SORT menu selection sorts the waveform/image buttons in alphanumeric order from a to z within each category. The buttons are then rearranged.

**Command Equivalent**

---

EDIT.SORT

**EDIT.RENAME**

The EDIT.RENAME menu selection renames the last waveform/image selected to a new name obtained from a prompt window. The new name is forced to be unique compared to all current waveforms or images or the operation is not performed.

**Command Equivalent**

---

EDIT.RENAME  
<new waveform/image name>

**EDIT.SELECT**

The EDIT.SELECT menu selection asks for a wild card waveform/image designation and then selects all of the waveforms/images meeting that criteria. At least one asterisk must be present in the response. For example, PIN\* as the wild card input would select PIN, PIN1, and PIN007.

**Command Equivalent**

---

EDIT.SELECT  
<wildcard string>

**EDIT.DELETE**

The EDIT.DELETE menu selection deletes all the selected waveforms/images and their buttons. There is no "Are you sure?" prompt. If you have insufficient memory failures, edit the **XI\_edit.pro** file and find the **XIedit\_delete** section. There is a comment with a note to uncomment one line and comment out another line to improve the memory management at the expense of speed.

**Command Equivalent**

---

EDIT.DELETE

**EDIT.KEEP**

The EDIT.KEEP menu selection deletes all waveforms/images except the selected waveforms/images. There is no "Are you sure?" prompt. If you have insufficient memory failures, edit the **XI\_edit.pro** file and find the **XIedit\_keep** section. There is a comment with a note to uncomment one line and comment out another line to improve the memory management at the expense of speed.

**Command Equivalent**

---

EDIT.KEEP

**EDIT.RESTORE**

The EDIT.RESTORE menu selection restores all the currently selected waveforms/images from the data file to their state after the last save operation.

**Command Equivalent**

---

EDIT.RESTORE

**EDIT.NOTES**

The EDIT.NOTES menu selection allows you to edit notes attached to each waveform/image. (Notes appear in the lower left corner of the graphics area when a single waveform/image is plotted.) These notes can be printed using the **PRINT.NOTES COMMAND** or viewed using the **TOOLS.SHOW NOTES** command. Editing areas for notes about all of the selected waveforms/images, or all waveforms/images if none are selected, will appear. In automatic execution, this command returns control to you for the input of notes. Automatic execution continues when you press the **DONE** button.

**Command Equivalent**

---

EDIT.NOTES

**EDIT.FILE NOTES**

The EDIT.FILE NOTES menu selection allows you to edit notes attached to the overall data set. An editing area for the notes will appear. These notes can be printed using the PRINT.FILE NOTES COMMAND or viewed using the TOOLS.SHOW FILE NOTES command. In automatic execution, this command returns control to you for the input of file notes. Automatic execution continues when you press the DONE button.

**Command Equivalent**

EDIT.FILE NOTES

**EDIT.TITLE**

The EDIT.TITLE menu selection allows you to edit the shot title information attached to the overall data set. An editing area for the title will appear. In automatic execution, this command returns control to you for the input of the shot title. Automatic execution continues when you press the RETURN key or DONE button.

**Command Equivalent**

EDIT.TITLE

THIS PAGE INTENTIONALLY LEFT BLANK

---

## *Print Menu*

---

All printing is done using PostScript and is placed in a temporary file named **xdamp.out** in your current directory. To minimize paper, all printing occurs in append mode until you request that the file be queued. *xdamp* overrides the font selections when using PostScript output to be certain output will fit on the page. The **xdamp.out** file is not deleted upon exiting *xdamp* so you can reprint the file if needed.

**PRINT.SCREEN**

The PRINT.SCREEN menu selection causes the current graphics view to be redrawn to a file that can be printed later (see PRINT.QUEUE OUTPUT).

**Command Equivalent**

---

PRINT.SCREEN

**PRINT.ALL**

The PRINT.ALL menu selection causes all of the waveforms/images to be drawn to a file for later printing using the current selection of one, two, or four plots per page.

**Command Equivalent**

---

PRINT.ALL

**PRINT.SELECTED**

The PRINT.SELECTED menu selection causes all of the selected waveforms/images to be drawn to a file for later printing. The number of plots per page selection is honored for this process but only one plot to a plot area.

**Command Equivalent**

---

PRINT.SELECTED

**PRINT.REGISTERS**

The PRINT.REGISTERS menu selection causes the contents of the registers to be sent to the print file.

**Command Equivalent**

---

PRINT.REGISTERS

**PRINT.SUMMARY**

The PRINT.SUMMARY menu selection places a summary sheet in the file. For waveforms, the summary sheet contains the waveform name, maximum, minimum, vertical units, full width at half maximum, rise time, fall time and time units for each selected waveform. For images, the summary sheet contains the image name, maximum, minimum, x units, y units, and z units. If no waveforms/images are selected, the information for all waveforms/images is printed. This command also creates an ASCII file called **xdamp.ss** suitable for inclusion in a spreadsheet using single spaces as delimiters.

**Command Equivalent**

---

PRINT.SUMMARY

**PRINT.HEADER**

The PRINT.HEADER menu places the header (if one exists) of the original file opened, in the file for later printing.

**Command Equivalent**

---

PRINT.HEADER

- PRINT.NOTES**                   The PRINT.NOTES menu selection places the notes for all the selected waveforms/images in the file for later printing. If no waveforms/images are selected, then all of the notes are printed.
- Command Equivalent**  
-----  
PRINT.NOTES
- PRINT.AUDIT TRAIL**           The PRINT.AUDIT TRAIL menu selection places the audit trails for all the selected waveforms/images in a file for later printing. Audit trails are automatically generated for each image/waveform as you operate on them. If no waveforms/images are selected, then all of the audit trails are printed.
- Command Equivalent**  
-----  
PRINT.AUDIT TRAIL
- PRINT.FILE NOTES**           The PRINT.FILE NOTES menu selection places the file notes in the printer file for later printing.
- Command Equivalent**  
-----  
PRINT.FILE NOTES
- PRINT.PRINTER TYPE**         The PRINT.PRINTER TYPE menu selection pops a selection widget to choose the type of printer you have. You can either choose PS (PostScript) or CPS (Color PostScript) type printers or a Cancel button if the current selection is adequate. This can also be changed in the OPTIONS widget.
- Command Equivalent**  
-----  
PRINT.PRINTER TYPE  
PS or CPS
- PRINT.PRINTER SETUP**       The PRINT.PRINTER SETUP menu selection invokes the native printer setup utility to allow selection of output printer and some customizing of the output.
- Command Equivalent**  
-----  
PRINT.PRINTER SETUP
- PRINT.QUEUE OUTPUT**        The PRINT.QUEUE OUTPUT menu selection closes the current print file and queues it to the line printer for Unix systems. For Macintosh computers, you need to print out the **xdamp.out** file manually. One way to accomplish this is to use the shareware program DropPs available from mac.archive.umich.edu via anonymous ftp. For DOS/Windows machines, you need to make certain that the **idlspawn.pif** file was moved to the windows\system directory and was edited (using the PIF editor) to have the exclusive attribute set. On DOS/Windows machines (but not Windows 95 or NT), you will be stuck in DOS mode for the duration of the print.
- Command Equivalent**  
-----  
PRINT.QUEUE OUTPUT
- PRINT.QUEUE NAME**           The PRINT.QUEUE NAME menu selection allows Unix system users to change print queue commands. The default is set in the **XIsys\_sp.pro** file by the system manager.

The full command is required, for instance `lpr -Psata002` would use the print command `lpr` with queue `sata002`. This feature also allows you to set print options.

**Command Equivalent**

---

PRINT.QUEUE NAME  
<queue print command + name>

**PRINT.PRINTER RESET**

The PRINT.PRINTER RESET menu selection closes the current print file and does not queue it to the printer. The second argument in the command equivalent is only required if there is an uncleared print file in the queue.

**Command Equivalent**

---

PRINT.PRINTER RESET  
YES or NO

THIS PAGE INTENTIONALLY LEFT BLANK

## Options Window

Unlike the prior menus, the OPTIONS menu button pops up a window (Fig. 4) with a series of push buttons and input boxes. These allow you to select your preferred mode of viewing the data. All of the OPTIONS options may be set in the `xdamp.ini` file. The x-, y-, and z-axis preferences selections are identical and are only discussed under the equivalent x-axis heading. Multiple OPTIONS options may be placed in a series with the block of commands surrounded by the following commands. However, to perform even a single options command, you must surround it by these.

### Command Equivalent

```

OPTIONS
One or more commands here
DONE
  
```

FIGURE 4. Options Window Layout

The screenshot shows the 'X-DAMP OPTIONS' dialog box with the following settings:

- Buttons:** DONE, CURSOR, AUTO PLOT, Clear Stack After Operations, WRITE INITIALIZATION FILE
- Plotting:** PS (selected), CPS, QUEUE NAME: [ ], GRAPHIC AREA SIZE: 750 x 850
- Plot Options:** Plot 1/Page, Plot 2/Page, Plot 4/Page, Today's Date, Shot Date, PRETTY
- Symbols:** SYMBOLS, Monochrome (selected), Color, LINE THICKNESS X: 2.00000
- Grid:** PICTURE, WIRE, SURFACE, CONTOUR, GRID ZERO, GRID OFF, GRID ON
- Contours:** FILL CONTOURS, CONTOUR LINES, NUMBER OF CATEGORIES: 10,000
- Font:** HARDWARE FONT, VECTOR FONT, FONT SIZE X: 12,000
- Other:** MAXMIN, FWHM, RISE/FALL, BEST GUESS, PEAK DOWN, ENDS IN
- Compare:** COMPARE BASELINE FIXED, COMPARE BASELINE VARIABLE
- File Filter:** \*.hdf, DAMP, SICDAS, EXCEL, ASCII, USER
- X-AXIS PREFERENCES:** XLINEAR, XLOG, XROUNDED, XEXACT, XEXTENDED, XSUPPRESS ZERO, XMINORTICKS OFF, XMINORTICKS AUTO
- Y-AXIS PREFERENCES:** YLINEAR, YLOG, YROUNDED, YEXACT, YEXTENDED, YSUPPRESS ZERO, YMINORTICKS OFF, YMINORTICKS AUTO
- Z-AXIS PREFERENCES:** ZLINEAR, ZLOG, ZROUNDED, ZEXACT, ZEXTENDED, ZSUPPRESS ZERO, ZMINORTICKS OFF, ZMINORTICKS AUTO

### CURSOR

The CURSOR preference button determines how inputs will be given to `xdamp` for truncation, time shifting, zooming, and baselining operations. If the cursor button is down it is ON and the cursor will become active for these inputs. The current cursor coordinates are shown continuously in the message window. You must press a mouse button to pick

the final desired cursor-input position. If cursor mode is not selected, a numeric input box will automatically appear. Only a few operators, shown in Table 3, use the cursor.

**TABLE 3. Operations that can use the cursor**

Waveform Mode	Image Mode
CURSOR	CURSOR
H BARS	
VBARS	
ZOOM	ZOOM
BASELINE	BASELINE
SHIFT	SHIFT
CEILING	CEILING
FLOOR	FLOOR
WINDOW	WINDOW
INTERPOLATE	CENTER
DEFIDU	RHO PROFILE
CONCATENATE	LINE PROFILE
CALORIMETER	

The CURSOR command equivalent toggles the current state. In command mode, if the cursor preference is on, the automatic execution pauses for cursor input. If the cursor preference is off, the cursor position was saved (in data units) and is reused. Alternatively, you may use CURSOR ON or CURSOR OFF (with or without the OPTIONS and DONE) if you are uncertain of the state of the CURSOR command.

**Command Equivalent**

CURSOR or CURSOR ON or CURSOR OFF

**AUTO PLOT**

The AUTO PLOT preference button determines whether selected waveforms/images will be immediately plotted, or if the PLOT button must be used to make a plot. This is a toggle button. The command equivalent to toggle the state is shown below. Alternatively, you may use AUTO PLOT ON or AUTO PLOT OFF (with or without the OPTIONS and DONE) if you are uncertain of the state of the AUTO PLOT command.

**Command Equivalent**

AUTO PLOT or AUTO PLOT ON or AUTO PLOT OFF

**Clear Stack After Operations**

The Clear Stack after Operations preference button determines whether the stack will be automatically cleared after each operation. This is a toggle button. The command equivalent to toggle the state is shown below.

**Command Equivalent**

CLEAR STACK

**MAKE INITIALIZATION FILE**

The MAKE INITIALIZATION FILE button allows the creation of an **xdamp.ini** file containing all the currently enabled settings. The file selection widget allows you to

choose a different location or name for the initialization file but only the name **xdamp.ini** will be recognized at start-up.

**PS**  
**CPS**

The PS and CPS buttons are mutually exclusive. These determine what type of output device is in use. The PS button is monochrome PostScript and the CPS button is color PostScript.

**Command Equivalent**

PS or CPS

**QUEUE NAME**

The QUEUE NAME input area allows Unix system users to change print queue commands. The default is set in the **XIsys\_sp.pro** file by the system manager. The full command is required, for instance `lpr -Psata002` would use the print command `lpr` with queue `sata002`.

**Command Equivalent**

QUEUE NAME = <Queue command and name>

**GRAPHIC AREA SIZE**

The GRAPHIC AREA SIZE input area allows you to interactively change the allowed plotting area. The default is set in the **XIsys\_sp.pro** file by the system manager. The input is in pixels, (width x height) and the lower case x is required between the two arguments. The plotting area will resize when you hit a carriage return or the DONE button. This command is best set using the Graphic Area = command in the **xdamp.ini** initialization file.

**Command Equivalent**

GRAPHIC AREA = <Xpixels> x <Ypixels>

**PLOT #/PAGE**

The PLOT 1/PAGE, PLOT 2/PAGE, and PLOT 4/PAGE buttons are mutually exclusive. These determine whether one, two, or four plots per page will be shown both on the screen and for hardcopies.

**Command Equivalent**

PLOT 1/PAGE or PLOT 2/PAGE or PLOT 4/PAGE

**SHOT DATE**  
**TODAYS DATE**

The SHOT DATE and TODAYS DATE buttons are mutually exclusive. If a shot date was encoded in the data file, then that date and time are shown on each plot when SHOT DATE is selected. Otherwise, if TODAYS DATE is selected, the current date is placed on each plot.

**Command Equivalent**

SHOT DATE or TODAYS DATE

**PRETTY**

The PRETTY button toggles between pretty mode and normal mode. Pretty mode suppresses the file name and shot date in the plot title as well as the audit trail printing. This allows nicer looking graphics to be generated for presentations. PRETTY OFF is the default.

**Command Equivalent**

PRETTY or PRETTY ON or PRETTY OFF

**SYMBOLS**

The SYMBOLS button determines whether symbols will be used on each plot in addition to the line. If it is pressed, then 25 symbols across the plot will be used. In com-

mand mode, this is a toggle command. Alternatively, you may use SYMBOLS ON or SYMBOLS OFF (with or without the OPTIONS and DONE) if you are uncertain of the state of the SYMBOLS command.

**Command Equivalent**

SYMBOLS or SYMBOLS ON or SYMBOLS OFF

**MONOCHROME  
COLOR**

The MONOCHROME and COLOR buttons are mutually exclusive and choose what type of screen display will be generated. If COLOR is chosen, then all linetypes are solid while if MONOCHROME is chosen, all lines are different. The color map is determined by the TOOLS.COLOR TABLE. This button also determines whether PostScript output files are in color or black and white.

**Command Equivalent**

MONOCHROME or COLOR

**LINE THICKNESS**

The LINE THICKNESS MULTIPLIER determines how thick the graphics lines will be. The default is 1.0 and making lines too thick significantly slows down the graphics response. Valid values are between 0.2 and 5.0. The equals sign in the command equivalent is required.

**Command Equivalent**

LINE THICKNESS = <Line Thickness Multiplier>

**GRID TYPE**

The GRID ZERO, GRID OFF and GRID ON buttons are mutually exclusive. If GRID OFF is selected, no grid is shown. If GRID ON is selected, then a dotted grid is shown at the major tick marks. If GRID ZERO is selected, then only the x and y axis zero values of the grid are drawn as bold lines.

**Command Equivalent**

GRID ON or GRID OFF or GRID ZERO

**PICTURE  
WIRE  
SURFACE  
CONTOUR**

The PICTURE, WIRE, SURFACE and CONTOUR buttons select the plotting mode for images. PICTURE treats the image as a 8-bit color bitmap when plotting. WIRE plots a 3-D wire frame version of the image. SURFACE plots a color, shaded 3-D wire frame view of the image. CONTOUR plots a contour level version of the image according to whether the contours are chosen to be filled and/or have lines at the different levels.

**FILL CONTOURS**

The FILL CONTOURS button selects whether the contour levels will be filled with solid colors. This choice is independent of the CONTOUR LINE selection.

**CONTOUR LINES**

The CONTOUR LINES button selects whether lines separating the contours are drawn. Used in conjunction with the independent FILL CONTOURS button, numerous different effects are possible.

**NUMBER OF CONTOURS**

The NUMBER OF CONTOURS entry box allows you to choose the number of contour levels desired in the plot. The equals sign in the command equivalent is required.

**Command Equivalent**

NUMBER OF CONTOURS = <Number>

**HARDWARE FONT  
VECTOR FONT**

The VECTOR FONT button selects the IDL Hershey vector fonts for the graphics area. The HARDWARE FONT button selects the terminal default. The hardware fonts can be changed through either the XFONT button or the TOOLS.XFONT menu item for Unix systems.

**Command Equivalent**


---

VECTOR FONT or HARDWARE FONT

**GRAPHICS FONT SIZE**

The GRAPHICS FONT SIZE MULTIPLIER is only active for vector fonts. It modifies all the text in the graphics window. Valid values are 0.2 to 5.0. If you have selected hardware fonts, this field is inactive and the font size multiplier is set to 1.0. The equals sign in the command equivalent is required.

**Command Equivalent**


---

FONT SIZE = <Font Size Multiplier>

**XFONT**

The XFONT button pops up the xfont selector widget if HARDWARE FONT has been selected on Unix computers. This widget allows you to preview and select a hardware font from all of those available to your terminal. This font becomes the graphics area font. This application has no command equivalent but a desired font can be placed in the **xdamp.ini** file. This application does not function for Macintosh or Windows computers.

**MAX/MIN**

The MAX/MIN toggle button selects whether the maximum and minimum values of the selected waveforms/images will be shown on the left side of each plot (in ONE/page mode). This button is not mutually exclusive with either the FWHM or RISE/FALL buttons. These quantities are calculated every time a waveform/image is plotted, whether or not MAX/MIN is selected, and are stored in registers. Alternatively, you may use MAX/MIN ON or MAX/MIN OFF (with or without the OPTIONS and DONE) if you are uncertain of the state of the MAX/MIN command.

**Command Equivalent**


---

MAX/MIN or MAX/MIN ON or MAX/MIN OFF

**FWHM**

The FWHM toggle button selects whether the full-width at half-maximum (FWHM) values of the selected waveforms will be shown on the left side of each plot (in ONE/page mode). This button is not mutually exclusive with either the MAX/MIN or RISE/FALL buttons. This quantity is calculated every time a waveform is plotted, whether or not FWHM is selected, and is stored in a register. Alternatively, you may use FWHM ON or FWHM OFF (with or without the OPTIONS and DONE) if you are uncertain of the state of the FWHM command.

**Command Equivalent**


---

FWHM or FWHM ON or FWHM OFF

**RISE/FALL**

The RISE/FALL toggle button selects whether the 10%-to-90% rise-time and fall-time values of the selected waveforms will be shown on the left side of each plot (in ONE/page mode). This button is not mutually exclusive with either the MAX/MIN or FWHM buttons. These quantities are calculated every time a waveform is plotted, whether or not RISE/FALL is selected, and are stored in registers. Alternatively, you may use RISE/

FALL ON or RISE/FALL OFF (with or without the OPTIONS and DONE) if you are uncertain of the state of the RISE/FALL command.

**Command Equivalent**

RISE/FALL or RISE/FALL ON or RISE/FALL OFF

**BEST GUESS**

The BEST GUESS button is mutually exclusive with the PEAK DOWN and ENDS IN buttons. It selects the mode used to calculate the FWHM and rise and fall times. The algorithm used calculates all parameters using both the PEAK DOWN and ENDS IN techniques. The longest FWHM and the shortest rise and fall times are then selected.

**Command Equivalent**

BEST GUESS

**PEAK DOWN**

The PEAK DOWN button is mutually exclusive with the BEST GUESS and ENDS IN buttons. It selects the mode used to calculate the FWHM and rise and fall times. The algorithm is to calculate all parameters starting from the waveform peak and proceeding outward to find the first 90%, 50%, and 10% values for the calculations.

**Command Equivalent**

PEAK DOWN

**ENDS IN**

The ENDS IN button is mutually exclusive with the BEST GUESS and PEAK DOWN buttons. It selects the mode used to calculate the FWHM and rise and fall times. The algorithm is to calculate all parameters starting from the beginning and the end of the waveform to find the first 10%, 50% and 90% values for the calculations.

**Command Equivalent**

ENDS IN

**COMPARE BASELINE  
FIXED**

The COMPARE BASELINE FIXED button is mutually exclusive with the COMPARE BASELINE VARIABLE button. These buttons control how the COMPARE routine performs curve fits. In fixed mode, the baseline is not allowed to vary and the two waveforms are fit by the equation  $\text{Reference} = \text{Scale} * \text{Variable}$ . No offset is allowed.

**Command Equivalent**

COMPARE BASELINE FIXED

**COMPARE BASELINE  
VARIABLE**

The COMPARE BASELINE VARIABLE button is mutually exclusive with the COMPARE BASELINE FIXED button. These buttons control how the COMPARE routine performs curve fits. In variable mode, the baseline is allowed to vary and the two waveforms are fit by the equation  $\text{Reference} = \text{Scale} * \text{Variable} + \text{Offset}$ .

**Command Equivalent**

COMPARE BASELINE VARIABLE

**FILE FILTER**

The FILE FILTER entry box allows you to change the default file filter to be applied in future OPEN, SAVE, and SAVE AS operations. This is identical to changing the filter in the FILE.SET FILTER menu selection. The file filter can be very sophisticated and use

multiple wild card characters (\*) or simply specify the desired file extension. Example formats are \*.hdf or saturn\_012\*.dat.

**Command Equivalent**

FILE FILTER = <filter string>

**DATA FORMAT**

The DATA FORMAT selection buttons allow you to change the default data structure for the input file. Whatever the selection is, HDF files are always automatically recognized for reading. This command is identical to changing the selection in the FILE.SET DATA TYPE menu selection. As distributed, the EXCEL format reads comma delimited spreadsheet format data. The USER format is in the file **XI\_user.pro** segment of the **XI\_file.pro** file and must be written by the user.

**Command Equivalent**

DAMP or SICDAS or EXCEL or USER or ASCII

**LINEAR**

The LINEAR button is mutually exclusive with the LOG button. This selects whether the axis will be linear or logarithmic.

**Command Equivalent**

XLINEAR or YLINEAR or ZLINEAR

**LOG**

The LOG button is mutually exclusive with the LINEAR button. This selects whether the axis will be linear or logarithmic. For log axes, negative and zero values are ignored.

**Command Equivalent**

XLOG or YLOG or ZLOG

**ROUNDED**

The ROUNDED button is mutually exclusive with the EXACT and EXTENDED buttons. This selects rounded axis limits where IDL chooses the axis limits after examining the waveforms/images.

**Command Equivalent**

XROUNDED or YROUNDED or ZROUNDED

**EXACT**

The EXACT button is mutually exclusive with the ROUNDED and EXTENDED buttons. This selects exact axis limits and IDL chooses the axis limits after examining the waveforms/images.

**Command Equivalent**

XEXACT or YEXACT or ZEXACT

**EXTENDED**

The EXTENDED button is mutually exclusive with the ROUNDED and EXACT buttons. This selects extended axis limits where IDL chooses the axis limits after examining the waveforms/images and then adds approximately 20% to both ends.

**Command Equivalent**

XEXTENDED or YEXTENDED or ZEXTENDED

**SUPPRESS 0**

The SUPPRESS 0 toggle button is not mutually exclusive with the ROUNDED, EXACT and EXTENDED buttons. Normally, IDL will choose to have the value 0

included in the axis limits for both axes. This button removes this restriction and is akin to ac-coupling an oscilloscope.

**Command Equivalent**

XSUPPRESS ZERO or  
YSUPPRESS ZERO or  
ZSUPPRESS ZERO

**MINOR TICKS AUTO**

The MINOR TICKS AUTO button is mutually exclusive with the MINOR TICKS OFF button. This enables minor axis tick marks and IDL chooses the spacing after examining the waveforms/images.

**Command Equivalent**

X MINOR TICKS AUTO or  
Y MINOR TICKS AUTO or  
Z MINOR TICKS AUTO

**MINOR TICKS OFF**

The MINOR TICKS OFF button is mutually exclusive with the MINOR TICKS AUTO button. This disables minor axis tick marks.

**Command Equivalent**

X MINOR TICKS OFF or  
Y MINOR TICKS OFF or  
Z MINOR TICKS OFF

---

## Tools Menu

---

**TOOLS.MACRO**

The TOOLS.MACRO menu has two sub-menu selections: Single File and Sequence of Files. Both of these modes require a command file (the default extension is **.dcf**) composed of a sequence of *xdamp* commands (see Table 4 for an example). The appropriate command file equivalent is listed at the end of each command description. You can incorporate comments into your command files by placing a semicolon at the beginning of the line. These comments will be echoed in the message area of the *xdamp* window during execution. Whitespace lines are ignored and can be inserted to make the command file easier to read but do not end a file with whitespace or comments. The single file version of this command can be used to nest macros up to 10 deep. The sequence of files macro is not callable in a macro.

---

**TABLE 4. Sample Macro to calculate power-weighted mean voltage.**

---

```

; Macro to calculate power and power-weighted mean voltage
; from synchronized load voltage waveform VL and load current IL
; Always start with CLEAR.
CLEAR
; Copy voltage to a new waveform named Power
WAVEFORM.VL
EDIT.COPY
Power
; Now enter the Power waveform prior to multiplying by current.
CLEAR
WAVEFORM.Power
ENTER
WAVEFORM.IL
MULTIPLY
; Now fix up units of new waveform for future reference.
CLEAR
WAVEFORM.Power
UNITS
seconds
MW
; Copy Power to a new waveform to make the power-weighted mean
CLEAR
WAVEFORM.Power
EDIT.COPY
PWMV
;Enter the new waveform and multiply by VL to get V*V*I
CLEAR
WAVEFORM.PWMV
ENTER
WAVEFORM.VL
MULTIPLY
; Fix the units on this waveform too.

```

TABLE 4. Sample Macro to calculate power-weighted mean voltage.

---

```

CLEAR
WAVEFORM.PWMV
UNITS
seconds
kV*MW
; Manually truncate the PWMV and Power waveforms to the region of interest
CLEAR
WAVEFORM.PWMV
WAVEFORM.Power
CURSOR ON
WINDOW
; Integrate the Power to get energy
CLEAR
WAVEFORM.Power
INTEGRATE
; Fix units of integrated power waveform INT_Power
UNITS
seconds
MJoules
; Store maximum energy in register 0 for later use.
EDIT.STORE
#MAX
0
; Divide the PWMV waveform by the energy in register 0
CLEAR
WAVEFORM.PWMV
DIVIDE
#R0
; Fix the units of PWMV
UNITS
seconds
kV/s
; Plot the final result - the integral will give the averaged PWMV
INTEGRATE
UNITS
seconds
kV
EDIT.STORE
#MIN
0
; Register zero now has PWMV in volts

```

**TOOLS.MACRO.Single File**

The single-file mode will pop the file selection widget to select an *xdamp* command file. This command file will then be executed and you will be returned to *xdamp*. The *xdamp.ini* file is not executed and the waveform/image buttons are left as is. Thus, you

can select a waveform/image and then perform a calculation on that waveform. To add a macro within another macro, use the following command syntax.

**Command Equivalent**

TOOLS.MACRO.SINGLE FILE

<full file name>

**TOOLS.MACRO.Sequence of Files**

The sequence-of-files selection pops the file selection widget three times. The first time, you are asked to pick the beginning data file of a sequence, the second time you pick the final data file, and the third time you choose the command file to operate on each of these files. Each file is automatically opened and closed, so you do not need to perform these operations in the command file. However, you must perform an explicit FILE.SAVE operation in the command file or the modifications to the data will not be permanently saved. The *xdamp.ini* file will be automatically executed prior to opening each file so that a known initial state is present. This option allows an easy way to translate a group of files to HDF format. Simply create a one line command file containing the command FILE.SAVE. Modify your *xdamp.ini* to reflect the data format and extension that you will be translating from. When you perform automatic multiple file execution, the original data files will be overwritten with HDF data files.

You need to be very careful of the state of *xdamp* when you begin an automatic execution. *xdamp* does not clear all of the waveform/image selection buttons and does not initialize many variables in single file execution mode. However, the initialization file is executed for the initial command file of a nested set and for each pass through a sequence of files loop. If you must assume a particular waveform/image selection state, it is wise to perform a CLEAR operation in your command file and select the desired waveforms prior to executing the operation.

**TOOLS.SHOW REGISTERS**

The TOOLS.SHOW REGISTERS menu selection pops a widget that shows the name and contents of each register. No action is taken and the widget remains visible until you press the DONE button. This selection has the command equivalent shown below but you must manually press the DONE button to continue with automatic execution.

**Command Equivalent**

TOOLS.SHOW REGISTERS

**TOOLS.SHOW LIMITS**

The TOOLS.SHOW LIMITS menu selection pops a widget that displays the current limits on the number and length of waveforms and the number and size of images. This selection has the command equivalent shown below but you must manually press the DONE button to continue with automatic execution. To change the limits, you must exit *xdamp* and edit your initialization file to change the MAXARRAYS= and/or MAX-SIZE= commands to the new limits.

**Command Equivalent**

TOOLS.SHOW LIMITS

**TOOLS.SHOW NOTES**

The TOOLS.SHOW NOTES menu selection pops a widget that shows the name and contents of each note associated with a waveform. No action is taken and the widget remains visible until you press the DONE button. You cannot edit the notes with this selection, instead use EDIT.NOTES. This selection has the command equivalent shown

below but you must manually press the DONE button to continue with automatic execution.

**Command Equivalent**

---

TOOLS.SHOW NOTES

**TOOLS.SHOW FILE NOTES**

The TOOLS.SHOW FILE NOTES menu selection pops a widget that shows the file notes. No action is taken and the widget remains visible until you press the DONE button. You cannot edit the file notes with this selection, instead use EDIT.FILE NOTES. This selection has the command equivalent shown below but you must manually press the DONE button to continue with automatic execution.

**Command Equivalent**

---

TOOLS.SHOW FILE NOTES

**TOOLS.SHOW HEADER**

The TOOLS.SHOW HEADER menu selection pops a widget that shows the file header information. No action is taken and the widget remains visible until you press the DONE button. You cannot edit the header information with this selection, instead use EDIT.HEADER. This selection has the command equivalent shown below but you must manually press the DONE button to continue with automatic execution.

**Command Equivalent**

---

TOOLS.SHOW HEADER

**TOOLS.XFONT**

The TOOLS.XFONT menu selection pops up the xfont selector widget on Unix systems. This widget allows you to preview and select a hardware font from all of those available to your terminal. This font becomes the graphics area font. This application has no command equivalent but can be placed in the **xdamp.ini** file.

**TOOLS.COLOR TABLE**

The TOOLS.COLOR TABLE menu selection pops up a widget used to load any of the standard color tables. Use this application to change to reverse video mode in the graphics area by using the options section and selecting reverse color table. This selection has the command equivalent shown below but you must manually press the DONE button to continue with automatic execution.

**Command Equivalent**

---

TOOLS.COLOR TABLE

**TOOLS.PALETTE**

The TOOLS.PALETTE menu selection pops up a widget used to select color palettes. Thus, you can change a color table by modifying the specific colors. This selection has no command equivalent.

---

## *Help Window*

---

Pressing the ? button will activate a sub menu allowing access to either the *xdamp* help or the IDL help widget. This latter choice is equivalent to typing a question mark at the IDL prompt. By pressing the *xdamp* button in the widget, an information window pops telling you to open the html file containing the *xdamp* help with its full location using your preferred browser. The ? button is ignored in automatic execution mode.

This method allows the help window to co-exist with the *xdamp* window rather than blocking *xdamp* until the help window is dismissed.

THIS PAGE INTENTIONALLY LEFT BLANK

## Operators

### OPERATOR OVERLOADING

There are two types of operators in *xdamp*; single-waveform/image operators and multiple-waveform/image operators. As a consequence, the behavior of the operators is slightly different.

For single-waveform/image operators (i.e. ABS, SHIFT, BASELINE), if multiple waveforms/images are selected, the same operation is performed on all of the selected waveforms/images.

For multiple-waveform/image operators, if multiple waveforms/images are selected, then a constant to operate on all the selected waveforms/images is requested. If you wish to perform a multiple waveform/image operation on two waveforms/images (i.e. add two waveforms/images), then the first waveform/image must be selected and then ENTERed.

If no waveforms/images at all have been selected, then it is assumed that you wish to perform register arithmetic on the pre-defined registers #R0 through #R9 for the following operators: ADD, SUBTRACT, MULTIPLY, DIVIDE, ABS, INVERSE, POWER, SQRT, LOG, LN, EXP, 10<sup>^</sup>.

Thus, many of the operators are overloaded and operate on registers, single waveforms, single images, multiple waveforms in two possible modes, or multiple waveforms. Addition of waveforms to images is not supported.

TABLE 5. Automatic Register Names

Register Name	Quantity
#NX	Number of points in waveform or x points in image
#NY	Number of y points in image
#DX	$\Delta t$ for waveform or dx for image
#DY	dy for image
#XMIN	Minimum time of waveform or minimum x of image
#XMAX	Maximum time of waveform or maximum x of image
#YMIN	Minimum y of image
#YMAX	Maximum y of image
#MIN	Minimum of waveform
#MAX	Maximum of waveform
#PEAK	Peak value of waveform/image (sign sensitive)
#XATMIN	Time or x-value at minimum
#XATMAX	Time or x-value at maximum
#XATPEAK	Time or x-value at peak
#YATMIN	Y-value at minimum
#YATMAX	Y-value at maximum
#YATPEAK	Y-value at peak

TABLE 5. Automatic Register Names

Register Name	Quantity
#FWHM	Full-width-at-half-maximum of waveform
#RISE	10%-90% rise time of waveform
#FALL	90%-10% fall time of waveform
#XSHIFT	Time shift from COMPARE
#YATX	Y value from Y at X command
#XATY1	First X value from X at Y command
#XATY2	Second X value from X at Y command
#XATY3	Third X value from X at Y command
#XATY4	Fourth X value from X at Y command
#SCALE	Scale factor from COMPARE
#OFFSET	Offset from COMPARE
#CURSORX	Time or x value from last Cursor operation
#CURSORX2	Second time or x value from H BARS or V BARS
#CURSORDX	Delta time or x from last H BARS or V BARS command
#CURSORY	y value of last Cursor operation
#CURSORY2	Second y value from H BARS or VBARS command
#CURSORDY	Delta y from last H BARS or V BARS command
#CURSORZ	z value of last Cursor operation
#INTERCEPT	Intercept from Calorimeter operation
#R0 - #R9	User number registers

When entering a constant to operate on single or multiple waveforms/images there are two possibilities. You can enter an explicit number in free format, or you can use one of the registers (assuming it is loaded). The registers are all prefixed with the character #. A variety of registers are automatically loaded during each plot operation when the pulse parameters are calculated. These are shown in Table 5. *The values come from the last waveform/image selected and plotted.* The peak register will be the maximum if the absolute value of the maximum is greater than the absolute value of the minimum, otherwise it is the minimum value.

All of the single- and multiple-waveform/image operators have similar command file sequences (see Table 4 for an example). The single-waveform/image operators allow the selection of as many waveforms/images as desired and then use the command sequence below to execute the command "AARDVARK".

#### Command Equivalent

AARDVARK

If the operator requires a constant input, it is provided in the following manner on the next line of the command file. If performing a cursor operation and the cursor preference is on, the command file temporarily becomes interactive. You may wish to dynamically prompt for a numeric input during automatic execution. To do this, replace the

<number or #register> in the following table with a question mark followed immediately by a prompt string without quotes. The prompt string will become the title of the input box and you will be prompted for the information to be loaded in the register.

**Command Equivalent**

<number> or <#register> or ?PROMPT STRING

Multiple-waveform/image operators all operate in one of two modes: operating on two waveforms/images or operating on many waveforms/images with the same constant. When a first waveform/image is selected, followed by an ENTER, and then a second waveform/image is selected, followed by pressing an operator button, the second waveform/image is added to the first. The first waveform/image is overwritten by the operation on both waveforms/images after truncating the first waveform/image to the common time domain and resampling the second waveform/image if necessary. The second waveform/image is not modified. The command equivalent form of multiple-waveform/image operations for the ADD routine is shown next, including the waveform/image selection portion, where test1 is the waveform/image to be modified and test2 is to be added to test1.

**Command Equivalent**

```
WAVEFORM.test1
ENTER
WAVEFORM.test2,
ADD
```

In the second mode of operation, if one or more waveforms/images are selected without an ENTER, then you are prompted for a constant to operate on all of the waveforms/images selected. All of them are modified by this process. The command equivalent will require a <variable value> line after the operator command to perform correctly unless the operator uses the cursor and cursor mode is on. In this situation, the program pauses for cursor movement so you can interact with *xdamp*. Below is the command sequence (notice the lack of an ENTER command) to add the constant 3.95 to several images.

**Command Equivalent**

```
IMAGE.test1
IMAGE.test2
ADD
3.95
```

**ADD**

ADD is a multiple-waveform/image operator that adds waveforms/images together, numbers to waveforms/images, or numbers to registers. This command modifies the ENTERed waveform/image or all of the selected waveforms/images.

**SUBTRACT**

SUBTRACT is a multiple-waveform/image operator that subtracts waveforms/images from each other, numbers from waveforms/images, or numbers from registers. This command modifies the ENTERed waveform/image or all of the selected waveforms/images.

**MULTIPLY**

MULTIPLY is a multiple-waveform/image operator that multiplies waveforms/images by each other, waveforms/images by a number, or registers by a number. This command modifies the ENTERed waveform/image or all of the selected waveforms/images.

**DIVIDE**

DIVIDE is a multiple-waveform/image operator that divides waveforms/images by each other, waveforms/images by a number, or registers by a number. This command modifies the ENTERed waveform/image or all of the selected waveforms/images. Zero values of the divisor are masked out so numerical errors are trapped.

**ABS**

ABS is a single-waveform/image operator that takes the absolute value of waveforms/images or registers. This command modifies all of the selected waveforms/images.

**AVERAGE**

AVERAGE is a multiple-waveform/image operator that averages all selected waveforms/images and places the result in a new waveform/image with the name of the first selected waveform/image prefixed by "AVG\_". This command does not modify any of the waveforms/images.

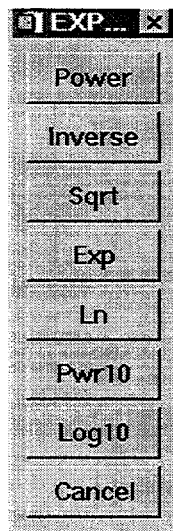
**EXPONENTS**

EXPONENTS pops a window (Fig. 5) to select among the various power, logarithm, and inverse logarithm single-waveform/image operators: POWER, INVERSE, SQRT, EXP, LN, 10<sup>^</sup>, LOG. These modify all of the selected waveforms/images or the register chosen. A typical command sequence is shown, assuming the waveforms/images have already been selected.

**Command Equivalent**


---

EXPONENTS  
POWER

**FIGURE 5. EXPONENTS subwindow****POWER**

POWER is a single-waveform/image operator that takes a waveform/image or register to an *integer* power. This command modifies all of the selected waveforms/images. For non-integer powers, use EXP(f\*LN(array)) where f is the non-integer power.

**INVERSE**

INVERSE is a single-waveform/image operator that takes the inverse (1/x) of waveforms/images or a register. This command modifies all of the selected waveforms/images.

- SQRT** SQRT is a single-waveform/image operator that takes the square root of waveforms/images or registers. This command modifies all of the selected waveforms/images.
- EXP** EXP is a single-waveform/image operator that takes  $e$  raised to the waveform/image or  $e$  to the register. This command modifies all of the selected waveforms/images.
- LN** LN is a single-waveform/image operator that takes the natural logarithm of the waveform/image or register. This command modifies all of the selected waveforms/images.
- PWR10** PWR10 is a single-waveform/image operator that takes 10 raised to the waveform/image or register. This command modifies all of the selected waveforms/images.
- LOG10** LOG10 is a single-waveform/image operator that takes the base-10 logarithm of the waveform/image or register. This command modifies all of the selected waveforms/images.
- UNITS** UNITS is a single-waveform/image operator that allows you to change all of the axis-units labels for one or more waveforms/images. This command modifies all of the selected waveforms/images. Its command equivalent requires two or three additional inputs depending whether you are in waveform or image mode
- Command Equivalent**
- 
- UNITS  
 <time or x units string>  
 <amplitude or y units string>  
 <amplitude units string> (image mode only)
- SCALE AXIS** SCALE AXIS is a single-waveform/image operator that allows you to change the axis scales for one or more waveforms/images. This command does not allow you to modify the vertical axis scale, use multiply or divide instead. This command modifies all of the selected waveforms/images. Its command equivalent requires one or two additional inputs depending whether you are in waveform or image mode
- Command Equivalent**
- 
- SCALE AXIS  
 <time scaling factor>  
 <y units scaling factor> (image mode only)
- CEILING** CEILING is a single-waveform/image operator that truncates the waveforms/images at values above those determined by the cursor or manual input. Cursor input does not work for image mode. This command modifies all of the selected waveforms/images.
- FLOOR** FLOOR is a single-waveform/image operator that truncates the waveforms/images at values below those determined by the cursor or manual input. Cursor input does not work for image mode. This command modifies all of the selected waveforms/images.
- WINDOW** WINDOW is a single-waveform/image operator that truncates the waveforms or images within a window defined by a box cursor or a series of numbers, depending on the cursor preference setting. This command modifies all of the selected waveforms.

**SHIFT**

SHIFT is a single-waveform/image operator that shifts the waveforms or sets the zero coordinates of an image according to the cursor or an entered value. For cursor inputs, the cursor location becomes the new location of time zero for waveforms or the x-y axis zero coordinates for images. For entered inputs the values are added to the starting time of the waveform or the current coordinate system. This command modifies all of the selected waveforms. In command mode, one or two inputs will be required depending on the mode.

**BASELINE**

BASELINE is a single-waveform/image operator that "baselines" the waveforms or images according to the cursor or an entered value. The cursor or entered input defines a time for waveforms or an area for images over which the integral of all the selected waveforms is set to zero. Thus, offset before the main pulse can be removed from the overall integral of waveforms or background noise set to zero for images. This command modifies all of the selected waveforms/images.

**COMPARE**

COMPARE is a multiple-waveform operator that compares two waveforms. This command *must* have the reference waveform selected and ENTERed. Then the variable waveform to be compared to the reference is selected and the COMPARE button is pressed. A copy of the variable waveform is used for all operations so the original is not modified. First, the two waveforms are resampled to the same time intervals. Then a cross-correlation operation is performed to find the optimum time shift to best time align the two waveforms and the variable waveform/image is time shifted and truncated to completely overlap the reference waveform. Finally, an IDL polynomial fitting routine is used to fit the equation:

$$\text{REFERENCE} = \text{SCALE} * \text{VARIABLE} + \text{OFFSET.} \quad (\text{EQ } 1)$$

Here, the scale is the multiplier that should be applied to the variable waveform to best fit it to the reference waveform and the offset is the constant that must be added to it. There are two possible modes for the COMPARE operation: BASELINE FIXED and BASELINE VARIABLE. If the fixed mode is chosen, then the offset is zero. The reference waveform and the best fit variable waveform are plotted with the scale, offset, time shift, standard deviation, and correlation coefficient of the fit shown on the left side of the graphics area. The fitting parameters are stored in registers that can be used as input for subsequent operations according to the following scheme.

**TABLE 6. Register Names and Contents from COMPARE Operation**

Register Name	Quantity
#SCALE	Scale factor
#OFFSET	Baseline offset
#XSHIFT	Time shift

After the compare operation is complete, you can press the PRINT button to generate a hardcopy or press the ZOOM button to zoom in on the compared plot (and print this if you desire).

**NORMALIZE**

NORMALIZE is a single-waveform/image operator that divides each selected waveform/image by the absolute value of its peak value thus maintaining the polarity. This

command modifies all of the selected images/waveforms which will then range from -1 to 1.

### INTEGRATE

INTEGRATE is a single-waveform/image operator that integrates the waveforms point by point using Simpson's rule. A new waveform is created for each one integrated if sufficient waveform memory is available; otherwise, the old waveforms are overwritten. These new waveforms are named the same as the original waveforms except the names are prefixed with "INT\_". This command does not modify any of the waveforms if sufficient memory exists. For images, two new waveforms are created, one being the x-integral and one being the y-integral. Integrating either of these waveforms gives the total integral over the image.

### DIFFERENTIATE

DIFFERENTIATE is a single-waveform operator that differentiates the waveforms point by point using a three-point central differencing technique. A new waveform is created for each one differentiated if sufficient waveform memory is available; otherwise, the old waveforms are overwritten. These new waveforms are named the same as the original waveforms except the names are prefixed with "DIF\_". This command does not modify any of the waveforms if sufficient memory exists. Differentiate is undefined for images but see the Filter commands for edge enhancement routines.

### TRANSFORMS

The TRANSFORMS button pops a window (Fig. 6) with all available transforms accessed by buttons. The window that is created depends on the mode, image or waveform, because different transforms are appropriate in the different modes. For waveforms the routines FFT, IFFT, CONVOLVE, and CROSS CORRELATE are provided. For images, the routines ROTATE, MIRROR (about 4 different axes), and UNWRAP are available. The command mode operation is a bit different for these operators because an additional level of widgets exists. The following sequence would work for FFT with similar sequences for the others.

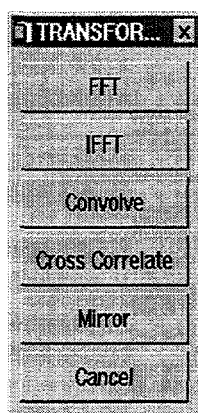
#### Command Equivalent

```

WAVEFORM.name
TRANSFORMS
FFT

```

FIGURE 6. Waveform TRANSFORMS subwindow



**FFT**

FFT is a single-waveform operator that takes the discrete Fourier transform of the selected waveforms. It returns complex waveforms, so for each waveform with name "DATA" three new waveforms ("RE\_FFT\_DATA", "IM\_FFT\_DATA", and "MAG\_FFT\_DATA") are created. The first waveform contains the real component, the second contains the imaginary component, and the third contains the magnitude of the FFT. Time-phase information is lost during this process because it cannot be easily removed from the inverse operation IFFT. Therefore, if you transform a waveform and then invert the transform you will recover the shape but the waveform will start at t=0. FFT results are displayed from the negative Nyquist frequency to the positive Nyquist frequency. If the original waveform has an even number of points, one point is removed in the FFT so that the dc component will fall at 0 Hz on the plot.

**IFFT**

IFFT is a multiple-waveform operator that takes the inverse discrete Fourier transform of the selected waveforms. The real component must be designated with the ENTER button, followed by selecting the imaginary component. This operation returns complex waveforms, so for each waveform "DATA" two new waveforms ("RE\_IFFT\_DATA" and "IM\_IFFT\_DATA") are created. The first waveform contains the real component and the second contains the imaginary component of the IFFT. Time-phase information is lost during this process. Therefore, if you transform a waveform and then invert the transform you will recover the shape but the waveform will start at t=0. A typical command sequence is shown below.

**Command Equivalent**


---

```
WAVEFORM.RE_Waveform
ENTER
WAVEFORM.IM_Waveform
TRANSFORMS
IFFT
```

**CONVOLVE**

CONVOLVE is a multiple-waveform operator that performs the mathematical convolution operation by serial addition (rather than an FFT/IFFT pair). The first waveform must be ENTERed even though the operation is commutative. A new waveform, containing the convolution of the two will be created with its name based on the ENTERed waveform name prefixed with "CONV\_". The new waveform is longer than either original waveform.

**CROSS CORRELATE**

CROSS CORRELATE is a multiple-waveform operator that performs a cross correlation by serial addition (rather than an FFT/IFFT pair). The first waveform must be ENTERed because this operation is not commutative. A new waveform, containing the cross correlation of the two will be created with its name based on the ENTERed waveform name prefixed with "XCOR\_". The new waveform is longer than either original waveform.

**MIRROR**

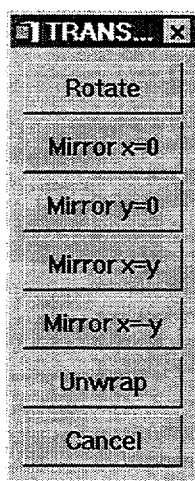
MIRROR is a single-waveform operator that reverses the waveforms about zero time. This command modifies all of the selected waveforms

**Command Equivalent**


---

```
TRANSFORMS
MIRROR
```

FIGURE 7. Image TRANSFORMS window

**ROTATE**

ROTATE is a single-image operator that will rotate an image about the point defined by the  $x=0$ ,  $y=0$  by an arbitrary angle. You are prompted for the angle to rotate the image by in a *clockwise* sense. The image is modified and undefined areas of the rotated image are filled with zeros.

**Command Equivalent**


---

 TRANSFORMS

ROTATE

&lt;CW rotation angle&gt;

**MIRROR X=0**  
**MIRROR Y=0**  
**MIRROR X=Y**  
**MIRROR X=-Y**

MIRROR X=0, MIRROR Y=0, MIRROR X=Y, and MIRROR X=-Y are single-image operators that reflect the image about the  $x=0$  (y-axis),  $y=0$  (x-axis),  $x=y$  diagonal, or  $x=-y$  diagonal, respectively. The image is modified.

**Command Equivalent**


---

 TRANSFORMS

MIRROR X=0 or MIRROR Y=0 or MIRROR X=Y or MIRROR X=-Y

**UNWRAP**

UNWRAP is a single-image operator that assumes that the digitized image was originally wrapped around a cylinder with axis along the y-axis of the cylinder. You are prompted for the cylinder radius. UNWRAP then corrects the x-axis units to account for unwrapping the film to digitize it. The image is modified.

**Command Equivalent**


---

 TRANSFORMS

UNWRAP

&lt;cylinder radius&gt;

**FILTERS**

As with TRANSFORMS, the FILTERS operation provides mode sensitive sub-windows. In waveform mode, the waveform FILTERS window appears while in image mode, the image FILTERS window appears.

FIGURE 8. Waveform FILTERS input window

**FILTER**

DO IT CANCEL

Number of Coefficients 50

Gibbs Supression (dB) -50.0000

LOW PASS  HIGH PASS  BAND PASS  NOTCH

Nyquist Frequency = 1.27750e+010 Hz

0.000000

LOWER FREQUENCY

1.27750E+010

UPPER FREQUENCY

The waveform FILTERS operator is a multiple-waveform operator that performs low-pass, high-pass, band-pass and notch filtering. The maximum filtering frequency depends on the waveform with the largest sampling interval of those selected. The FILTERS button pops a widget (Fig. 8) to obtain information about the desired filter and the filter frequencies desired. The original waveforms are overwritten. The filter algorithm uses a non-recursive digital filter with 50 coefficients and a 50dB Gibbs peak roll off obtained from a Kaiser windowing function. The number of coefficients and the Gibbs roll off are variable. For notch filters in the command equivalent, simply set  $f_{\text{lower}}$  greater than  $f_{\text{upper}}$ . These frequencies are in Hz in the command file. The Gibbs suppression number controls a Kaiser window. For very steep filters, set this number near 0dB. Register inputs are allowed in the command mode.

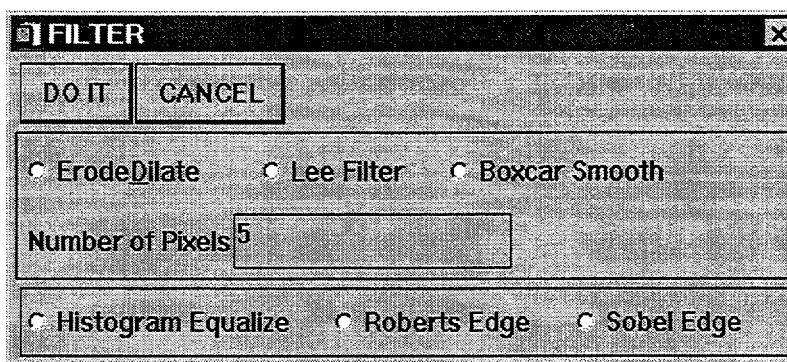
#### Command Equivalent

```

FILTERS,
<lower frequency (Hz)>
<upper frequency (Hz)>
<# Coefficients>
<Gibbs roll off (dB)>

```

FIGURE 9. Image FILTERS Window

**ERODE&DILATE**

The ERODE&DILATE image filter performs erosion followed by dilation back to the original image size. The number of pixels is used to create a square operator. The net effect of this filter is to remove all structure in the image smaller than the size of the square operator. Thus, if a 5 pixel argument is provided, all features smaller than 5x5 pixels will be lost. The image is modified.

**Command Equivalent**

```
FILTERS
ERODE&DILATE
<number of pixels>
```

**LEE FILTER**

The LEE FILTER performs the Lee filter algorithm on the image using a box of size NxN pixels where N is odd. If an even N is input, then the next smaller odd number will be used. The image is modified. The Lee filter algorithm (see IDL Reference Guide) smooths additive image noise by generating statistics in a local neighborhood and comparing them to the expected values.

**Command Equivalent**

```
FILTERS
LEE FILTER
<number of pixels>
```

**BOXCAR SMOOTH**

The BOXCAR SMOOTH filter performs simple boxcar smoothing on the image using a box of size NxN pixels (see IDL Reference Guide). If an even N is input, then the next larger odd number will be used. The image is modified.

**Command Equivalent**

```
FILTERS
BOXCAR SMOOTH
<number of pixels>
```

**HISTOGRAM EQUALIZE**

The HISTOGRAM EQUALIZE operator performs histogram equalization of the image (see IDL Reference Guide). The resulting image will maintain the existing peak value.

**Command Equivalent**

```
FILTERS
HISTOGRAM EQUALIZE
```

**ROBERTS EDGE**

The ROBERTS EDGE filter performs Roberts edge enhancement, a linear edge sharpening and isolation algorithm (see IDL Reference Guide), on the image.

**Command Equivalent**

FILTERS  
ROBERTS EDGE

**SOBEL EDGE**

The SOBEL EDGE filter performs Sobel edge enhancement using a nonlinear edge enhancement operator (see IDL Reference Guide).

**Command Equivalent**

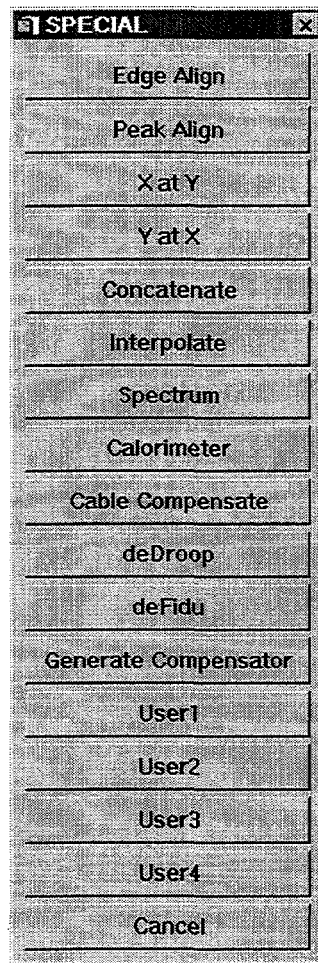
FILTERS  
SOBEL EDGE

**SPECIAL**

Many additional operations that are infrequently used or did not fit under any other category are grouped under the SPECIAL button. For instance, waveform cable compensation and various image profiling procedures are. The command mode operation is a bit different for these operators because an additional level of widgets exists. The following sequence would work for MIRROR with similar sequences for the others.

**Command Equivalent**

```
WAVEFORM.name
SPECIAL
MIRROR
```

**FIGURE 10. Waveform SPECIAL selection window****EDGE ALIGN**

EDGE ALIGN is a multiple-waveform/ operator that time aligns all waveforms with the ENTERed waveform/image based on an input fraction of the peak value of each waveform. The user is prompted for this fraction so in automatic execution mode, a fraction between -1 and 1 is needed. For positive fractions, the alignment is done according to the peak value of the waveform. For negative fractions, the alignment uses the opposite

polarity compared to the peak value of the waveforms. The numerical argument may be a register.

**Command Equivalent**

SPECIAL  
EDGE ALIGN  
<number>

**PEAK ALIGN**

PEAK ALIGN is a multiple-waveform/ operator that time aligns the selected waveforms using the cross-correlation algorithm of the COMPARE routine. This command modifies all of the selected waveforms except the ENTERed reference waveform.

**Command Equivalent**

SPECIAL  
PEAK ALIGN

**Y at X**

The Y at X command is a single waveform operator that places the value of the last waveform selected at the x value entered from the keyboard (the cursor cannot be used). This command does not operate on images. It loads the #YATX register for further manipulations.

**Command Equivalent**

SPECIAL  
Y at X  
<x value>

**X at Y**

The X at Y command is a single waveform operator that places up to 4 x values corresponding to the input y value entered at the keyboard (the cursor cannot be used). It operates only on the last waveform selected and not on images. The registers #XATY1, #XATY2, #XATY3, and #XATY4 are loaded with the crossing values. If no crossing is found, -1e36 is placed in the registers.

**Command Equivalent**

SPECIAL  
X atY  
<y value>

**CONCATENATE**

CONCATENATE is a dual-waveform operator to paste two waveforms together. A time is defined using either the cursor or a numeric time input, depending on the cursor preference. The late time portion of the second waveform is appended to the early-time portion of the entered waveform. The entered waveform is overwritten.

**Command Equivalent**

SPECIAL  
CONCATENATE

**INTERPOLATE**

INTERPOLATE is a single-waveform operator that allows you to change a section of a waveform to the values linearly interpolated between two defined times. In cursor mode, a box to define the times appears and in non-cursor mode, you are prompted for the two times. The box may be moved by pressing and holding the left mouse button, resized by pressing and holding the middle mouse button, and exited by pressing the right mouse button. (Mac users will need to use special keys with the single mouse key to perform these acts - see the ZOOM command on page 66.) This command modifies all of the

selected waveforms. In command mode, the two times must be entered on separate lines and may contain register references.

**Command Equivalent**

SPECIAL  
INTERPOLATE  
<tmin>  
<tmax>

## SPECTRUM

SPECTRUM is a dual-waveform operator to calculate the electron spectrum given the voltage and current waveforms in a diode. You must select the voltage waveform and ENTER it followed by the current waveform. The procedure assumes that these two waveforms are properly time aligned. Then you are prompted for the number of spectral bins and the desired maximum energy. If the voltage or current are negative polarity, they are converted to positive polarity so the maximum spectral energy is always positive. SPECTRUM creates two new pseudo-waveforms, ESPECTRUM and CUMSPECTRUM. ESPECTRUM contains the differential electron number spectrum, normalized to an integral of one electron. The CUMSPECTRUM pseudo-waveform contains the cumulative number spectrum integral of ESPECTRUM, normalized to unity. The standard INTEGRATE command is not proper because of some tricks used in the binning of the electron spectrum to make the ESPECTRUM histogram look nice. The bottom bin begins at  $\Delta E_{\text{bin}}/2$  and will have the number of particles with energies between 0 and  $\Delta E_{\text{bin}}$ . All other bin widths are  $\Delta E_{\text{bin}}$ . The peak bin ends at the maximum energy plus  $\Delta E_{\text{bin}}/2$ . For CUMSPECTRUM, the bottom bin begins at 0.0 and the top bin ends at the maximum energy. Note that any waveform with SPECTRUM in the name will be plotted as a histogram. The original two waveforms are not altered by this process. If you repeat the operation, the existing ESPECTRUM and CUMSPECTRUM waveforms will be overwritten unless you rename them.

**Command Equivalent**

WAVEFORM.V  
WAVEFORM.I  
SPECIAL  
SPECTRUM  
<nbins>  
<maximum energy>

## CALORIMETER

CALORIMETER is a single-waveform operator that calculates the value of the selected waveform at the time  $t=0$  by linear extrapolation. The peak of the waveform is used as one point and the second is determined by the cursor or a delta time can be entered in an input box if the cursor preference is off.

**Command Equivalent**

SPECIAL  
CALORIMETER

## CABLE COMPENSATE

CABLE COMPENSATE is a multiple-waveform operator that performs the mathematical convolution operation by serial addition of a response waveform with a cable compensator waveform. The first waveform must be ENTERed and then the cable compensator selected. The original waveform is overwritten and is not lengthened.

**DEDROOP**

DEDROOP is a single-waveform operator that removes an RC type roll off from a set of waveforms. This is done by serial convolution with a correction waveform. The original waveforms are overwritten.

**Command Equivalent**

SPECIAL  
DEDROOP  
<RC time in s>

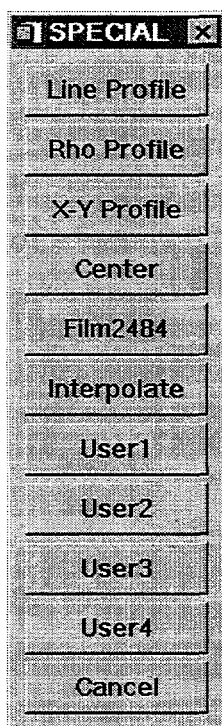
**DEFIDU**

DEFIDU is a single-waveform operator that removes a positive fiducial marker from a set of waveforms. If the cursor preference is on, a cursor appears to define the portion of the waveform containing the fiducial marker. Otherwise, it is assumed that the fiducial marker is within the first 10% of the waveform. The peak of the fiducial is assumed to define time zero and the waveform is deleted up to the point at which the fiducial drops to 10% of its peak plus three points. The original waveforms are overwritten.

**GENERATE  
COMPENSATOR**

GENERATE COMPENSATOR is a multiple-waveform operator that performs the mathematical deconvolution operation to find the cable compensator waveform. The reference waveform (directly into the digitizer) must be ENTERed and then data with the cable in place is selected. Next you are prompted for the truncation fraction (0.5 is a good start) and the desired length of the compensator array. The desired length defaults to a multiple of 500 but may need to be a power of two for some digitizers. This operation performs the serial deconvolution operation described in Boyer's report SAND87-3072. The cable compensator is given the name of the second waveform prefixed by "COMP\_". The reference data, cable data, and compensated cable data are then plotted. If you repeat the operation with a different truncation fraction, then the compensator waveform is overwritten.

FIGURE 11. Image SPECIAL selection window

**LINE PROFILE**

LINE PROFILE uses a line cursor or four inputs to define a line in the image. A new waveform is created containing the profile of the image along this line.

**RHO PROFILE**

RHO PROFILE performs averaging of line profiles along a number of radii of a circle. You can select the number of averages and the starting and ending angles. With the cursor preference on, RHO PROFILE uses an input box to define the starting angle, ending angle, and number of radii followed by three cursor points to define a circle. With the cursor preference off, you are prompted for the coordinates of the center of the circle, the radius of the circle, the starting and ending angles, and the number of radii to average over. Profiles are generated along the radii and averaged together to create a new waveform named <RHO>\_PROFILE containing an averaged profile. The command equivalent with cursor preference off is shown below.

**Command Equivalent**

```

SPECIAL
RHO PROFILE
<x coordinate of center>
<y coordinate of center>
<circle radius>
<start angle in degrees>
<end angle in degrees>
<number of radii>

```

**X-Y PROFILE**

X-Y PROFILE uses the box cursor or four inputs to define a sub area of the image to create both x and y profiles for. Two new waveforms are created that have the average profile along the x and y directions.

**CENTER**

The CENTER algorithm allows you to define three points on the circumference of a circle using the cursor. The center of this circle is then redefined as the center of the image. This command requires the use of the cursor to define the circle.

**FILM2484**

The FILM2484 algorithm converts optical density (OD) for an image to energy fluence (F in ergs/cm<sup>2</sup>) assuming that type 2484 film has been used. The correction algorithm is  $F = 0.0002 * \exp(0.3687 * OD)$ .

**Command Equivalent**

SPECIAL  
FILM2484

**USER1  
USER2  
USER3  
USER4**

USER1 through USER4 are distributed as template procedures that every user can modify easily to create their own operators. USER1 and USER2 are designed as single waveform operators while USER3 and USER4 are designed as dual waveform operators. To create your own procedures, copy the **XI\_user.pro** file from the *xdamp* library location to your local directory. Edit the file appropriately, but do not change the names of the operators from USERN to something else. When you run *xdamp*, your own routine will be compiled and executed when these buttons are pushed. The distribution file has sufficient boiler plate information to show how to construct an operator. Knowledge of IDL programming language is assumed.

## *Speed Buttons*

---

A series of speed buttons are placed immediately above the waveform button area and just below the operator area. These are for the most commonly used tasks and some of the buttons are also available from the menus.

### **WAVEFORM/IMAGE**

The WAVEFORM and IMAGE toggle buttons serve as another way to change display modes. The button shows the current status. Pressing it changes to the other mode. In command mode, WAVEFORM will change to the waveform view and plot any selected waveforms. Similarly IMAGE will change to the image view and plot any selected waveforms. The command IWTOGGLE will switch you from whatever mode you are in to the other mode.

#### **Command Equivalent**

WAVEFORM	forces WAVEFORM mode
IMAGE	forces IMAGE mode
IWTOGGLE	switches to other mode

### **ENTER**

The ENTER button is used to designate one waveform as special. For operations adding, subtracting, multiplying, or dividing two waveforms, the ENTER button designates the waveform to be overwritten. It is best thought of as a reverse Polish notation (RPN) style ENTER button. For operations that use one waveform as a reference (i.e. COMPARE or cable compensator creation) the ENTER button designates the reference waveform.

### **CLEAR**

The CLEAR button deselects all waveforms, clears the plot area, and clears the message box.

#### **Command Equivalent**

CLEAR

### **SELECT**

The SELECT button selects waveforms based on an input string containing wildcard characters (\*) in the specification. For instance V\* would select all waveforms beginning with the capital letter V. The string specification is case sensitive.

#### **Command Equivalent**

SELECT  
<wildcard selection string>

### **ALL**

The ALL button selects all of the waveforms.

#### **Command Equivalent**

ALL

### **PRINT SCREEN**

The PRINT SCREEN button causes whatever is in the current plot window to be placed in a print file. This is also accessible via the PRINT menu as PRINT.SCREEN. The print file is not spooled until you exit the program or force a spool through the PRINT menu.

#### **Command Equivalent**

PRINT SCREEN

**PLOT**

The PLOT button causes all waveforms selected to be immediately plotted. If no waveforms have been selected, then all the waveforms are plotted sequentially with a two second delay.

**Command Equivalent**

PLOT

**ZOOM**

The ZOOM button replots the selected waveforms and places a zoom box on the screen if the cursor preference is on. The zoom box may be moved by pressing and holding the left mouse button, resized by pressing and holding the middle mouse button, and exited by pressing the right mouse button. (Mac users will need to use special keys with the single mouse key to perform these acts and Windows users must have a three-button mouse.) The zoomed image is then plotted. If the cursor preference is off, you are prompted for the four limits of the zoom plot. In the command equivalent, the last four inputs are only needed if you are not in cursor mode. You may use register inputs in this case. Note that older versions of *xdamp* placed all four numbers on one line, separated by commas. Now each number must be on a separate line.

**Command Equivalent**

ZOOM

<tmin or xmin in data units>  
<tmax or xmax in data units>  
<ymin in data units>  
<ymax in data units>

**CURSOR**

The CURSOR button allow you to move the cursor on the screen and observe values in the message window. When you click on the left mouse, the current cursor values are loaded in the registers CURSORX, CURSORY and CURSORZ if you are in image mode. You may then perform register operations using these values. This command ignores the cursor preference and always places a cursor on the screen, even in automatic execution mode where it is effectively interactive.

**Command Equivalent**

CURSOR

**H BARS**

The H BARS button creates two horizontal cursor bars for use in loading registers with measurement data from waveforms. The cursors are moved as a pair with the left mouse button, individually with the middle mouse button and the right mouse button ends movement. The x,y values based on the last waveform plotted for the cursors are loaded into the registers #CURSORX, #CURSORY, #CURSORX2, and #CURSORY2, while the delta values are placed in #CURSORDX and #CURSORDY.

**Command Equivalent**

H BARS

**V BARS**

The V BARS button creates two vertical cursor bars for use in loading registers with measurement data from waveforms. The cursors are moved as a pair with the left mouse button, individually with the middle mouse button and the right mouse button ends movement. The x,y values based on the last waveform plotted for the cursors are loaded

into the registers #CURSORX, #CURSORY, #CURSORX2, and #CURSORY2, while the delta values are placed in #CURSORDX and #CURSORDY.

**Command Equivalent**

V BARS

THIS PAGE INTENTIONALLY LEFT BLANK

---

## *Waveform/Image Buttons*

---

The names of the waveform/image buttons vary with the application. The technique used to select any waveform is identical; simply push the button. To deselect a waveform, simply push the button again. The command syntax is shown below for a waveform named Example. Note that the names are case sensitive.

### **Command Equivalent**

---

WAVEFORM.Example

To select an image, simply use a similar command with IMAGE replacing WAVEFORM.

If too many waveforms exist to fit in the waveform button box, then the box becomes a scrolling region and you may need to use the scroll bars to find the waveforms to select.

Beware of two things. The waveform selection process is a toggle that may cause problems in automatic execution mode if a waveform is selected for a second time and becomes deselected. Also, waveform names are case sensitive.

In command mode, three methods are available for changing modes. First, you can simply select an image or waveform to switch modes. Second you can use WAVEFORM or IMAGE to force the particular mode. Third, you can use IWTOGGLE to change modes.

THIS PAGE INTENTIONALLY LEFT BLANK

---

## *Problems and Troubleshooting*

---

If you experience a problem, *first* look in the message area for an informational message about anything that may be illegal such as performing an operation that requires an ENTER or opening a file with the incorrect format. Next, look in the IDL command window area for error messages that might indicate problems with memory or a corrupted *xdamp* module.

If you can't get *xdamp* to start properly on a DOS computer, be certain that the file IDL-SPAWN.PIF has been moved to the windows subdirectory and edited to be exclusive. (This is not necessary on Windows 95 or Windows NT computers.)

If your display does not appear similar to Figure 1, "xdamp screen layout in waveform mode," on page 17, then you probably need to change the widget font size in your *.Xdefaults* file (See "GETTING STARTED" on page 13.) and then reboot on a Unix system. PC users can use the `Wfont=SYSTEM*12` (for instance) command in the *xdamp.ini* file and Macintosh computers can adjust this font size in the IDL preferences menus. If this doesn't work, reduce the size of the graphics font area according to your display's limitations by using the GRAPHIC AREA command in the *xdamp.ini* file or in the OPTIONS screen.

If you have trouble when opening a file, you may need to edit the *XIsys\_sp.pro* file and increase the maximum length of allowable waveforms or create an *xdamp.ini* file that modifies these limits. For efficiency, be sure that this length is not appreciably longer than necessary for your longest record. You can use `TOOLS/LIMITS` to see what the current values are.

If *xdamp* fails when you first try to plot a waveform, then you probably have an illegal font name in your *xdamp.ini* file. Remove the `GFONT=font_string` line from this file and try *xdamp* again.

Difficulties with the Unix file selection widget are usually due to rapid multiple clicking on selections. This widget wants *single* clicks (on Unix systems) and is not particularly fast because it must spawn system commands to obtain the directory information. Be patient. The file selection widget is also case sensitive.

A common difficulty is not recognizing that waveform names *are* case sensitive when defining a select string with wild cards. Similarly, the file filter definition must contain an asterisk like `*.hdf` and the file names are also case sensitive. *Without a wild card, nothing will be selected.*

The single file automatic execution mode does not initialize many variables (for maximum flexibility). Therefore, you should start your macro file assuming that the waveform selection state is ill-defined. However, multiple file automatic execution re-initializes based on your *xdamp.ini* file for each new file opened.

When using automatically loaded registers, remember that the pulse parameter values come from the last waveform selected, not the first waveform selected. When in doubt,

just plot one waveform at a time. The pulse parameters are generated whether or not you elect to have the results displayed on the screen.

The xfont widget application is not very robust. Frequently, it will have conflicts if another application (FrameMaker) is using display PostScript. It is best to set the font by using this widget *immediately* after starting only IDL, reading the font in the message window and placing this in the `xdamp.ini` file with the `GFONT=font_string` command.

Color table problems can occur in this application. A specific symptom is if the zoom box is not visible. Other applications and IDL can use the same color map, not always with benign interactions. For Unix systems, use the `Idl.colors: 64` command in your `.Xdefaults` file to reserve some colors to IDL. PC and Mac systems seem to operate best if only 256 colors are enabled for the display. For other systems, the order in which applications are started can affect what happens.

If you have insufficient memory failures while deleting or keeping waveforms, edit the `XI_edit.pro` file and find the `XIedit_delete` section. There is a comment with a note to uncomment one line and comment out another line to improve the memory management at the expense of speed. Do the same thing with the `XIedit_keep` section. This will slow the delete and keep commands but generally prevents insufficient memory failures.

In automatic execution mode, if you are automatically clearing the stack after each operation and are using a journal file generated with this setting disabled, very strange things can occur. Many of the operators are overloaded and think that trying to add without any waveform selected means that you want to add a constant to a register. When there is no valid information for the register arithmetic, the whole automatic execution will fail. This cannot be trapped easily so you must generate command files with the stack clearing preference you will use when executing them.

Do not end a macro file with a comment or blank line. `xdamp` thinks another command follows this and fails when it reaches the end of file unexpectedly.

Do not use `FILE.GET DATA` to append files with only a single array. Instead use the `FILE.APPEND` which is faster and requires less input from the user.

---

## *Macintosh Specific Issues*

---

If your display does not appear like Figure 1, "xdamp screen layout in waveform mode," on page 17, then you probably need to reduce the widget font size via the preferences section. If this doesn't work, then reduce the size of the graphics font area in the file **XIsys\_sp.pro** or with the GRAPHIC AREA command in the **xdamp.ini** file according to your display's limitations.

Do not use the close window box to complete data entries. Use the DONE button in the widget instead. Otherwise, IDL will not receive an event to process the input stream.

Macintosh output cannot be automatically spooled. You must print the **xdamp.out** file yourself after performing a PRINT.QUEUE command. This can be done using the shareware program DropPs available from [mac.archive.umich.edu](http://mac.archive.umich.edu) via anonymous ftp.

Macintosh journal file output will appear on a single line if you are using the Power PC version. Using an editor, change all the linefeeds (ASCII 10) to carriage returns and continue to edit the file normally. This is a known IDL bug.

To use the ZOOM, you need three mouse buttons. The left button is the normal Mac button, the center button is the Option key with the mouse button and the right button is the Apple key with the mouse button.

The Xfont application does not function for Macintosh computers. Instead, use the preferences section of the Macintosh menu bar to set the desired font for the application.

On a Macintosh, DO NOT use the FILE.QUIT on the screen menu bar because this will not properly close the journal file.

---

## *Windows Specific Issues*

---

After some operations, you sometimes wind up back at the main IDL window instead of at the *xdamp* window. Use CONTROL-TAB to switch back to the desired window.

If you don't have a PostScript printer and want to print the **manual.ps** file, download the Ghostscript, Ghostview, GSview freeware from:

<http://www.cs.wisc.edu/~ghost/index.html>

and install it on your system. Alternatively, you can use Adobe Acrobat Reader 3.0 to read the **manual.pdf** file. The free Acrobat Reader is available at:

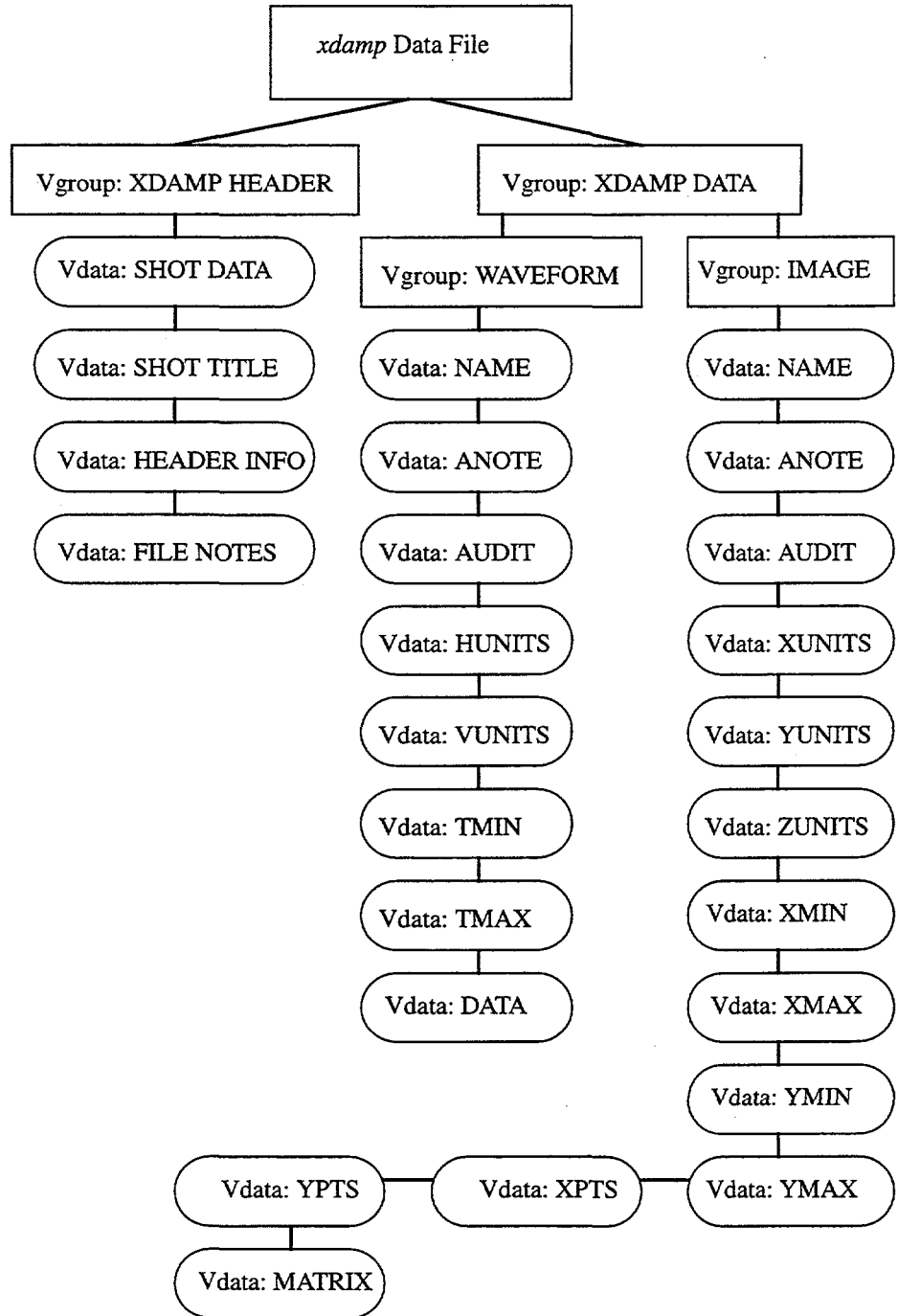
<http://www.adobe.com/>

To use zoom, you need a three button mouse and appropriate mouse driver software. So far, Mouse Systems and Logitech mice have been tested and work, without bothering with the driver.

If you are having difficulty with *xdamp* recognizing the **xdamp.ini** file, check the default working directory in any shortcut you are using. *xdamp* will look in this default file folder for the **xdamp.ini** file.

## HDF Data File Organization

Vgroup and Vdata structure for *xdamp* HDF file format with field names.



THIS PAGE INTENTIONALLY LEFT BLANK

*HDF read and write IDL procedures***READ HDF FILE  
PROCEDURE**

```

;+
; Copyright (c) 1998, Sandia Corporation. The United States
; Government retains a nonexclusive license in this
; software as prescribed in AL 88-1 and AL 91-7. Export
; of this program may require a license from the United
; States Government.
;*****
; procedure XI_readfile, filename, append keyword, overwrite
; keyword, getarray keyword
;
; Internal routine that opens the file, reads the data from HDF
; and loads the data common block.
;-----
PRO XI_readfile, fname, APPEND = append, OVERWRITE = overwrite, $
    GETARRAY = getarray

@XI_bases.cmn
@XI_data.cmn
@XI_jrnl.cmn
@XI_stack.cmn

XI_print, "Reading "+fname

IF KEYWORD_SET(OVERWRITE) THEN over = 1 ELSE over = 0
IF KEYWORD_SET(APPEND) THEN appnd = 1 ELSE appnd = 0
IF KEYWORD_SET(GETARRAY) THEN geta = 1 ELSE geta = 0

;*** figure out the prepend string
str = XIfile_pre(fname)
IF (geta OR over OR nfiles EQ 1) THEN str = ""
IF (auto_flag AND NOT over AND NOT appnd AND NOT geta) THEN $
    str = ""

;*** loop through the file reading the information and saving it
IF (narrays EQ 0) THEN i=1 ELSE i=narrays+1
IF (nimages EQ 0) THEN ii=1 ELSE ii=nimages+1

IF (over + appnd + geta EQ 0) THEN BEGIN
    shot_title = " "
    shot_date = " "
ENDIF
anote = " "
audit = " "
npts = 0L
s_buffer="abcdefghijklmnopqrstuvwxyabcdefghijklmnopqrstuvwxy"+ $
    "abcdefghijklmnopqrstuvwxyabcdefghijklmnopqrstuvwxy"+ $
    "abcdefghijklmnopqrstuvwxyacdefgh"

;*** HDF FORMAT
IF (HDF_ISHDF(fname)) THEN BEGIN
hdf_handle = HDF_OPEN(fname, /READ)

;*** be sure not an image file incorrectly selected
IF (HDF_DFR8_NIMAGES(fname) GT 0) THEN BEGIN
    XI_print, fname+" is an image file."
    HDF_CLOSE, hdf_handle
RETURN

```

```

ENDIF

;*** find header Vgroup
vgxh = -1
REPEAT BEGIN
  t = vgxh
  vgxh = HDF_VG_GETID(hdf_handle, t)
  vid = HDF_VG_ATTACH(hdf_handle, vgxh, /READ)
  HDF_VG_GETINFO, vid, NAME=tname
  HDF_VG_DETACH, vid
ENDREP UNTIL (tname EQ "XDAMP HEADER" OR vgxh EQ -1)
IF (vgxh NE -1) THEN BEGIN
  vgxhh = HDF_VG_ATTACH(hdf_handle, vgxh, /READ)

;*** find shot information Vgroup
vgxi = -1
inote = 0
ihead = 0
REPEAT BEGIN
  t = vgxi
  vgxih = HDF_VG_GETNEXT(vgxhh, t)
  IF (HDF_VG_ISVD(vgxhh, vgxi)) THEN BEGIN
    vid = HDF_VD_ATTACH(hdf_handle, vgxi, /READ)
    HDF_VD_GET, vid, FIELDS=tname
    IF (NOT appnd AND NOT geta) THEN BEGIN
      CASE tname OF
"SHOT DATE":k = HDF_VD_READ(vid,shot_date,FIELDS="SHOT DATE")
"SHOT TITLE":k=HDF_VD_READ(vid,shot_title,FIELDS="SHOT TITLE")
      "HEADER INFO": BEGIN
        k = HDF_VD_READ(vid, stg, FIELDS="HEADER INFO")
        header(ihead) = STRING(stg)
        ihead = ihead+1
      ENDCASE
      "FILE NOTES": BEGIN
        k = HDF_VD_READ(vid, stg, FIELDS="FILE NOTES")
        fnotes(inote) = STRING(stg)
        inote = inote+1
      ENDCASE
      ELSE:
      ENDCASE
    ENDIF
    HDF_VD_DETACH, vid
  ENDIF
  ENDREP UNTIL (vgxi EQ -1)
  IF (NOT appnd AND NOT geta) THEN BEGIN
    shot_date = STRING(shot_date)
    shot_title = STRING(shot_title)
  ENDIF
  HDF_VG_DETACH, vgxhh
ENDIF

;*** find data Vgroup
vgxd = -1
REPEAT BEGIN
  t = vgxd
  vgxd = HDF_VG_GETID(hdf_handle, t)
  vid = HDF_VG_ATTACH(hdf_handle, vgxd, /READ)
  HDF_VG_GETINFO, vid, NAME=tname
  HDF_VG_DETACH, vid
ENDREP UNTIL (tname EQ "XDAMP DATA")

```

```

vgxdh = HDF_VG_ATTACH(hdf_handle, vgxd, /READ)

;*** find waveform or image Vgroup
vgwd = -1
REPEAT BEGIN
  t = vgwd
  vgwd = HDF_VG_GETNEXT(vgxdh, t)
  IF (HDF_VG_ISVG(vgxdh, vgwd)) THEN BEGIN
    vid = HDF_VG_ATTACH(hdf_handle, vgwd, /READ)
    HDF_VG_GETINFO, vid, NAME=tname
    HDF_VG_DETACH, vid
  ENDIF
ENDREP UNTIL (tname EQ "WAVEFORM" or tname EQ "IMAGE")
vgwdh = HDF_VG_ATTACH(hdf_handle, vgwd, /READ)

;*** now loop through the waveforms or images
id = -1
done = 0
id = HDF_VG_GETNEXT(vgwdh, id)
REPEAT BEGIN
  flag = 0
  REPEAT BEGIN
    vds = HDF_VD_ATTACH(hdf_handle, id, /READ)
    HDF_VD_GET, vds, FIELDS=testfield
    CASE testfield OF
      "NAME":      k = HDF_VD_READ( vds, aname, FIELDS="NAME")
      "ANOTE":     k = HDF_VD_READ( vds, anote, FIELDS="ANOTE")
      "AUDIT":     k = HDF_VD_READ( vds, audit, FIELDS="AUDIT")
      "HUNITS":    k = HDF_VD_READ( vds, hunit, FIELDS="HUNITS")
      "VUNITS":    k = HDF_VD_READ( vds, vunit, FIELDS="VUNITS")
      "TMIN":      k = HDF_VD_READ( vds, tmin, FIELDS="TMIN")
      "TMAX":      k = HDF_VD_READ( vds, tmax, FIELDS="TMAX")
      "XUNITS":    k = HDF_VD_READ( vds, xunit, FIELDS="XUNITS")
      "YUNITS":    k = HDF_VD_READ( vds, yunit, FIELDS="YUNITS")
      "ZUNITS":    k = HDF_VD_READ( vds, zunit, FIELDS="ZUNITS")
      "XPTS":      k = HDF_VD_READ( vds, xpts, FIELDS="XPTS")
      "YPTS":      k = HDF_VD_READ( vds, ypts, FIELDS="YPTS")
      "XMIN":      k = HDF_VD_READ( vds, xmin, FIELDS="XMIN")
      "XMAX":      k = HDF_VD_READ( vds, xmax, FIELDS="XMAX")
      "YMIN":      k = HDF_VD_READ( vds, ymin, FIELDS="YMIN")
      "YMAX":      k = HDF_VD_READ( vds, ymax, FIELDS="YMAX")
      "DATA":      BEGIN
                    k = HDF_VD_READ( vds, ary, FIELDS="DATA")
                    flag = 1
                  ENDCASE
      "MATRIX":    BEGIN
                    k = HDF_VD_READ( vds, mat, FIELDS="MATRIX")
                    flag = 1
                  ENDCASE
    ELSE:
    ENDCASE
    HDF_VD_DETACH, vds
    id = HDF_VG_GETNEXT(vgwdh, id)
  ENDREP UNTIL flag

  CASE tname OF
    "WAVEFORM":   BEGIN
                  dbname(0).name = str+STRING(aname)
                  dbname(0).anote = STRING(anote)
                  dbname(0).audit = STRING(audit)

```

```

IF (STRLEN(STRCOMPRESS(dbase(0).audit,/REMOVE_ALL))EQ 0) THEN $
  dbase(0).audit = dbase(0).name+":"
npts = N_ELEMENTS(ary)
IF (npts GT maxsize) THEN BEGIN
  XI_print, "Data array too large, resampling."
  nskip = FIX((npts-1)/maxsize)+1
  nold = npts
  npts = FIX(npts/nskip)
  tmax = tmin+nskip*(npts-1)*(tmax-tmin)/(nold-1)
  ary = ary(nskip*INDGEN(npts))
ENDIF
dbase(0).npts = npts
dbase(0).xunits = STRING(hunit)
dbase(0).yunits = STRING(vunit)
dbase(0).tmin = tmin
dbase(0).tmax = tmax
dbase(0).yarray(*) = 0.0
dbase(0).yarray = ary
IF (over) THEN BEGIN
  FOR j = 0, next_wstack-1 DO BEGIN
    k = wstack(j)
    IF dbase(k).name EQ dbase(0).name THEN dbase(k) = dbase(0)
  ENDFOR
ENDIF ELSE BEGIN
  dbase(i) = dbase(0)
  i = i+1
  IF (i GT maxarrays) THEN BEGIN
    id = -1
    XI_print, "Too many waveforms, increase maxarrays."
  ENDF
ENDELSE
;*** Done with waveforms, may be images to read
IF (id EQ -1) THEN BEGIN
  HDF_VG_DETACH, vgw dh
  vgw d = -1
  REPEAT BEGIN
    t = vgw d
    vgw d = HDF_VG_GETNEXT(vgxdh, t)
    IF (vgw d NE -1 AND HDF_VG_ISVG(vgxdh, vgw d)) THEN BEGIN
      id = HDF_VG_ATTACH(hdf_handle, vgw d, /READ)
      HDF_VG_GETINFO, id, NAME=tname
      HDF_VG_DETACH, id
    ENDF
  ENDRP UNTIL (tname EQ "IMAGE" OR id EQ -1 OR vgw d EQ -1)
  IF (vgw d NE -1) THEN BEGIN
    vgw dh = HDF_VG_ATTACH(hdf_handle, vgw d, /READ)
    id = -1
    done = 0
    id = HDF_VG_GETNEXT(vgw dh, id)
  ENDF ELSE id = vgw d
ENDIF
ENDCASE

"IMAGE": BEGIN
  ibase(0).name = str+STRING(aname)
  ibase(0).anote = STRING(anote)
  ibase(0).audit = STRING(audit)
  IF (STRLEN(STRCOMPRESS(ibase(0).audit,/REMOVE_ALL))EQ 0) THEN $
    ibase(0).audit = ibase(0).name+":"
  IF (xpts GT nxmax OR ypts GT nymax) THEN BEGIN

```

```

        XI_print, "Image arrays too large, increase size."
        RETURN
    ENDIF
    ibase(0).xpts = xpts
    ibase(0).ypts = ypts
    ibase(0).xunits = STRING(xunit)
    ibase(0).yunits = STRING(yunit)
    ibase(0).zunits = STRING(zunit)
    ibase(0).xmin = xmin
    ibase(0).xmax = xmax
    ibase(0).ymin = ymin
    ibase(0).ymax = ymax
    ibase(0).zmat(*) = 0.0
    ibase(0).zmat(0:xpts-1,0:ypts-1) = mat
    IF (over) THEN BEGIN
        FOR j = 0, next_istack-1 DO BEGIN
            k = istack(j)
            IF ibase(k).name EQ ibase(0).name THEN ibase(k) = ibase(0)
        ENDFOR
    ENDIF ELSE BEGIN
        ibase(ii) = ibase(0)
        ii = ii+1
        IF (ii GT maximages) THEN BEGIN
            id = -1
            XI_print, "Too many images, increase maximages."
        ENDIF
    ENDELSE
    ENDCASE
    ENDCASE
    ENDREP UNTIL (id EQ -1)
    HDF_CLOSE, hdf_handle
ENDIF

IF (NOT over) THEN narrays = i-1 & nimages = ii-1

;*** run through strings, stripping out any exclamation points,
; they are bad
FOR i = 1, narrays DO BEGIN
    j = STRPOS(dbase(i).name, "!")
    WHILE (j NE -1) DO BEGIN
        a = dbase(i).name
        STRPUT, a, "|", j
        dbase(i).name = a
        j = STRPOS(dbase(i).name, "!")
    ENDWHILE
    j = STRPOS(dbase(i).yunits, "!")
    WHILE (j NE -1) DO BEGIN
        a = dbase(i).yunits
        STRPUT, a, "|", j
        dbase(i).yunits=a
        j = STRPOS(dbase(i).yunits, "!")
    ENDWHILE
    j = STRPOS(dbase(i).xunits, "!")
    WHILE (j NE -1) DO BEGIN
        a = dbase(i).xunits
        STRPUT, a, "|", j
        dbase(i).xunits = a
        j = STRPOS(dbase(i).xunits, "!")
    ENDWHILE

```

```

ENDFOR

;*** similarly for images
FOR i = 1, nimages DO BEGIN
  j = STRPOS(ibase(i).name,"!")
  WHILE (j NE -1) DO BEGIN
    a = ibase(i).name
    STRPUT, a, "|", j
    ibase(i).name = a
    j = STRPOS(ibase(i).name,"!")
  ENDWHILE
  j = STRPOS(ibase(i).zunits,"!")
  WHILE (j NE -1) DO BEGIN
    a = ibase(i).zunits
    STRPUT, a, "|", j
    ibase(i).zunits=a
    j = STRPOS(ibase(i).zunits,"!")
  ENDWHILE
  j = STRPOS(ibase(i).yunits,"!")
  WHILE (j NE -1) DO BEGIN
    a = ibase(i).yunits
    STRPUT, a, "|", j
    ibase(i).yunits=a
    j = STRPOS(ibase(i).yunits,"!")
  ENDWHILE
  j = STRPOS(ibase(i).xunits,"!")
  WHILE (j NE -1) DO BEGIN
    a = ibase(i).xunits
    STRPUT, a, "|", j
    ibase(i).xunits = a
    j = STRPOS(ibase(i).xunits,"!")
  ENDWHILE
ENDFOR

IF (narrays EQ 0 AND nimages EQ 0) THEN XI_print, $
  "Incorrect format or bad data, select another." $
ELSE BEGIN
  IF (narrays GT 0) THEN XI_print, $
    "Done reading "+STRTRIM(STRING(narrays),2)+" waveforms."
  IF (nimages GT 0) THEN XI_print, $
    "Done reading "+STRTRIM(STRING(nimages),2)+" images."
ENDELSE

RETURN
END ;===== end of XI_readfile routine =====

;+
; Copyright (c) 1998, Sandia Corporation. The United States Government
; retains a nonexclusive license in this software as prescribed in
; AL 88-1 and AL 91-7. Export of this program may require a license from
; the United States Government.
;+-----+
; procedure XI_savefile, filename
;
; Internal routine that actually writes a file in sicdas format
;-----+
PRO XI_savefile, new_filename

@XI_data.cmn

```

## WRITE HDF FILE PROCEDURE

```

;*** declare variables
XI_fields = ["NAME", "ANOTE", "AUDIT", "VUNITS", "HUNITS", $
            "TMIN", "TMAX", "DATA"]
XI_ftype = ["BYTE", "BYTE", "BYTE", "BYTE", "BYTE", $
            "FLOAT", "FLOAT", "FLOAT"]
nfields = N_ELEMENTS(XI_fields)

XI_ifields = ["NAME", "ANOTE", "AUDIT", "XUNITS", "YUNITS", "ZUNITS", $
            "XPTS", "YPTS", "XMIN", "XMAX", "YMIN", "YMAX", "MATRIX"]
XI_iftype = ["BYTE", "BYTE", "BYTE", "BYTE", "BYTE", "BYTE", $
            "INT", "INT", "FLOAT", "FLOAT", "FLOAT", "FLOAT", "MATRIX"]
nifields = N_ELEMENTS(XI_ifields)

s_buffer = "abcdefghijklmnopqrstuvwxyabcdefghijklmnopqrstuvwxy" + $
          "abcdefghijklmnopqrstuvwxyabcdefghijklmnopqrstuvwxy" + $
          "abcdefghijklmnopqrstuvwxyacdefgh"
f_buffer = 1.0

;*** open a new HDF file, get the file handle
hdf_handle = HDF_OPEN(new_filename, /WRITE, /CREATE)

;*** create a vgroup id, attach it and set the name
vg_header = HDF_VG_ATTACH(hdf_handle, -1, /WRITE)
HDF_VG_SETINFO, vg_header, NAME="XDAMP HEADER"

;*** do another for the data
vg_data = HDF_VG_ATTACH(hdf_handle, -1, /WRITE)
HDF_VG_SETINFO, vg_data, NAME="XDAMP DATA"

;*** first write shot date and shot title
vds = HDF_VD_ATTACH(hdf_handle, -1, /WRITE)
HDF_VD_FDEFINE, vds, "SHOT DATE", /BYTE
HDF_VD_WRITE, vds, "SHOT DATE", STRTRIM(shot_date)
HDF_VD_INSERT, vg_header, vds
HDF_VD_DETACH, vds
vds = HDF_VD_ATTACH(hdf_handle, -1, /WRITE)
HDF_VD_FDEFINE, vds, "SHOT TITLE", /BYTE
HDF_VD_WRITE, vds, "SHOT TITLE", STRTRIM(shot_title)
HDF_VD_INSERT, vg_header, vds
HDF_VD_DETACH, vds

;*** now write in header stuff
FOR i = 0, N_ELEMENTS(header)-1 DO BEGIN
    temp = STRTRIM(header(i))
    IF (STRLEN(temp) GT 0) THEN BEGIN
        vds = HDF_VD_ATTACH(hdf_handle, -1, /WRITE)
        HDF_VD_FDEFINE, vds, "HEADER INFO", /BYTE
        HDF_VD_WRITE, vds, "HEADER INFO", temp
        HDF_VD_INSERT, vg_header, vds
        HDF_VD_DETACH, vds
    ENDIF
ENDIF
ENDFOR

;*** now add in the file notes to the header
FOR j = 0, N_ELEMENTS(fnotes)-1 DO BEGIN
    temp = STRTRIM(fnotes(j))
    IF (STRLEN(temp) GT 0) THEN BEGIN
        vds = HDF_VD_ATTACH(hdf_handle, -1, /WRITE)
        HDF_VD_FDEFINE, vds, "FILE NOTES", /BYTE
        HDF_VD_WRITE, vds, "FILE NOTES", temp
    ENDIF
ENDIF

```

```

        HDF_VD_INSERT, vg_header, vds
        HDF_VD_DETACH, vds
    ENDIF
ENDFOR

;*** now add waveforms if any in memory
IF (narrays GT 0) THEN BEGIN
    vg_wvfm = HDF_VG_ATTACH(hdf_handle, -1, /WRITE)
    HDF_VG_SETINFO, vg_wvfm, NAME="WAVEFORM"

;*** first get a set of vdata ids to put a waveform in
    FOR j = 1, narrays DO BEGIN
        FOR i = 0, nfields-1 DO BEGIN
            vds = HDF_VD_ATTACH(hdf_handle, -1, /WRITE)
            CASE XI_ftype(i) OF
                "BYTE":    HDF_VD_FDEFINE, vds, XI_fields(i), /BYTE
                "LONG":    HDF_VD_FDEFINE, vds, XI_fields(i), /LONG
                "FLOAT":   HDF_VD_FDEFINE, vds, XI_fields(i), /FLOAT
            ENDCASE
            CASE XI_fields(i) OF
                "NAME":    HDF_VD_WRITE, vds, "NAME", dbase(j).name
                "ANOTE":   HDF_VD_WRITE, vds, "ANOTE", dbase(j).anote
                "AUDIT":   HDF_VD_WRITE, vds, "AUDIT", dbase(j).audit
                "HUNITS":  HDF_VD_WRITE, vds, "HUNITS", dbase(j).xunits
                "VUNITS":  HDF_VD_WRITE, vds, "VUNITS", dbase(j).yunits
                "TMIN":    HDF_VD_WRITE, vds, "TMIN", dbase(j).tmin
                "TMAX":    HDF_VD_WRITE, vds, "TMAX", dbase(j).tmax
                "DATA":    HDF_VD_WRITE, vds, "DATA", $
                        dbase(j).yarray(0:dbase(j).npts-1)
            ELSE:
            ENDCASE
            HDF_VG_INSERT, vg_wvfm, vds
            HDF_VD_DETACH, vds
        ENDFOR
    ENDFOR
ENDIF

;*** now build an image description
IF (nimages GT 0) THEN BEGIN
    vg_imag = HDF_VG_ATTACH(hdf_handle, -1, /WRITE)
    HDF_VG_SETINFO, vg_imag, NAME="IMAGE"

    FOR j = 1, nimages DO BEGIN
        FOR i = 0, nifields-1 DO BEGIN
            vds = HDF_VD_ATTACH(hdf_handle, -1, /WRITE)
            CASE XI_iftype(i) OF
                "BYTE":    HDF_VD_FDEFINE, vds, XI_ifields(i), /BYTE
                "INT":     HDF_VD_FDEFINE, vds, XI_ifields(i), /INT
                "LONG":    HDF_VD_FDEFINE, vds, XI_ifields(i), /LONG
                "FLOAT":   HDF_VD_FDEFINE, vds, XI_ifields(i), /FLOAT
                "MATRIX":  HDF_VD_FDEFINE, vds, XI_ifields(i), /FLOAT, $
                        ORDER=ibase(j).xpts
            ENDCASE
            CASE XI_ifields(i) OF
                "NAME":    HDF_VD_WRITE, vds, "NAME", ibase(j).name
                "ANOTE":   HDF_VD_WRITE, vds, "ANOTE", ibase(j).anote
                "AUDIT":   HDF_VD_WRITE, vds, "AUDIT", ibase(j).audit
                "XUNITS":  HDF_VD_WRITE, vds, "XUNITS", ibase(j).xunits
                "YUNITS":  HDF_VD_WRITE, vds, "YUNITS", ibase(j).yunits
                "ZUNITS":  HDF_VD_WRITE, vds, "ZUNITS", ibase(j).zunits
            ENDIF
        ENDFOR
    ENDFOR
ENDIF

```

```
      "XPTS": HDF_VD_WRITE, vds, "XPTS", ibase(j).xpts
      "YPTS": HDF_VD_WRITE, vds, "YPTS", ibase(j).ypts
      "XMIN": HDF_VD_WRITE, vds, "XMIN", ibase(j).xmin
      "XMAX": HDF_VD_WRITE, vds, "XMAX", ibase(j).xmax
      "YMIN": HDF_VD_WRITE, vds, "YMIN", ibase(j).ymin
      "YMAX": HDF_VD_WRITE, vds, "YMAX", ibase(j).ymax
      "MATRIX": HDF_VD_WRITE, vds, "MATRIX", $
                ibase(j).zmat(0:ibase(j).xpts-1,0:ibase(j).ypts-1)
    ELSE:
    ENDCASE
    HDF_VG_INSERT, vg_imag, vds
    HDF_VD_DETACH, vds
    ENDFOR
  ENDFOR
ENDIF

;*** finished writing stuff, close file
IF (narrays GT 0) THEN BEGIN
  HDF_VG_INSERT, vg_data, vg_wvfm
  HDF_VG_DETACH, vg_wvfm
ENDIF
IF (nimages GT 0) THEN BEGIN
  HDF_VG_INSERT, vg_data, vg_imag
  HDF_VG_DETACH, vg_imag
ENDIF
HDF_VG_DETACH, vg_data
HDF_VG_DETACH, vg_header
HDF_CLOSE, hdf_handle
XI_print, "Done saving "+new_filename
RETURN
END ;===== end of XI_savefile routine =====
```

THIS PAGE INTENTIONALLY LEFT BLANK

THIS PAGE INTENTIONALLY LEFT BLANK

---

*References*

---

*IDL User's Guide, Interactive Data Language, Version 4*, Research Systems, Inc., March 1995.

*IDL Reference Guide, Interactive Data Language, Version 4*, Research Systems, Inc., March 1995.

*NCSA HDF Specification and Developer's Guide*, National Center for Supercomputing Applications, 152 Computing Applications Building, 605 East Springfield Avenue, Champaign, IL 61820.

*NCSA HDF Vset*, National Center for Supercomputing Applications, 152 Computing Applications Building, 605 East Springfield Avenue, Champaign, IL 61820.

R. Bracewell, *The Fourier Transform and Its Applications*, McGraw-Hill, New York, 1965.

R. B. Spielman, *A Non-Recursive Lowpass Digital Filter Algorithm*, Sandia National Laboratories Report No. SAND82-0930, November 1982.

W. B. Boyer, *Computer Compensation for Cable Signal Degradations*, Sandia National Laboratories Report No. SAND87-3072, December 1987.

---

## *List of Distribution Files*

---

### In *xdamp.tar* or *xdamp.zip* file

<i>xdamp.pro</i>	Main procedure
<i>xdamp.ini</i>	sample <i>xdamp</i> initialization file
<i>manual.ps</i>	PostScript version of manual
<i>manual.pdf</i>	Acrobat Reader version of the manual
<i>xdamp.html</i>	Hypertext help file
<i>pwmv.dcf</i>	sample command file
<i>XI_auto.pro</i>	automatic execution files
<i>XI_edit.pro</i>	edit menu procedures
<i>XI_etest.pro</i>	error testing procedure
<i>XI_file.pro</i>	file menu procedures
<i>XI_ftr.pro</i>	filter procedures
<i>XI_input.pro</i>	input procedures
<i>XI_legnd.pro</i>	legend procedure
<i>XI_ops_1.pro</i>	single waveform operators procedures
<i>XI_ops_2.pro</i>	dual waveform operators procedures
<i>XI_user.pro</i>	user modifiable operators procedures
<i>XI_optns.pro</i>	options procedures
<i>XI_others.pro</i>	routines from other libraries
<i>XI_print.pro</i>	print menu procedures
<i>XI_pulse.pro</i>	pulse parameter calculation procedures
<i>XI_query.pro</i>	query box procedures
<i>XI_subs.pro</i>	sub-widget box procedures
<i>XI_tools.pro</i>	tools menu procedures
<i>XI_user.pro</i>	user modifiable procedures
<i>XI_utils.pro</i>	utility procedures
<i>XIsys_sp.pro</i>	system specific procedures
<i>XI_anote.cmn</i>	array notes common
<i>XI_bases.cmn</i>	widget bases common
<i>XI_data.cmn</i>	data block common
<i>XI_ftr.cmn</i>	filter widget common
<i>XI_fnote.cmn</i>	file notes common
<i>XI_geta.cmn</i>	get array common
<i>XI_jrnl.cmn</i>	journal common
<i>XI_optns.cmn</i>	options widget common
<i>XI_plot.cmn</i>	plot information common
<i>XI_query.cmn</i>	query widget common
<i>XI_regs.cmn</i>	registers common
<i>XI_stack.cmn</i>	data stack common
<i>examplei.hdf</i>	sample hdf data file

### separately

*AAREADME.TXT*      Installation advice

---

*Distribution*

---

- 1 M. Christopher  
Mission Research Corporation  
6703 Odyssey Drive, Suite 101  
Huntsville, AL 35806
- 3 K. J. Dudley  
Building D3  
AWE  
Aldermaston, Reading, Berkshire RG74PR  
England
- 1 T. Goldsack  
Building N56  
AWE  
Aldermaston, Reading, Berkshire RG74PR  
England
- 2 J. Levine  
Physics International Corporation  
PO Box 5010  
San Leandro, CA 94577
- 2 T. Lockner, R. Stinnett  
Quantum Manufacturing Technologies Inc.  
3701 Hawkins, St. NE  
Albuquerque, NM 87109
- 10 MS1106 R. Broyles, 9342-1  
1 MS1106 R. E. Craven, 9342  
1 MS1106 J. S. Gustwiller, 9342-1  
1 MS1155 W. Beezhold, 9300  
1 MS1159 W. H. Barrett, 9344  
1 MS1159 L. B. Bishop, 9344  
1 MS1159 J. I. Greenwoll, 9344  
1 MS1159 M. A. Hedemann, 9344  
1 MS1159 R. E. Klingler, 9344  
1 MS1167 T. A. Zarick, 9343  
1 MS1168 W. B. Boyer, 9321  
1 MS1168 P. F. McKay, 9321  
10 MS1168 R. O'Rourke, 9309  
1 MS1168 J. W. Lee, 9321  
1 MS1168 R. E. French, 9309  
1 MS1168 D. B. Hughes, 9342  
10 MS1179 W. P. Ballard, 9341  
1 MS1179 D. E. Beutler, 9341  
1 MS1179 J. R. Lee, 9340  
1 MS1179 T. B. Parson, 9341

---

1	MS1186	D. B. Seidel, 9542
1	MS1186	L. P. Mix, 9542
1	MS1188	M. Trahan, 9512
1	MS1188	J. S. Wagner, 9512
1	MS1194	C. Deeney, 9544
1	MS1194	D. H. McDaniel, 9540
1	MS1194	L. Ruggles, 9573
1	MS1194	J. F. Seamen, 9573
1	MS1194	R. B. Spielman, 9544
2	MS1194	J. L. Porter, 9573
1	MS1196	R. C. Mock, 9577
1	MS1196	T. J. Nash, 9577
1	MS9018	Central Technical Files, 8940-2
2	MS0899	Technical Library, 4916
2	MS0619	Review & Approval Desk, 12690 For DOE/OSTI
3	MS0161	Patent and Licensing Office, 11500