



Copyright Notice

© 2005 Hamilton Technical Services  
All Rights Reserved

The information in this document is subject to change without notice, and is provided without any warranty of any type. Please report any errors or corrections to the address below. No part of this document may be reproduced or retransmitted in any form or by any means, electronically or mechanically, including photocopying or scanning, for any purpose other than the purchaser's personal use, without the express written permission of Hamilton Technical Services. The software described in this document is licensed, and may be used only in accordance with the terms of that agreement. It may not be copied on any medium except as specifically allowed in the license agreement.

Stable32 User Manual  
Version 1.46  
12/30/2005

Hamilton Technical Services  
650 Distant Island Drive  
Beaufort, SC 29907-1580  
USA

Phone: 843-525-6495  
Fax: 843-525-0251  
E-Mail: [stable32@wriley.com](mailto:stable32@wriley.com)  
Web: [www.wriley.com](http://www.wriley.com)

Table of Contents

GENERAL

- Stable32.....1
- Installation.....3
- Program Model.....5
- Menu Tree.....7
- Status Bar.....9
- Data Format.....11
- Archive File Format.....15
- Filenames.....17
- Timetags.....19
- Command Line Options.....21
- Automated Operation.....23
- Log File.....25
- Configure Function.....27
- Configuration File.....29

TIME DOMAIN STABILITY

- Time Domain Stability.....31
- Allan Variance.....33
- Overlapping Allan Variance.....35
- Modified Allan Variance.....37
- Time Variances.....39
- Total Allan Variance.....41
- Total modified Variance.....43
- Hadamard Variance.....45
- Overlapping Hadamard Variance.....47
- Total Hadamard Variance.....49
- Thêo1.....51
- MTIE.....55
- TIE rms.....57
- Dynamic Stability.....59
- Confidence Intervals.....61
- Bias Functions.....67

Dead Time.....	71
Autocorrelation .....	73
FREQUENCY DOMAIN STABILITY	
Frequency Domain Stability .....	75
Noise Processes .....	77
Noise Spectra .....	81
Domain Conversions .....	83
FILE FUNCTIONS	
File Functions .....	85
Filenames Function .....	87
Open Function .....	89
Read Archive Function .....	91
Read Archive Configure .....	93
Multicolumn Data.....	95
Large Data File .....	97
Data File Opened.....	99
Inputs .....	101
Save Function .....	103
Add Function.....	105
Database Function.....	107
Database Details .....	111
Database Configure .....	113
READ FUNCTION	
Read Function.....	115
Read Plot Function.....	117
Read Options Function .....	119
Read Lines Function.....	121
Read Notes Function .....	123
EDITING FUNCTIONS	
Print Function .....	125
Edit Function.....	127
Find Function.....	131
Replace Function.....	133

Convert Function .....	135
Normalize Function .....	137
Average Function .....	139
Fill Function .....	141
Scale Function .....	143
Part Function.....	145
ANALYSIS FUNCTIONS	
Statistics Function .....	147
Check Function.....	149
Drift Function.....	151
Sigma Function .....	155
Power Function.....	161
Autocorrelation Function .....	167
Histogram Function .....	171
Plot Function.....	173
Data Plot Function .....	175
Plot Options Function .....	177
Plot Lines Function .....	181
Plot Notes Function.....	183
Timetags Function .....	185
Noise Function .....	187
Clear Function.....	191
Run Function .....	193
Run Plot Function .....	197
Run Options Function.....	199
Run Lines Function.....	201
Run Notes Function.....	205
Dynamic Stability Function .....	207
Regularize Function .....	211
Domain Function.....	213
MISCELLANEOUS FUNCTIONS	
Auto# Functions .....	215
3-Cornered Hat Function.....	219

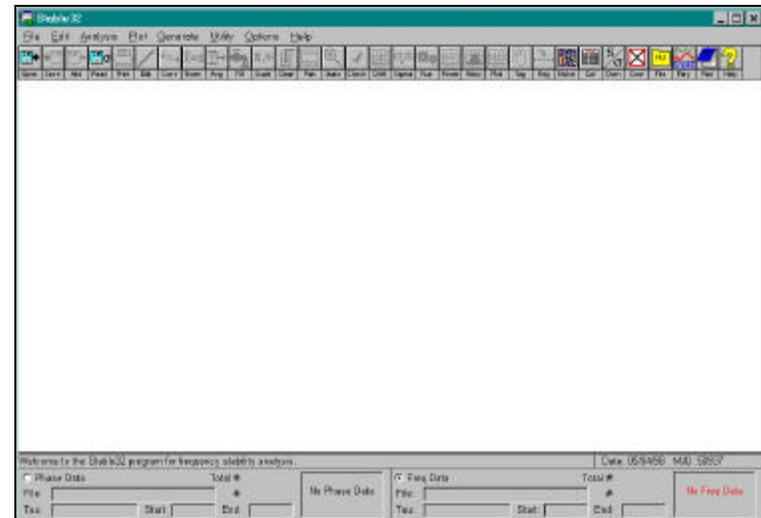
Calendar Function.....	221
Pad.....	223
Play.....	225
<b>OTHER INFORMATION</b>	
Glossary.....	227
References.....	231
Example.....	235
Test Suite.....	237
Analysis Procedure.....	241
Validation.....	245
Acknowledgments.....	247
<b>APPENDICES</b>	
I. Stable32 Functions.....	249
II. Special Characters.....	251
III. Editing Keys.....	257
VI. Configuration File.....	259
V. Hints and Notes.....	269
VI. 5110Comm Communications Program.....	273
<b>INDEX</b>	
Index.....	275
Notes.....	281

## Stable32

### • Introduction

Stable32 is a 32-bit PC program for the analysis of frequency stability. It allows the entry and editing of phase and frequency data, the calculation of stability statistics, and the plotting and printing of phase, frequency and stability data. Stable32 runs under Microsoft Windows 95/NT or higher, and requires a 200 MHz Pentium-class computer with a CD drive, 32MB of RAM and a XGA color monitor. A graphics printer is needed for text and graphics printouts. General help information is available via the F1 key, or by accessing the Help menu or toolbutton, while context-sensitive help is available by pressing the Help button within a particular function. Complete information about the Stable32 program is available in the User Manual.

Stable32 uses the GraphiC scientific plotting package under the terms of a royalty license agreement with Scientific Endeavors Corporation, Kingston, TN 37763. It may include the PlayW program for replaying the \*.TKF graphics files produced by GraphiC



### • Features

Stable32 file manipulations include entering and opening phase and frequency files, combining data files, and storing all or a portion of a data array. Data is stored in space or comma-delimited multicolumn ASCII format, with gaps denoted by a value of zero.

Editing features include displaying, printing changing, correcting, converting, scaling, inserting, or deleting data, or removing outliers, filling gaps and adding timetags.

Plotting can be done for all or a portion of the data, with automatic or user-defined scales, titles and annotations.

Stable32 includes provisions for the generation of simulated power-law clock noise with the desired offset, stability and drift. It also has a function for the conversion between time and frequency domain measures of frequency stability.

- **License**

A license is hereby granted to the purchaser for the use of this program. Stable32 may be installed on any number of computers within the immediate user group that purchased it, and backup copies may be made, but only one copy of the program may be used at any time. Additional copies may not be distributed to others.

This program has been extensively tested, but it is never possible to declare a program completely bug-free. No warranty is made, nor is any liability assumed, in connection with the use of the program.

- **Release Notes**

These notes describe the latest changes to the Stable32 program as of the date shown on the cover of this User Manual. Please see the User Manual Supplement pages, if any, for more recent changes, and the Readme.txt file for a more complete revision history. Please use the context-sensitive help topics for more detailed information about new features.

Release 1.45      Added provisions for including masks on stability plots.  
Release 1.46      Added the dynamic stability and database functions.

- **Support**

Support for the Stable32 program is available from:

W. J. Riley	Telephone:
Hamilton Technical Services	843-525-6495
650 Distant Island Drive	Fax: 843-525-0251
Beaufort, SC 29907 USA	E-Mail: stable32@wiley.com

On-line support, including tutorial papers, test data and updates, is available from the Hamilton Technical Services web site at:

<http://www.wiley.com>

## Installation

- **Installation**

Stable32 is distributed with a standard Windows installation program called SETUP.EXE, which should start automatically when the CD is inserted into the drive. To install Stable32 manually, use the Add/Remove Programs applet in Control Panel under Start/Settings, or simply execute D:\SETUP from Start/Run (where D is the drive letter of your CD drive). By default, this will create a folder called STABLE32 under the C:\Program Files directory on your hard disk, and will copy and decompress all the Stable32 files to it. No changes will be made to the Windows Registry, WIN.INI or any other system files except to create a Stable32 program group containing an icon for Stable32. Only about 5 MB of disk space is required. Use the included uninstall program to remove the Stable32 program files (any user-generated files must be removed manually).

- **Files**

The Stable32 installation comprises the following files, which are all located in the STABLE32 directory. Please examine the README.TXT file for any supplementary information. It may be read with Notepad.

**Executables:**

STABLE32.EXE	PLAYWNT.EXE*	5110COMM.EXE*
--------------	--------------	---------------

**Dynamic Link Libraries:**

DTBL32.DLL	FREQC32.DLL	GPCWDLL.DLL
GPCWDRVN.DLL	LICENS32.DLL	CGM.DLL
HPGL.DLL	POSTSCR.DLL	TIFF.DLL
SCODL.DLL	PIC.DLL	GEM.DLL
LIBPQ.DLL	LIBEAY32.DLL	LIBINTL-2.DLL
LIBICONV-2.DLL	SSLEAY32.DLL	

**GraphicC Fonts:**

BLOCK.FNT	COMPGRMA.FNT	COMPITAL.FNT
COMPLEX.FNT	COMSCR.FNT	DUPLEX.FNT
ENGOTH.FNT	MATRIX.FNT	MICROB.FNT
MICROG.FNT	NEWS.FNT	NEWSGRM.FNT
RUSSIAN.FNT	SIMGRMA.FNT	SIMPLEX.FNT
SIMSCR.FNT	SWISS.FNT	SWISSBIT.FNT
SWISSBLD.FNT	SWISSITL.FNT	SWISSN.FNT
TRIPITAL.FNT	TRIPLEX.FNT	

**Help Files:**

STABLE32.HLP	GRAPHICW.HLP	PLAYW.HLP*
5110COMM.HLP*		

**Sample Data:**

FREQ.DAT	PHASE.DAT	SAMPLE.DAT
TST_SUIT.DAT	NBS.DAT	NOISEPIX.TKF

### Miscellaneous:

MESSAGES.TAB	MESSAGES.DAT	HDR.TIF
HDR1.POS	HDR2.POS	STABLE32.INI
README.TXT	5110COMM.INI*	

### Documentation:

MANUAL146.PDF	WELCOME.PDF	CHART.PDF
FUNCTIONS.PDF		

\* = Optional File

## Program Model

### • Overview

The Stable32 program is based on a simple main window with a number of modal dialog boxes to implement the functions. These functions manipulate phase or frequency data arrays. The overall approach is similar to that used in the popular Stable/DOS and Stable/Win programs.

### • Main Window

The main window is empty except for menu and toolbar rows at the top, and status bar rows at the bottom. The menu and toolbar provide the command interface, while the status bar displays information about the current phase and frequency data.

### • Dialog Boxes

Modal dialog boxes support all of the Stable32 functions. These child windows use a consistent button naming convention. The **OK** and **Cancel** buttons either accept or reject any changes that have been made, while the **Calc** and **Close** buttons control functions that can be either repeated or ended. All of the Stable32 dialog boxes may be closed with the **Escape** key. Other standard Windows buttons and controls are used as appropriate for the particular function.

The use of default buttons is avoided so that the **Enter** key can be used to terminate edit control entries. This causes the entry to be reformatted and checked for validity.

The only modeless dialog boxes are those for the Find and Replace functions.

### • Data Arrays

The basic Stable32 idiom is a pair of phase (really time) and frequency data arrays,  $x[i]$  and  $y[i]$  respectively, where the index  $i$  refers to data points equally-spaced in time. These usually contain equivalent data, and conversions between them are provided. The  $x$  values have units of time in seconds, and the  $y$  values are (dimensionless) fractional frequency,  $\Delta f/f$ . The  $x(t)$  time fluctuations are related to the phase fluctuations by  $\phi(t) = x(t) \cdot 2\pi\nu_0$ , where  $\nu_0$  is the carrier frequency in Hz. Both are commonly called "phase" to distinguish them from the independent time variable,  $t$ . The data sampling or measurement interval,  $\tau_0$ , has units of seconds. It is called "tau" in many of the dialog boxes. The analysis or averaging time,  $\tau$ , may be a multiple of  $\tau_0$  ( $\tau = m\tau_0$ , where  $m$  is the averaging factor). If units other than seconds are used for tau, some of the results (such as the FFT Fourier frequency scale) will not be correct.

In addition, there are corresponding phase and frequency timetag arrays. These optional arrays are useful for data editing, but are not needed for analysis.

- **Results Display**

Results are displayed in text controls without boxes.

- **Parameter Entry**

Parameter entries are made in edit controls denoted by boxes. Default values are offered for most entries, and these often provide the desired result. Entry error trapping is done either by correcting the value or displaying a dialog box showing the allowable range. The program then restores the previous value and positions the caret in the field for editing.

## Menu Tree

- **Menu Tree**

The Stable32 program uses the following menu tree:

<u>Item</u>	<u>Function</u>	<u>Description</u>
<u>F</u> ile		
·	<u>O</u> pen	Open phase or frequency data file
·	<u>S</u> ave	Save current phase or frequency data
·	<u>A</u> dd	Add data file to current phase or frequency data
·	<u>R</u> ead	Read stability data file
·	<u>D</u> atabase	Read data from TSC MMS database
·	<u>P</u> rint	Print phase or frequency data
·	<u>E</u> xit	Exit Stable32 program
<u>E</u> dit		
·	<u>E</u> dit	Edit phase or frequency data
·	<u>C</u> onvert	Convert between phase and frequency data
·	<u>N</u> ormalize	Normalize phase or frequency data for zero mean
·	<u>A</u> verage	Combine data into a longer averaging time
·	<u>F</u> ill	Fill gaps in phase or frequency data
·	<u>R</u> egularize	Regularize data by using timetags to insert gaps
·	<u>S</u> cale	Scale phase or frequency data
·	<u>P</u> art	Remove all except part of phase or frequency data
·	<u>C</u> lear	Clear phase or frequency data
<u>A</u> nalysis		
·	<u>S</u> tatistics	Calculate basic statistics for phase or frequency data
·	<u>C</u> heck	Check frequency data for outliers
·	<u>D</u> rift	Calculate drift for phase or frequency data
·	<u>S</u> igma	Calculate Allan deviation for phase or frequency data
·	<u>R</u> un	Calculate stability statistics
·	<u>D</u> AVAR	Perform a dynamic stability analysis
·	<u>P</u> ower	Plot power spectrum of phase or frequency data
·	<u>H</u> istogram	Plot histogram for phase or frequency data
·	<u>A</u> utocorrelation	Plot autocorrelation for phase or frequency data
<u>P</u> lot		
·	<u>P</u> hase	Plot phase data
·	<u>F</u> requency	Plot frequency data
<u>G</u> enerate		
·	<u>N</u> oise	Generate simulated power law clock noise
·	<u>T</u> imetags	Generate timetag data
<u>U</u> tility		
·	<u>C</u> alendar	Display a DOY/MJD calendar
·	<u>D</u> omain	Perform frequency-time domain conversions
·	<u>I</u> nputs	Set input file scaling values
·	<u>P</u> lay	Replay a *.TKF plot file
·	<u>T</u> SC 5110A	Invoke the 5110Comm applet
·	<u>S</u> table32.ini	Open Stable32 INI file
·	<u>L</u> og	Open Stable32 log file
·	<u>A</u> uto#	Execute automatic analysis script 1-3

## Options

- **C**onfigure Set Stable32 program configuration options
- **F**ilenames Set filenames
- **S**tatus Bar Opens submenu to access status bar controls

## Help

- **A**bout Display information about the Stable32 program
- **I**ndex Display the general Help index
- **U**sing Help Display information about using Help
- **S**table on the Web Access the Hamilton Technical Services web site
- **F**eedback Send feedback re Stable32 via e-mail

## Status Bar

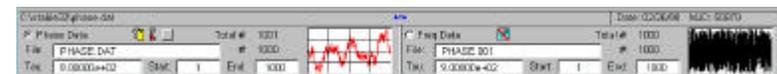
### • Purpose

The Status Bar is a permanent area at the bottom of the screen that displays status information about the current phase and frequency data arrays. Some of this information is editable.

The top row of the status bar displays either a prompt message about the current operation, or shows the full filename of the active data plus the current date and MJD. The active data type is selected with the Phase Data or Freq Data radio buttons, or by clicking the L mouse button on their respective plots. This row also shows the total # of points in each data file. The next row shows the file titles and the # of analysis points. The file title may be edited. The bottom row shows the tau value, and the analysis start and end points. Those parameters may also be changed by entering the desired values into their corresponding edit boxes.

The Stable32 status bar optionally shows small plots of the phase and frequency data. Those plots are activated by the Status Plots checkbox of the Configure function. The active status bar plot is red and the inactive one is black (which can be made the active data type by clicking with the L mouse button). The active plot may be zoomed in (centered on the position of the mouse cursor) with the R mouse button, and un-zoomed out with the L mouse button. The analysis limits change accordingly. The status plots are available only with XGA (1024x768) or higher screen resolution, and then only if the main window width is large enough.

The status bar also displays bitmaps for each data type to indicate (1) the data has changed and is unsaved, (2) the data has corresponding timetags, and (3) the analysis limits are restricted. The full analysis limits may be restored by clicking on the small button next to the analysis limit bitmap. A blue bitmap at the top center of the status bar indicates when the two data arrays contain corresponding data.



### • Controls

The Status Bar contains the following controls:

<b>Control</b>	<b>Type</b>	<b>Description</b>
Phase Data	Radiobutton	Select the phase array as the active data array.
Freq Data	Radiobutton	Select frequency array as the active data array.
Total #	Text	Display the total # of points in the data array
File	Text	Display the current phase or freq data file title.
#	Text	Display the # of analysis points.
Tau	Edit	Enter the averaging time, $t$ , for the current data.

Start	Edit	Enter the analysis start data point #.
End	Edit	Enter the analysis end data point #.
Plot	Graphic	Show data plot between analysis limits.
Unmarked	Button	Restore full analysis range.

- **Bitmaps**

Several bitmaps appear on the status bar beside the Phase and Freq Data buttons to indicate when the changed data is unsaved, when there are timetags, and whether the analysis limits are either restricted or too small. The latter bitmap changes when the # of analysis points is below the minimum. A button to the right of the analysis limit bitmap restores the full analysis range. A bitmap at the top center of the status bar indicates when the phase and frequency arrays contain equivalent data.

- **Operation**

Click on the **Phase** or **Freq** button or plot to select the desired data type. Enter the **Tau** value (in seconds) that applies to the current phase and frequency data. Optionally, enter values for the corresponding **Start** and **End** analysis point #s if it is desired to restrict the analysis to a portion of the data. Enter the title (short filename) of the phase or frequency data in the **File** edit box. Right and left click on the **Plot** to zoom and un-zoom the data display.

The Options - Status Bar menu items may be used to select status bar controls from the keyboard.

- **Limits**

The size of the phase and frequency data arrays is essentially unlimited, and the maximum number of data points that can be processed is determined mainly by the computer memory and speed (as set by the MaxSize configuration variable). Only the top status bar message displays the full filename drive and path information. The tau value must be positive and greater than zero. The start and end analysis point #s must be between 1 and the total # of data points.

- **Main Window**

The Stable32 main window may be resized and repositioned by the usual methods, and these settings will be saved when the program is closed and reopened again. A narrow main window width will cause the status plots to be eliminated, and a very narrow setting will make it difficult to operate the status bar controls. The toolbutton and status bar controls automatically update whenever the Windows system colors are changed.

- **Caution**

Be certain to enter the correct tau value for the phase and frequency data. Many of the conversion and analysis functions depend on this value to obtain the correct results, so enter the tau value immediately after the data is opened.

## Data Format

- **Data Format**

Stable32 data files are comprised of essentially any number (513 in Student Version) of lines of numeric data in ASCII format, with each line terminated by a carriage return/line feed. (Such a file can be read and edited with Windows Notepad.) The data may be in integer, floating point, or exponential format. Each line can have 1 or more space or comma-delimited columns, and up to 8 columns can be processed. A value of zero is treated as a gap in the data. Stable32 reads, processes and stores data in double-precision (15-digit) floating-point format.

The Stable32 file reading routine is quite robust. It will read an input file with any number of columns, and will allow selection from up to 8 columns of data. Additional columns are simply ignored. One or more spaces or commas must separate the columns, and each line may contain up to 1024 characters. A single column is assumed to the phase or frequency data, which is automatically loaded into the selected array. Multicolumn data opens a special dialog box where the desired columns for the data and (optional) timetags can be selected. The last line of the file that contains a numeric (0-9) character determines the number of columns, and first 80 characters are displayed for easy reference. Non-numeric header and trailer lines at the top and bottom of the file are ignored (including lines beginning with the UNIX-style # comment symbol). Fields of numeric data in header rows may be read incorrectly, and require subsequent editing, but they will not interfere with the reading of later rows of actual data. Slashes (/) and colons (:), as used in dates (04/23/97) and times (18:50), are converted to zeros (0) so that numeric data on the same row can be read.

Provisions are included for ignoring a selected # of non-numeric characters from the beginning of each row of phase or frequency data as it is read. The NonNums configuration variable sets the # of non-numeric characters that are skipped at the beginning of each line of data. This value (normally 0) can be used to read data files that contain non-numeric markers at the start of each line. This value can be set in either the Configure dialog box or the STABLE32.INI configuration file. Data files are parsed for numeric values one line at a time, and a non-numeric character normally terminates the reading of the current line. While this is desirable to reject comments, it will also prevent the reading of a line of actual data. Thus, if the data lines contain non-numeric characters, the NonNums feature will allow them to be read.

The MaxSize configuration variable sets the maximum number of data points that can be read without averaging. An attempt to load a file that is either larger or too small (<3 points) will generate an error message. A larger file can be processed with the Large Data File dialog box either by averaging it or by loading only the first or last portion. A file that is smaller than 3 data points cannot be meaningfully processed and is not read.

Setting the Comma configuration variable with the Configure function or directly in the [Preferences] section of the STABLE32.INI file allows commas to be substituted for decimal points in data values.

- **Header**

An optional header can be written at the top of phase and frequency data files, as shown below, by checking the Header option in the Configure function.

```
File: C:\Data\PHASE.DAT
Date: 04/08/01 17:37:59
Type: Phase
Points: 8192
Tau: 1.000000e+000
```

This information can help to identify the data, and can be shown by pressing the Display button in the Data File Opened dialog box when the file is subsequently read. The header lines are ignored when the actual data is read, and are not displayed in the timetag and data column fields of that dialog box.

- **Data Entry**

Data entry is generally done within edit controls in dialog boxes. A default entry is usually presented, and it generally represents an acceptable choice, or a way to abort the current function. The usual Windows editing keys apply.

- **Gaps & Outliers**

Stable32 includes provisions for handling gaps and outliers that often occur in time and frequency stability data. A value of zero (0) is treated as a gap in a Stable32 data file, unless it is the first or last point of phase data. Stable32 data files are simple vectors of equally spaced phase (seconds) or fractional frequency (dimensionless) values. Gaps should be included to maintain the proper implied time interval between measurements. The Stable32 analysis functions will then produce meaningful results for data having gaps. The analysis functions simply skip those points that involve a gap. For example, in the calculation of the Allan variance for frequency data, if either of the two points involved in the first difference is a gap, that Allan variance point is skipped in the summation.

Stable32 also has provisions for filling gaps in phase or frequency data by replacing them with interpolated values. Use the Fill function for this purpose. Fill first removes any leading and trailing gaps. It then uses the two values immediately before and after any interior gaps to determine linearly interpolated values within the gap. Filling is recommended before using the modified and time Allan variance functions, especially for frequency data, because these calculations are much faster for gapless data.

Stable32 has a Check function for locating and removing outliers from frequency data. While the definition of an outlier is somewhat a matter of

judgment, it is important to find and remove such points in order to use the rest of the data. Stable32 defines an outlier on the basis of its deviation from the median of the data. The user may enter a deviation limit in terms of the median absolute deviation. A  $5\sigma$  limit is the default. This is a robust way to determine an outlier, which is then replaced by a gap. Automatic outlier removal is also provided, which iteratively applies this method to remove all outliers. It should be an adjunct to, and not a substitute for, visual inspection of the data.

A message (controlled by the WarnZeroFreq configuration variable) can be displayed during phase data entry to warn of equal adjacent phase values, which will result in zero frequency points that are treated as gaps.

- **Zeros**

Besides their use to denote gaps, zeros can occur as actual data values. This happens most often at the beginning and end of phase data, and those zeros are not considered to be gaps. All other (embedded) zeros are considered as gaps in phase data, as are all zeros in frequency data. To preserve such zeros as actual values, change them to a value close to zero (e.g.  $1e-99$ ). The Find and Replace feature of the Edit function may be used to make that substitution.

A zero value in frequency data can also occur as the result of the conversion of two equal adjacent phase data points (perhaps because of limited measurement resolution), and an optional warning message is available to flag this possibility when phase data is read. This situation can be handled in the Convert function by activating the option to adjust the value of the converted frequency point from zero to  $1e-99$ .

A certain number of zeros (gaps) can be inserted into, or deleted from, phase or frequency data with the Edit After Highlighted Point feature of the Edit function. This can be used to insert gaps that correspond to an interruption in the data record. That process can be accomplished automatically for time-tagged data with the Regularize function.

- **Caution**

Header rows in a data file are not read and will be lost if the file is overwritten.

Unexpected results can occur if two adjacent phase data points are exactly the same. This corresponds to zero frequency, and can cause anomalies in the resulting stability analysis. If converted to frequency data, the resulting zero frequency data point will be treated as a gap unless the Adjust Zero Frequency option is activated.

## Archive File Format

- **TSC Archive File Format**

Timing Solutions Corporation (TSC) archive files have lines that start with either a \$ or # symbol. The \$ lines have 1 additional field, the MJD for the data on the following # lines. The MJD is a 18-character string comprised of a 5-digit integer part, a decimal point, and a 12-digit fractional part. A sample \$ line is as follows (where the \$ is in 1st column and is followed by a single space):

```
$ 48888.612500000000
```

The # lines have 5 additional fields, a 1 or 2-digit channel # between 1 and N (the # of channels in the system), a 20-character clock ID string, and 3 numerical fields. The 1st number is the nominal clock frequency (in MHz) for that channel. It is expressed as a 1 or 2-digit integer part, a decimal point, and a 15-digit fractional part, thus forming a 17 or 18-character string. The 2nd and 3rd numbers are clock phase in cycles with respect to the offset reference oscillator. The 2nd number is a rounded integer tag, while the 3rd has the full precision with a decimal point and a 12-digit fractional part. The integer parts of these numbers grow with the beat note cycles of the offset oscillator, about 10 per second, so a 9digit field is sufficient for about 3 years. Thus the 2nd number string can have 9 characters, and the 3rd number string can have 22 characters. The number fields are each separated by 2 spaces. Including spaces, the maximum length of a # line is 79 characters. A sample # line is as follows (where # is in 1st column and is followed by a single space):

```
# Clock S/N 0123456789 5.0000000000000000 12007355  
12007354.000000000000
```

Raw clock time data (in seconds) is calculated by dividing the phase number by the nominal frequency (in MHz). Corrected clock time is obtained from that value by subtracting the corresponding value for the reference channel. This subtraction removes the contribution of the common offset local oscillator. Fractional frequency is, of course, the difference between two adjacent time values divided by the time interval between them.

- **Archive File Processing**

Clock data for a certain time span can be extracted directly from TSC archive files by choosing the desired measurement and reference channels, and then reading an archive file that has data for those sources during that period. TSC archive file data can be processed either by using data from two measurement channels (one serving as the reference), or by using the common offset local oscillator (LO) as the reference. In the former case, the contribution of the offset LO cancels out. Using the LO as the reference usually requires that it be synthesized from a standard source rather than being a free-running crystal oscillator, but measurements against a low-noise crystal source can be an advantage for short averaging times. An important feature of the TSC clock measuring system is that the reference

for a particular measurement channel can be chosen after the fact, and a single archive file can provide data versus several references.

Stable32 can process TSC archive data either between two measurement channels or between a single channel and the offset LO. The latter is accomplished by using a reference channel # = 0.

- **Archive File Records**

It is convenient to adopt the convention that, while a gap in the data for all channels is a missing data point, a gap in the data for an individual measurement channel represents the start of a new record.

Stable32 can process TSC archive data for a certain record #, for the last record, for a span of MJD dates, or from a particular starting MJD to the end of the file.

- **Archive File Indexing**

Locating the correct file and channel # for a certain clock is easier with an index of the various measuring system transactions (a list of the test start and end times, channel #s and record #s for each clock under test). Use the Index option of the Read Archive function to generate an index for TSC archive files. This generates and displays an ASCII file named ARCINDEX.DAT that may be printed and/or searched for the desired clock data.

## Filenames

- **Data Filenames**

Stable32 data filenames follow the usual DOS conventions. While any names can be used, the preferred extensions are .DAT, .PHD, and .FRD for general, phase and frequency data respectively. A function that changes the data in memory optionally alters the extension to a number, which is then incremented each time that the data is changed. Phase data numerical extensions are even, and those for frequency data are odd. Automatic filename extension incrementing helps keep track of changes and avoid overwriting previous data. It is controlled by the Autoinc Filename Ext checkbox of the Configure function. The filename of the current phase or frequency data file can be changed by editing it directly in the corresponding edit control of the status bar at the bottom of the screen.

- **Plot Filenames**

A \*.TKF plot file in Tektronix 4105 format is automatically written to the current disk directory after each plot. The default names for these files are PHASE.TKF, FREQ.TKF, SIGMA.TKF, SPECTRUM.TKF, AUTO.TKF and HISTO.TKF for phase data, frequency data, stability, power spectrum and histogram plots respectively. These filenames may be changed as an option within their respective Plot Options functions. A name of NOTEK will suppress the writing of the \*.TKF file.

- **Stability Filenames**

A stability file is optionally written to the current disk directory at the end of each stability run. This file contains a table of averaging time, # of analysis points, and minimum, nominal and maximum sigma information that can be printed or re-plotted with the Read function. The default stability filename is SIGMA.TAU, which may be changed in the Run Options dialog box. This file is particularly useful to combine with other such files to create a composite stability plot for multiple runs using data over a wide range of averaging times. Writing of the stability file is controlled by the Write Sigma File checkbox of the Configure function.

- **Plot Data Filenames**

The Autocorrelation and Dynamic Stability functions write plot data files each time a plot is drawn. The autocorrelation plot data file (default name AUTO.DAT) comprises rows of lag # and ACF value. The dynamic stability plot data file (fixed name XYZ.DAT) comprises rows of averaging factor, window #, and sigma value.

## Timetags

- **Timetags**

The Stable32 program supports the limited use of timetags to help manipulate phase and frequency data. Timetag arrays are associated with each of the phase and frequency arrays.

- **Timetag Usage**

Timetags may be read from a file, or they may be generated automatically with the Timetags function. Actual timetags may be used to identify interruptions in the data with the Regularize function, which can also insert gaps into the data at these points. Generated timetags are suitable only for data inspection and editing. The presence of timetags is indicated by a small yellow tag next to the filename in the status bar. Timetags are not required for any of the analysis functions. They may be stored to disk along with their corresponding data.

If there is timetag data, and no tau input from the STABLE32.INI file, then the spacing of the first two timetags, scaled by the TagScale configuration file value, is used as the default tau when a data file is opened. For example, MJD timetags scaled by 86400 can be used to automatically set the tau value.

- **Timetag Editing**

Timetag editing is not supported by the Stable32 program, but may be accomplished by an external ASCII text-editing program such as Windows Notepad.

- **Timetag Processing**

Timetags are processed automatically by the Average and Part functions to yield the correct values that correspond to the averaged or remaining data.

## Command Line Options

- **Purpose**

Stable32 accepts several command line options. These options allow phase, frequency or archive data to be loaded, a certain tau value to be set, and an automation script to be executed when the program starts.

- **Operation**

The command line options may be entered directly from the command line, from the Windows Start-Run prompt, or from a batch file. They are entered immediately after the STABLE32 command that starts the program. The command line options begin with a - (minus), and may be entered in upper or lower case.

<u>Command</u>	<u>Description</u>
-P FILENAME	Automatically load the phase data file FILENAME
-F FILENAME	Automatically load the frequency data file FILENAME
-A FILENAME	Automatically open the archive file FILENAME
-T TAU	Automatically load the tau value TAU
-O SKIP	Skip display of multicolumn & file opened dialogs
-O AUTO#	Execute automation script (# = script number 1-3)

- **Example**

To start the Stable32 program by loading a  $\tau=900$  sec phase data file named CLOCK.DAT, enter and execute the following statement from the Windows95/NT command line:

```
STABLE32 -P CLOCK.DAT -T 900
```

- **Notes**

The tau value set with the -T command line option applies to whichever type of data is loaded. It takes precedence over any tau value set with the PhaseTau= or FreqTau= keyword items in the [Inputs] section of the STABLE32.INI configuration file.

Phase, frequency and archive documents can be automatically opened by Stable32 by associating an extension with a corresponding action that includes the respective command line option. For example, to automatically opening a \*.phd phase data file, associate the phd extension with the phase action by using c:\Program Files\Stable32\Stable32.exe -f.

If a filename is entered on the command line without specifying the data type, then Stable32 starts by invoking the Open Data File dialog box with that filename. This feature may be used by associating an extension with the Stable32 program, and clicking on any such file. More than one association may be used.

The -O SKIP option allows skipping display of the multicolumn and file opened dialog boxes when reading data with the -F or -P command. It uses the default timetag (1<sup>st</sup>) and data (2<sup>nd</sup>) columns for 2-column data, and no scaling factor is applied. Skipping is always done when a data file is opened with one of the three automation scripts.

## Automated Operation

- **Purpose**

Stable32 operation can be fully automated by using the WinBatch package from Wilson WindowWare, Inc., 5421 California Ave. SW, Seattle, WA 98136, 800-762-8383, [www.winbatch.com](http://www.winbatch.com).

- **Operation**

WinBatch uses Windows Interface Language (WIL) \*.WBT scripts. These scripts can be created with a text editor or recorded automatically from a sequence of keystrokes. They are then invoked from the command line or a menu to automate Stable32 operation by sending the series of keystrokes to the program. An example of a \*.WBT script to plot frequency data is shown below:

```
; Main menu Plot Frequency command: ALT P, F
SendKey(!PF)
; Plot Frequency function Plot command: P
If WinExist(Plot) Then WinActivate(Plot)
SendKey(P)
; Wait for plot to be drawn on screen
Delay(5)
; GraphiC menu File Plot command: ALT F, P
If WinExist(Stable32 Main) Then WinActivate(Stable32 Main)
SendKey(!FP)
; Wait for plot to be sent to printer
Delay(15)
; Graphic menu File Exit command: ALT F, X
SendKey(!FX)
; Close Plot function with Escape: ESC
If WinExist(Plot) Then WinActivate(Plot)
SendKey({ESCAPE})
```

Keystrokes corresponding to the ALT keyboard access codes for the Stable32 functions are sent via SendKey() statements. The "!" prefix denotes an ALT key. WinActivate() statements are used to direct these keystrokes to the appropriate window. The Delay() statements are needed to allow time for GraphiC plotting and printing. The ";" lines are simply comments.

- **See Also**

See the WinBatch documentation for more information regarding these programs and commands.

See the Auto# function for a limited means of stability analysis automation that is built into the Stable32 program.

## Log File

- **Log File**

A STABLE32.LOG file is automatically written to record a synopsis of all program activity during each run. This ASCII text file contains a line for each major function called. It gives a summary of the operation performed, and thus can be used to document the steps performed during an analysis. An example of a STABLE32.LOG file is shown below:

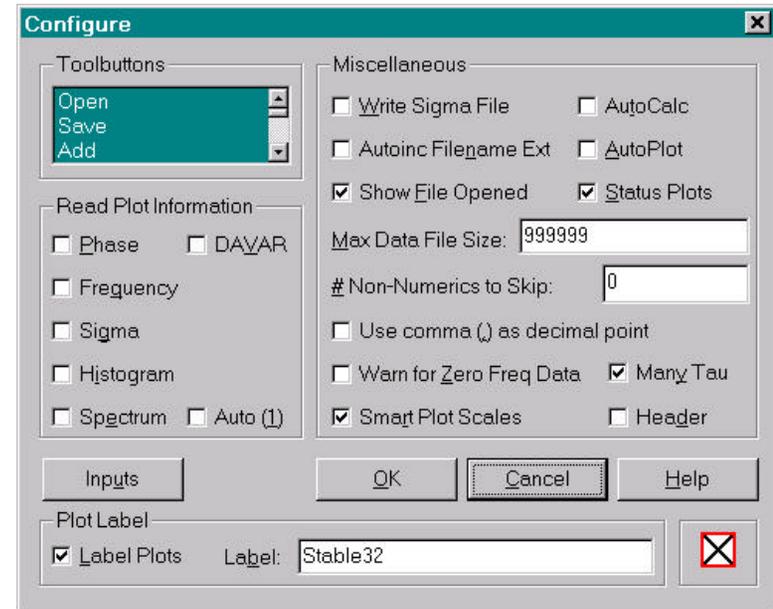
```
01/17/99 11:08 Stable32 Log File
01/17/99 11:08 Freq Data Opened: FREQ.DAT
01/17/99 11:09 Freq Sigma: Type=Overlapping,AF=1,
                Sigma=2.922319e-01
01/17/99 11:09 Freq Data Plot: C:\Stable32\FREQ.TKF
01/17/99 11:09 Stable32 Closed
```

The first two columns show the date and time in MM/DD/YY and HH:MM format respectively. The text on each line describes the operation performed and, as appropriate, a summary of the results obtained. A new log file is written each time Stable32 is started, so the previous one must be renamed in order to preserve it.

## Configure Function

- **Purpose**

Use the Configure function to set certain options of the Stable32 program. These options are stored in the STABLE32.INI configuration file.



- **Controls**

The Configure dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
<u>OK</u>	Pushbutton	Accept the configuration changes & close dialog box.
<u>Cancel</u>	Pushbutton	Abort the Configure function & close dialog box.
<u>Help</u>	Pushbutton	Invoke the Configure help topic.
Toolbuttons	Group/Combo	Select desired toolbuttons to appear on toolbar.
Read Plot Info	Group	Select plots for which to read configuration information.
<u>Phase</u>	Checkbox	Check to read configuration information for phase data plot.
<u>F</u> requency	Checkbox	Check to read configuration information for frequency data plot.
<u>S</u> igma	Checkbox	Check to read configuration information for stability plot.
<u>H</u> istogram	Checkbox	Check to read configuration information for histogram plot.

Spectrum	Checkbox	Check to read configuration information for spectrum plot.
DAVAR	Checkbox	Check to read configuration information for dynamic stability plot.
Auto (1)	Checkbox	Check to read configuration information for autocorrelation plot.
Miscellaneous	Group	Miscellaneous options.
AutoCalc	Checkbox	Check to automatically calculate when function opened.
AutoPlot	Checkbox	Check to automatically plot when function opened.
Write Sigma File	Checkbox	Check to write stability file after run.
Status Plots	Checkbox	Show status plots on status bar.
Autoinc File name	Checkbox	Check to increment filename extension when data changes.
Show File Opened	Checkbox	Check to show data plot and information when file opened.
Status Plots	Checkbox	Show mini data plots in status bar.
Max Data File Size	Edit	Set maximum data file size to read without averaging.
# Non-Numerics to Skip	Edit	Set # non-numeric characters to skip at beginning of each row of data
Use Comma (,) as Decimal Point	Checkbox	Use comma instead of decimal point for input data.
Warn for Zero Freq Data	Checkbox	Display warning if two adjacent phase data points are equal.
Smart Plot Scales	Checkbox	Use smart y-axis scales for data plots.
Header	Checkbox	Add header to data files.
Label Plots	Checkbox	Check to label plots at bottom left.
Label	Edit	Enter text to appear as plot label.

- **Operation**

Select the options desired and press **OK** to accept them or **Cancel** to abort. The **Miscellaneous** and **Toolbutton** options take effect immediately, while the **Read Plot Information** options will apply when the Stable32 program is restarted. All of these options may also be changed by editing the STABLE\_W.INI configuration file. The **Phase, Frequency, Sigma, Histogram, Spectrum, DAVAR and Auto** checkboxes control automatic reading and saving of plot options, lines and notes when the Stable32 program is opened and closed. This can be convenient for repetitive manual analysis or batch operations. Otherwise it is probably better to control the retrieving and storing of plot information manually with the individual Save and Read buttons of the Plot and Run function Options, Lines and Notes dialog boxes. The **Max Data File Size** value does not represent a program limitation, but rather is available as an option to bound the size of the data arrays. This is useful to match the computer memory and speed limits. The **Max Data File Size** value limits the number of data points that can be generated by the Noise function, and controls the opening of the Large Data File dialog box for averaging a data file as it is read.

## Configuration File

- **Description**

The STABLE32.INI file holds all the configuration data for the Stable32 program. The format of this file is that of a standard Windows \*.INI file (such as WIN.INI). The [Section] items define the sections of this file, while the Keyword=String items define the configuration information. Each keyword is set equal to a string, which may be either a Boolean value (0 or 1) or an unterminated string of characters (which may contain spaces). See Appendix IV for more information about the STABLE32.INI configuration file.

- **Operation**

Ordinarily, the reading and writing of STABLE32.INI is automatic, and it is not necessary to edit it directly. However manual editing is useful in batch file automation, and can be done from within the Stable32 program using the Pad function. The changes will take effect the next time the Stable32 program is started.

- **Caution**

Stable32 may be run without a STABLE.INI file, but no toolbuttons will be displayed. Use the Configure function to select the desired toolbuttons. A STABLE.INI file will be generated automatically when the program is closed.

## Time Domain Stability

- **Time Domain Stability**

Stable32 implements the following measures of time domain frequency stability:

<u>Statistic Name</u>	<u>Symbol</u>	<u>Description</u>
Allan Variance	$\sigma_y^2(\tau)$	Based on square of 1st differences of fractional frequency deviation values.
Modified Allan Variance	Mod $\sigma_y^2(\tau)$	Based on square of 2nd differences of averaged phase deviation values.
Total Allan Variance	$\sigma_{total}^2(\tau)$	Based on square of 1st differences of fractional frequency deviations of an extended data set.
Total Modified Variance	Mod $\sigma_{total}^2(\tau)$	Based on square of 2nd differences of averaged phase deviations of an extended data set.
Time Variance	$\sigma_x^2(\tau)$	Equal to $(\tau^2/3) \cdot$ Mod Allan variance.
Total Time Variance		Total version of time variance.
Hadamard Variance	$H\sigma_y^2(\tau)$	Based on square of 2nd differences of fractional frequency values.
Total Hadamard Variance	$H\sigma_y^2(\tau)$	Total version of Hadamard variance.
Theoretical Variance #1	Théo1	Allan estimator for large avg factor.
Bias Removed Théo1	ThéoBR	Same expected value as Allan variance.
Allan & Théo1 Hybrid	ThéoH	Combined Allan and ThéoBR.
RMS Time Interval Error	TIE rms	RMS value of time deviations.
Max Time Interval Error	MTIE	Max time deviation within window.

- **Allan Variances**

The most common time domain stability measure is the Allan variance (AVAR),  $\sigma_y^2(\tau)$ , which gives a value for the fractional frequency fluctuations as a function of averaging time,  $\tau$ . The other common time domain statistics are the modified Allan variance (MVAR), Mod  $\sigma_y^2(\tau)$  and the time variance,  $\sigma_x^2(\tau)$ . The modified Allan variance is better able to distinguish between white and flicker phase noise. The time variance (TVAR),  $\sigma_x^2(\tau)$  measures the time fluctuations of a source or time distribution system. All of these quantities are usually expressed as their square roots ADEV,  $\sigma_y(\tau)$ , MDEV, Mod  $\sigma_y(\tau)$  and TDEV,  $\sigma_x(\tau)$ . They are dimensionless except for the latter, which has units of seconds. The Allan variance can be calculated by either the non-overlapping or overlapping method. The latter provides better statistical confidence at the expense of a slight increase in computation time. The results can be shown for a single  $\tau$ , over a range of octave or decade  $\tau$  spacing, or at every  $\tau$  for which there is sufficient data. The total Allan and total modified Allan variances, TOTAVAR and TOTMVAR, are similar in purpose and expected value to the Allan and modified Allan variances respectively, but use extended data sets to provide improved confidence at larger averaging factors at the expense of a significantly longer computational time.

- **Hadamard Variances**

The Hadamard variance is a 3-sample variance that is similar to the 2-sample Allan variance. It examines the 2nd difference of the fractional frequencies, the 3rd difference of the phase variations. Because of this, the Hadamard variance, HVAR or  $H\sigma_y^2(\tau)$ , converges for the Flicker Walk FM ( $\alpha = -3$ ) and Random Run FM ( $\alpha = -4$ ) power-law noise types. It is also unaffected by linear frequency drift. The Hadamard variance can also be calculated by either the non-overlapping or overlapping method, the latter provides better statistical confidence at the expense of somewhat longer computational time. The total Hadamard variance, TOTHVAR, is a total version of the Hadamard variance.

- **Modified and Total Variances**

The modified and total variances are not distinct variances but are considered to be techniques applied to the Allan and Hadamard variances.

- **Thêo1 Variances**

Thêo1 is a special purpose statistic that provides information equivalent to the Allan variance out to averaging factors equal to 75% of the record length.

ThêoBR is a biased-removed version of Thêo1, while ThêoH is a hybrid statistic combining ThêoBR with the Allan variance.

- **Other Statistics**

Other statistics used for time domain frequency stability analysis include MTIE and TIE rms, which are most commonly used in the telecom industry.

- **Summary**

A summary of the Allan and Hadamard families of time domain stability measures can be found in the reference below, which categorizes them according to the following parameters:

Parameter	Symbol	Value	Variance	Remarks
Order of phase difference	d	2	Allan	
		3	Hadamard	
Allowable power law noise exponent	$\alpha > 1-2d$	> -3	Allan	W PM thru RW FM noise
		> -5	Hadamard	W PM thru RR FM noise
Filter factor	F	1	Unmodified	m = averaging factor = $\tau/\tau_0$
		m	Modified	
Stride factor (Stride = $\tau/S$ )	S	1	Non-overlapped estimator	
		m	Overlapped estimator	

- **Reference**

C. Greenhall and W. Riley, "Uncertainty of Stability Variances Based on Finite Differences", *Proc. 2003 PTIT Meeting*, December 2003.

## Allan Variance

- **Allan Variance**

The Allan, or 2-sample variance, AVAR, is the most common time domain measure of frequency stability. Its non-overlapping version is defined as:

$$s_y^2(t) = \frac{1}{2(M-1)} \sum_{i=1}^{M-1} [y_{i+1} - y_i]^2,$$

where  $y(i)$  is the  $i$ th of  $M$  fractional frequency values averaged over the measurement interval  $\tau$ .

In terms of phase data, the Allan variance may be calculated as:

$$s_y^2(t) = \frac{1}{2(N-2)t^2} \sum_{i=1}^{N-2} [x_{i+2} - 2x_{i+1} + x_i]^2,$$

where  $x(i)$  is the  $i$ th of the  $N = M+1$  phase values spaced by the measurement interval  $\tau$ .

The result is usually expressed as the square root,  $\sigma_y^2(\tau)$ , the Allan deviation, ADEV. The Allan variance is the same as the ordinary variance for white FM noise, but has the advantage, for more divergent noise types such as flicker noise, of converging to a value that is not dependent on the number of samples. The confidence interval of an Allan deviation estimate is also dependent on the noise type, but is often estimated as  $\pm\sigma_y(\tau)/\sqrt{N}$ .

## Overlapping Allan Variance

- **Overlapping Allan Variance**

The overlapping Allan variance is a version of the Allan variance,  $\sigma_y^2(\tau)$ , AVAR, that makes maximum use of a data set by forming all possible fully overlapping samples at each averaging time  $\tau$ . It can be estimated from a set of  $M$  frequency measurements for averaging time  $\tau = m\tau_0$ , where  $m$  is the averaging factor and  $\tau_0$  is the basic measurement interval, by the expression:

$$s_y^2(t) = \frac{1}{2m^2(M-2m+1)} \sum_{j=1}^{M-2m+1} \sum_{i=j}^{j+m-1} [y_{i+m} - y_i]^2,$$

In terms of phase data, the overlapping Allan variance can be estimated from a set of  $N = M+1$  time measurements as:

$$s_y^2(t) = \frac{1}{2(N-2m)t^2} \sum_{i=1}^{N-2m} [x_{i+2m} - 2x_{i+m} + x_i]^2,$$

The result is usually expressed as the square root,  $\sigma_y(\tau)$ , the Allan deviation, ADEV. The confidence interval of an overlapping Allan deviation estimate is better than that of a non-overlapping Allan variance estimation because, even though the additional overlapping differences are not all statistically independent, they nevertheless increase the number of degrees of freedom and thus improve the confidence in the estimation. Analytical methods are available for calculating the number of degrees of freedom for an overlapping Allan variance estimation, and using that to establish single or double-sided confidence intervals for the estimate with a certain confidence factor, based on Chi-squared statistics.

Sample variances are distributed according to the expression:

$$\chi^2 = \frac{df \cdot s^2}{\sigma^2},$$

where  $\chi^2$  is the Chi-square,  $s^2$  is the sample variance,  $\sigma^2$  is the true variance, and  $df$  is the # of degrees of freedom (not necessarily an integer). The  $df$  is determined by the # of data points and the noise type. The Stable32 program includes those procedures for the overlapping Variance function.

## Modified Allan Variance

- **Modified Allan Variance**

The modified Allan variance,  $\text{Mod } \sigma_y^2(\tau)$ , MVAR, is another common time domain measure of frequency stability. It is estimated from a set of  $M$  frequency measurements for averaging time  $\tau = m\tau_0$ , where  $m$  is the averaging factor and  $\tau_0$  is the basic measurement interval, by the expression:

$$\text{Mod} s_y^2(\mathbf{t}) = \frac{1}{2m^4(M-3m+2)} \sum_{j=1}^{M-3m+2} \sum_{i=j}^{j+m-1} \sum_{k=i}^{i+m-1} [y_{k+m} - y_k]^2,$$

In terms of phase data, the modified Allan variance is estimated from a set of  $N = M+1$  time measurements as:

$$\text{Mod} s_y^2(\mathbf{t}) = \frac{1}{2m^2 t^2 (N-3m+1)} \sum_{j=1}^{N-3m+1} \sum_{i=j}^{j+m-1} [x_{i+2m} - 2x_{i+m} + x_i]^2,$$

The result is usually expressed as the square root, MDEV or  $\text{Mod } \sigma_y(\tau)$ , the modified Allan deviation. The modified Allan variance is the same as the normal Allan variance for  $m = 1$ . It includes an additional phase averaging operation, and has the advantage of being able to distinguish between white and flicker PM noise. The confidence interval of a modified Allan deviation determination is also dependent on the noise type, but is often estimated as  $\pm \sigma_y(\mathbf{t})/\sqrt{N}$ .

## Time Variances

- **Time Variance**

The time variance, TVAR, is a measure of time stability based on the modified Allan variance. It is defined as:

$$\sigma_x^2(\tau) = (\tau^2/3) \cdot \text{Mod } \sigma_y^2(\tau).$$

The time Allan variance is equal to the standard variance of the time deviations for white PM noise. It is particularly useful for measuring the stability of a time distribution network. It is usually given as its square root, TDEV.

- **Total Time Variance**

The total time variance, TOTTVAR, is a similar measure of time stability based on the total modified variance. It is defined as:

$$\sigma_x^2(\tau) = (\tau^2/3) \cdot \text{Mod } \sigma_{\text{total}}^2(\tau).$$

It is usually given as its square root, TOTTDEV.

## Total Allan Variance

- **Total Allan Variance**

The total Allan variance, TOTAVAR, is similar to the 2-sample or Allan variance, and has the same expected value, but offers improved confidence at long averaging times. Work on total variance began with the realization that the Allan variance can "collapse" at long averaging factors because of symmetry in the data. An early idea was to shift the data by 1/4 of the record length and average the two resulting Allan variances [1]. The next step was to wrap the data in a circular fashion and calculate the average of all the Allan variances at every basic measurement interval,  $\tau_0$  [2, 3]. This technique is very effective in improving the confidence at long averaging factors but required end matching of the data. A further improvement of the total variance concept was to extend the data by reflection, first at one end of the record [4] and then at both ends [5]. This latest technique, called TOTAVAR, gives a very significant confidence advantage at long averaging times, and is an important new general statistical tool [6]. TOTAVAR is defined for phase data as:

$$Tot\ var(\mathbf{t}) = \frac{1}{2\mathbf{t}^2(N-2)} \sum_{i=2}^{N-1} [x_{i-m}^* - 2x_i^* + x_{i+m}^*]^2,$$

where  $\tau = m\tau_0$ , and the  $N$  phase values  $x_i$  measured at  $\tau = \tau_0$  are extended by reflection about both endpoints to form a virtual sequence  $x^*$  from  $i = 3-N$  to  $i = 2N-2$  of length  $3N-4$ . The original data is in the center of  $x^*$  with  $i = 1$  to  $N$  and  $x^* = x$ . The reflected portions added at each end extend from  $j = 1$  to  $N-2$  where  $x_{1-j}^* = 2x_1 - x_{1+j}$  and  $x_{N+j}^* = 2x_N - x_{N-j}$ .

TOTAVAR can also be defined for frequency data as:

$$Tot\ var(\mathbf{t}) = \frac{1}{2(M-1)} \sum_{i=1}^{M-1} [y_{i+j+1}^* - y_{i+j}^*]^2,$$

where the  $M = N-1$  fractional frequency values,  $y$ , measured at  $\tau = \tau_0$  ( $N$  phase values) are extended by reflection at both ends to form a virtual array  $y^*$ . The original data is in the center where  $y_i^* = y_i$  for  $i = 1$  to  $M$ , and the extended data for  $j = 1$  to  $M-1$  is equal to  $y_{1-j}^* = y_j$  and  $y_{M+1-j}^* = y_{M+1-j}$ .

The result is usually expressed as the square root,  $\sigma_{total}(\tau)$ , the total Allan deviation, TOTADEV. When calculated using the doubly reflected method described above, the expected value of TOTAVAR is the same as AVAR for white and flicker PM or white FM noise. Bias corrections of the form  $1/[1-a(\tau/T)]$ , where  $T$  is the record length, need to be applied for flicker and random walk FM noise, where  $a=0.481$  and  $0.750$  respectively.

The number of equivalent  $X^2$  degrees of freedom for TOTAVAR can be estimated for white FM, flicker FM and random walk FM noise by the expression  $b(T/\tau)-c$ , where  $b=1.500, 1.168$  and  $0.927$ , and  $c=0, 0.222$  and  $0.358$  respectively. For white and flicker PM noise, the edf for a total deviation estimate is the same as that for the overlapping ADEV with the number of  $X^2$  degrees of freedom increased by 2.

Stable32 supports the calculation of TOTAVEV from phase or frequency data with the Total option under the Sigma and Run Menus. The calculation is always done from phase data by the doubly reflected method that does not require endpoint matching. The Stable32 implementation follows the recipe described by D.A Howe and C.A Greenhall in Eq. (3) of reference 5 below.

• **References**

The following references trace the development of the TOTVAR statistic:

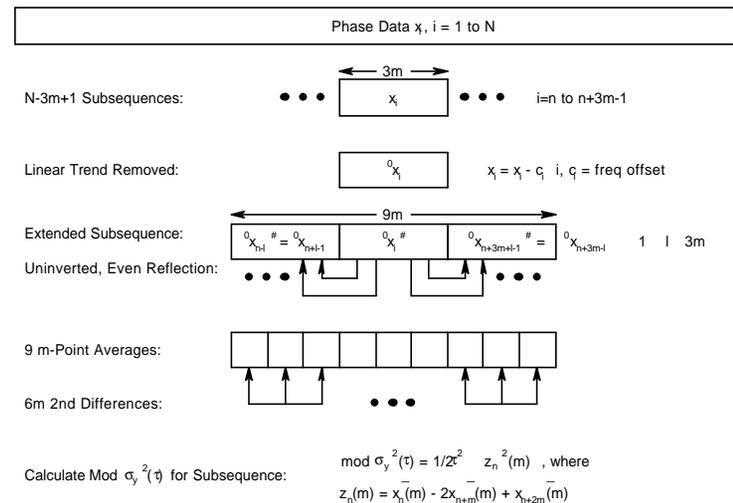
1. D.A. Howe, "An Extension of the Allan Variance with Increased Confidence at Long Term," Proc. 1995 IEEE Int. Freq. Cont. Symp., June 1995, pp. 321-329.
2. D.A. Howe and K.J. Lainson, "Simulation Study Using a New Type of Sample Variance," Proc. 1995 PTTI Meeting, Dec. 1995, pp. 279-290.
3. D.A. Howe and K.J. Lainson, "Effect of Drift on TOTALDEV," Proc. 1996 Intl. Freq. Cont. Symp., June 1996, pp. 883-889.
4. D.A. Howe, "Methods of Improving the Estimation of Long-term Frequency Variance," Proc. 1997 European Frequency and Time Forum, March 1997, pp. 91-99.
5. D.A. Howe and C.A. Greenhall, "Total Variance: a Progress Report on a New Frequency Stability Characterization," Proc. 1997 PTTI Meeting, Dec. 1997, pp. 39-48.
6. D.B. Percival and D.A. Howe, "Total Variance as an Exact Analysis of the Sample Variance", Proc. 1997 PTTI Meeting, Dec. 1997, pp.97-105.
7. C.A. Greenhall, D.A. Howe and D.B. Percival, "Total Variance, an Estimator of Long-Term Frequency Stability", August 11, 1998, (to be published - available on the Hamilton Technical Services web site).
8. D. Howe and T. Pepler, "Definitions of Total Estimators of Common Time-Domain Variances", Proc. 2001 Intl. Freq. Cont. Symp., June 2001, pp. 127-132.

## Total Modified Variance

• **Total Modified Variance**

The total modified variance, TOTMVAR, is another new statistic for the analysis of frequency stability. It is similar to the modified Allan variance, MVAR, and has the same expected value, but offers improved confidence at long averaging times. It uses the same phase averaging technique as MVAR to distinguish between white and flicker PM noise processes.

A TOTMVAR calculation, as described in reference [1] below, begins with an array of N phase data points (time deviates,  $x_i$ ) with sampling period  $\tau$ , which are to be analyzed at averaging time  $\tau=m\tau_0$ . TOTMVAR is computed from a set of N-3m+1 subsequences of 3m points. First, a linear trend (frequency offset) is removed from the subsequence by averaging the first and last halves of the subsequence and dividing by half the interval. Then the offset-removed subsequence is extended at both ends by uninverted, even reflection. Next the modified Allan variance is computed for these 9m points. Finally, these steps are repeated for each of the N-3m+1 subsequences, calculating TOTMVAR as their overall average. These steps are shown in the diagram below:



Computationally, the TOTMVAR process requires 3 nested loops:

- An outer summation over the N-3m+1 subsequences. The 3m-point subsequence is formed, its linear trend is removed, and it is extended at both ends by uninverted, even reflection to 9m points.

- An inner summation over the  $6m$  unique groups of  $m$ -point averages from which all possible fully overlapping  $2^{\text{nd}}$  differences are used to calculate MVAR.
- A loop within the inner summation to sum the phase averages for 3 sets of  $m$  points.

The final step is to scale the result according to the sampling period,  $\tau_0$ , averaging factor,  $m$ , and number of points,  $N$ . Overall, this can be expressed as:

$$\text{mod Tot var}(\tau) = \frac{1}{2(m\tau_0)^2(N-3m+1)} \sum_{n=1}^{N-3m+1} \left( \frac{1}{6m} \sum_{i=n-3m}^{n+3m-1} ({}^0Z_i^{\#}(m))^2 \right)$$

where the  ${}^0Z_i^{\#}(m)$  terms are the phase averages from the triply-extended subsequence, and the prefix  ${}^0$  denotes that the linear trend has been removed. At the largest possible averaging factor,  $m = N/3$ , the outer summation consists of only one term, but the inner summation has  $6m$  terms, thus providing a sizable number of estimates for the variance.

#### • References

The following reference describes the TOTMVAR statistic:

D.A. Howe and F. Vernotte, "Generalization of the Total Variance Approach to the Modified Allan Variance", Proc. 31<sup>st</sup> PTTI Meeting, December 1999, pp. 267-276.

## Hadamard Variance

### • Hadamard Variance

The Hadamard variance is a 3-sample variance that is similar to the 2-sample Allan variance. It examines the 2nd difference of the fractional frequencies, the 3rd difference of the phase variations. Because of this, the Hadamard variance, HVAR or  $H\sigma_y^2(\tau)$ , converges for the Flicker Walk FM ( $\alpha = -3$ ) and Random Run FM ( $\alpha = -4$ ) power-law noise types. It is also unaffected by linear frequency drift.

For frequency data, the non-overlapping Hadamard variance is defined as:

$$H\mathbf{s}_y^2(\mathbf{t}) = \frac{1}{6(M-2)} \sum_{i=1}^{M-2} [y_{i+2} - 2y_{i+1} + y_i]^2,$$

where  $y(i)$  is the  $i$ th of  $M$  fractional frequency values at averaging time  $\tau$ .

For phase data, the non-overlapping Hadamard variance is defined as:

$$H\mathbf{s}_y^2(\mathbf{t}) = \frac{1}{6\mathbf{t}^2(N-3)} \sum_{i=1}^{N-3} [x_{i+3} - 3x_{i+2} + 3x_{i+1} - x_i]^2,$$

where  $x(i)$  is the  $i$ th of  $N = M+1$  phase values at averaging time  $\tau$ .

Like the Allan variance, the Hadamard variance is usually expressed as its square-root, the Hadamard deviation, HDEV or  $H\sigma_y(\tau)$ .

Note: The Picinbono variance is a similar 3-sample statistic. It is identical to the Hadamard variance except for a factor of  $2/3$ .

## Overlapping Hadamard Variance

- **Overlapping Hadamard Variance**

In the same way that the overlapping Allan variance makes maximum use of a data set by forming all possible fully overlapping 2-sample pairs at each averaging time  $\tau$ , the overlapping Hadamard variance uses all 3-sample combinations. It can be estimated from a set of M frequency measurements for averaging time  $\tau = m\tau_0$ , where m is the averaging factor and  $\tau_0$  is the basic measurement interval, by the expression:

$$Hs_y^2(t) = \frac{1}{6m^2(M-3m+1)} \sum_{j=1}^{M-3m+1} \sum_{i=j}^{j+m-1} [y_{i+2m} - 2y_{i+m} + y_i]^2, \quad (2)$$

where  $y_i$  is the  $i$ th of M fractional frequency values at each measurement time.

In terms of phase data, the overlapping Hadamard variance can be estimated from a set of  $N = M+1$  time measurements as:

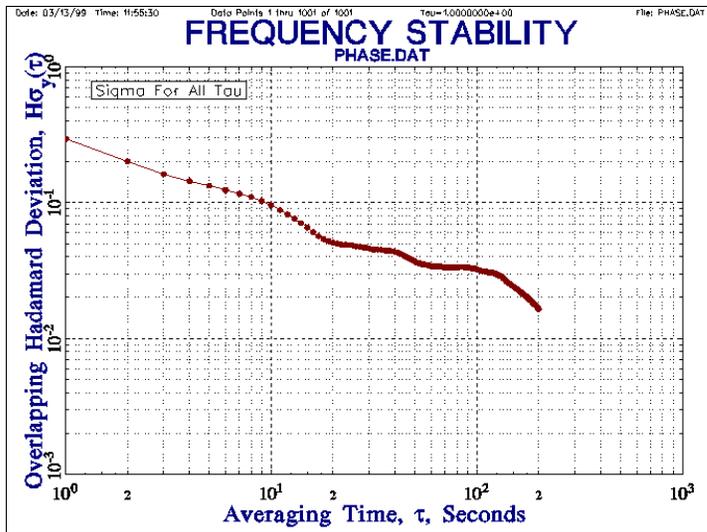
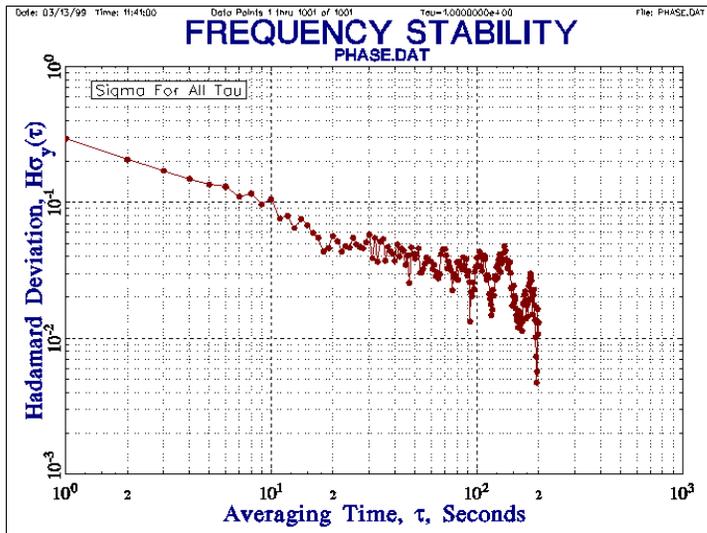
$$Hs_y^2(t) = \frac{1}{6(N-3m)t^2} \sum_{i=1}^{N-3m} [x_{i+3m} - 3x_{i+2m} + 3x_{i+m} - x_i]^2,$$

where  $x_i$  is the  $i$ th of  $N=M+1$  phase values at each measurement time.

Computation of the overlapping Hadamard variance is more efficient for phase data, where the averaging is accomplished by simply choosing the appropriate interval. For frequency data, an inner averaging loop over m frequency values is necessary. The result is usually expressed as the square root,  $H\sigma_y(\tau)$ , the Hadamard deviation, HDEV. The expected value of the overlapping statistic is the same as the normal one described above, but the confidence interval of the estimation is better. Even though all the additional overlapping differences are not statistically independent, they nevertheless increase the number of degrees of freedom and thus improve the confidence in the estimation. Analytical methods are available for calculating the number of degrees of freedom for an overlapping Allan variance estimation, and that same theory can be used to establish reasonable single or double-sided confidence intervals for an overlapping Hadamard variance estimate with a certain confidence factor, based on Chi-squared statistics:

$$\sigma_{\min}^2 = s^2 \cdot \frac{\chi^2(p, df)}{df} \quad \text{and} \quad \sigma_{\max}^2 = s^2 \cdot \frac{\chi^2(1-p, df)}{df},$$

where  $\chi^2$  is the Chi-square value for probability p and degrees of freedom df,  $s^2$  is the sample variance,  $\sigma^2$  is the true variance, and df is the # of degrees of freedom (not necessarily an integer). The df is determined by the number of data points and the noise type. The much smoother results it produces compared with the normal Hadamard statistic (as shown in the plots below) make it a more useful analytical tool.



• **References**

The following references apply to the overlapping Hadamard statistic:

1. T. Walter, "A Multi-Variance Analysis in the Time Domain", *Proc. 24<sup>th</sup> PTTI Meeting*, pp. 413-424, December 1992.
2. S. Hutsell, "Relating the Hadamard Variance to MCS Kalman Filter Clock Estimation", *Proc. 27<sup>th</sup> PTTI Meeting*, pp. 291-302, December 1995.

## Total Hadamard Variance

• **Total Hadamard Variance**

The total Hadamard variance, TOTHVAR, is total version of the Hadamard variance that rejects linear frequency drift and offers improved confidence at long averaging times. The algorithm for calculating the total Hadamard variance from a set of N fractional frequency values  $y_i$  at an averaging time  $\tau = m\tau_0$  is very similar to that used to calculate TOTMVAR from phase data:

1. Select a 3m-point subsequence  $y_n$  of the frequency data  $y_i$ ,  $i=n, n+3m, \dots, n+3m-1$ .
2. Remove the linear trend,  $c$ , (frequency drift) from the subsequence, obtaining  $y^*_i = y_i - c \cdot i$ , where  $c$  is found by averaging the 1st and last halves of the sequence divided by the interval.
3. Extend the subsequence at both ends by uninverted, even reflection to form the 9m-point subsequence  $\#y_i$  having the tripled range from  $i=n-3m$  to  $n+6m-1$ . The extended subsequence can be constructed by the expressions  $\#y_{n-k} = y^*_{n+k-1}$  and  $\#y_{n+3m+k-1} = y^*_{n+3m-k}$  for  $k=1$  to  $3m$ .
4. Calculate the Hadamard (3-sample) variance for this extended subsequence by forming the 6 m-point frequency averages and finding the square of their 2nd differences.
5. Find TOTHVAR as the simple average of the N-3m HVAR subestimates.

• **References**

The following references apply to the total Hadamard statistic:

1. D.A. Howe, et al., "A Total Estimator of the Hadamard Function Used For GPS Operations", *Proc. 32<sup>nd</sup> PTTI Meeting*, Nov. 29, 2000, pp. 255-268.
2. D.A. Howe, R.L. Beard, C.A. Greenhall, F. Vernotte, W.J. Riley and T.K. Pepler, "Enhancements to GPS Operations and Clock Evaluations Using a "Total" Hadamard Deviation", *IEEE Trans. UFFC*, Vol. 52, No. 8, August 2005.

## Thêo1

- **Thêo1**

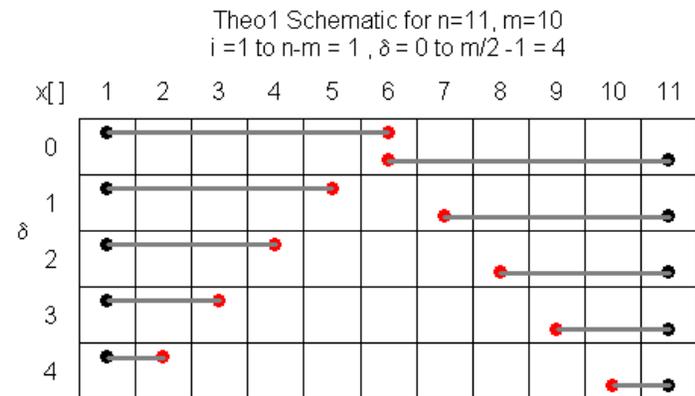
The Theoretical Variance #1, Thêo1, is a special-purpose statistic for the estimation of long-term frequency stability. It produces results similar to the Allan variance for averaging factors out to nearly the full data length.

The Thêo1 statistic is defined as follows:

$$\text{Theo1}(m, t_0, N) = \frac{1}{(N-m)(m t_0)^2 \cdot 0.75} \sum_{i=1}^{N-m} \sum_{d=0}^{m/2-1} \frac{1}{(m/2-d)} [(x_i - x_{i-d+m/2}) + (x_{i+m} - x_{i+d+m/2})]^2$$

for averaging factor  $m$  even, and  $10 \leq m \leq N-1$ , where  $N$  is the number of phase data points. It consists of  $(N - m)$  outer sums, and  $m/2$  inner sums. The Thêo1 deviation has the same power law noise tau dependence as the Allan deviation. It is an unbiased estimator of the Allan deviation for white FM noise, but requires bias corrections for the other power law noises.

An example of the sampling technique used by Thêo1 is shown in the figure below. This figure shows the  $m/2 = 5$  pairs of inner summation terms for  $N=11$  phase data points at the largest averaging factor  $m = 10$  for its single outer summation  $i = 1$ .



In general, there are  $m/2$  pairs of samples with spans ranging from  $m/2$  to 1, all weighted the same. The averaging time,  $\tau$ , associated with a Thêo1 value is  $\tau = 0.75 \cdot m \cdot \tau_0$ , where  $\tau_0$  is the measurement interval.

Stable32 offers the Thêo1 deviation alone and in combination with the Allan deviation, which provides stability data over the whole range of averaging times from the basic measurement interval to 75% of the record length.

- **NewThêo1**

The NewThêo1 algorithm of Reference [2] provides a method of automatic bias correction for a Thêo1 estimation based on the average ratio of the Allan and Thêo1 variances over a range of averaging factors from 10 to (N/30)-3, where N is the # of phase samples.

$$\text{NewThêo1}(m, \mathbf{t}_0, N) = \left[ \frac{1}{n+1} \sum_{i=0}^n \frac{\text{Avar}(m=9+3i, \mathbf{t}_0, N)}{\text{Thêo1}(m=12+4i, \mathbf{t}_0, N)} \right] \text{Thêo1}(m, \mathbf{t}_0, N),$$

where  $n = \left\lfloor \frac{N}{30} - 3 \right\rfloor$  and  $\lfloor \cdot \rfloor$  denotes the floor function.

- **ThêoBR and ThêoH**

The ThêoBR (Bias Removed) algorithm of Reference [3] provides an improved method of automatic bias correction. Stable32 uses this method in the Thêo1 Run function when automatic noise identification is selected.

$$\text{ThêoBR}(m, \mathbf{t}_0, N) = \left[ \frac{1}{n+1} \sum_{i=0}^n \frac{\text{Avar}(m=9+3i, \mathbf{t}_0, N)}{\text{Thêo1}(m=12+4i, \mathbf{t}_0, N)} \right] \text{Thêo1}(m, \mathbf{t}_0, N),$$

where  $n = \left\lfloor \frac{N}{6} - 3 \right\rfloor$  and  $\lfloor \cdot \rfloor$  denotes the floor function.

ThêoBR can be combined with the Allan variance to form the overlapping Allan and Thêo1 hybrid statistic ThêoH that is available in Stable32 as the ThêoH variance type.

$$\text{ThêoH}(m, \mathbf{t}_0, N) = \left\{ \begin{array}{l} \text{Avar}(m, \mathbf{t}_0, N) \text{ for } 1 \leq m < \frac{k}{\mathbf{t}_0} \\ \text{ThêoBR}(m, \mathbf{t}_0, N) \text{ for } \frac{k}{0.75\mathbf{t}_0} \leq m \leq N-1, m \text{ even} \end{array} \right\},$$

where  $k$  is the largest available  $\mathbf{t} \leq 20\% T$ .

- **Thêo1 Bias**

Reference [2] also includes an improved expression for the Thêo1 bias as a function of both noise type and averaging factor:

$$\text{Bias} = \text{AVAR}/\text{Thêo1} = a+b/m^c$$

where  $m$  is the averaging factor and the constants  $a$ ,  $b$  and  $c$  are given in the table below. Note that the effective tau for a Thêo1 estimation is  $\tau=0.75 \cdot m \cdot \tau_0$ , where  $\tau_0$  is the measurement interval.

Noise	Alpha	a	b	c
RW FM	-2	2.70	-1.53	0.85
F FM	-1	1.87	-1.05	0.79
W FM	0	1.00	0.00	0.00
F PM	1	0.14	0.82	0.30
W PM	2	0.09	0.74	0.40

This bias factor is applied to the Stable32 Thêo1 results.

- **Thêo1 EDF**

Empirical formulae for the Thêo1 EDF (as used in Stable32) will be found in Reference [1] below.

- **Thêo1 Confidence Intervals**

A detailed discussion of the confidence intervals for Thêo1 is given in Reference [4]. The Thêo1 confidence intervals are narrower and less skewed (more symmetric) than those based on chi-square, and Thêo1 therefore provides the highest confidence in estimating stability at large averaging factors. Stable32 sets the Thêo1 error bars based on a chi-square distribution and the Thêo1 equivalent  $X^2$  degrees of freedom as given in Reference [1].

- **References**

The following references apply to the Thêo1 statistic:

1. D.A. Howe and T.K. Pepler, " Very Long-Term Frequency Stability: Estimation Using a Special-Purpose Statistic", *Proceedings of the 2003 IEEE International Frequency Control Symposium*, pp. 233-238, May 2003.
2. T.N. Tasset and D.A. Howe, "A Practical Thêo1 Algorithm", Unpublished private communication, October 2003.
3. T.N. Tasset, "ThêoH", Unpublished private communication, July 2004.
4. T.N. Tasset, D.A. Howe and D.B. Percival, "Thêo1 Confidence Intervals", Unpublished paper, 2004.

**MTIE**

- MTIE**

The maximum time interval error, MTIE, is a measure of the maximum time error of a clock over a particular time interval. This statistic is very commonly used in the telecommunications industry. It is calculated by moving an n-point ( $n=\tau/\tau_0$ ) window through the phase (time error) data and finding the difference between the maximum and minimum values (range) at each window position. MTIE is the overall maximum of this time interval error over the entire data set:

$$MTIE(t) = \text{Max}_{1 \leq k \leq N-n} \{ \text{Max}_{k \leq i \leq k+n} (x_i) - \text{Min}_{k \leq i \leq k+n} (x_i) \}$$

where  $n=1,2,\dots, N-1$  and  $N= \#$  phase data points.

MTIE is a measure of the peak time deviation of a clock and is therefore very sensitive to a single extreme value, transient or outlier. The time required for an MTIE calculation increases geometrically with the averaging factor, n, and can become very long for large data sets (although faster algorithms are available – see Reference 4 below).

The relationship between MTIE and Allan variance statistics is not completely defined, but has been the subject of recent theoretical work (see References 2 and 3 below). Because of the peak nature of the MTIE statistic, it is necessary to express it in terms of a probability level,  $\beta$ , that a certain value is not exceeded.

For the case of white FM noise (important for passive atomic clocks such as the most common rubidium and cesium frequency standards), MTIE can be approximated by the relationship:

$$MTIE(\tau, \beta) = k_\beta \cdot \sqrt{(h_0 \cdot \tau)} = k_\beta \cdot \sqrt{2 \cdot \sigma_y(\tau) \cdot \tau}$$

where  $k_\beta$  is a constant determined by the probability level,  $\beta$ , as given in the table below, and  $h_0$  is the white FM power-law noise coefficient.

$\beta, \%$	$k_\beta$
95	1.77
90	1.59
80	1.39

- **References**

The following references apply to the MTIE statistic:

1. S. Bregni, "Clock Stability Characterization and Measurement in Telecommunications", *IEEE Trans. Instrum. Meas.*, Vol. 46, No. 6 pp. 1284-1294, Dec. 1997.
2. P. Tavella, A. Godone & S. Leschiutta, "The Range Covered by a Random Process and the New Definition of MTIE", *Proc. 28th PTI Meeting*, Dec. 1996, pp. 119-124.
3. P. Tavella and D. Meo, "The Range Covered by a Clock Error in the Case of White FM", *Proc. 30th PTI Meeting*, Dec. 1998 (to be published).
4. S. Bregni and S. Maccabruni, "Fast Computation of Maximum Time Interval Error by Binary Decomposition", *IEEE Trans. I&M*, Vol. 49, No. 6, Dec. 2000, pp. 1240-1244.

## TIE<sub>rms</sub>

- **TIE<sub>rms</sub>**

The rms time interval error, TIR<sub>rms</sub>, is another clock statistic commonly used by the telecommunications industry. TIE<sub>rms</sub> is defined by the expression:

$$TIE_{rms} = \sqrt{\frac{1}{N-n} \sum_{i=1}^{N-n} x_{i+n} - x_i^2}$$

where  $n=1,2,\dots, N-1$  and  $N=$  # phase data points.

For no frequency offset, TIE<sub>rms</sub> is approximately equal to the standard deviation of the fractional frequency fluctuations multiplied by the averaging time. It is therefore similar in behavior to TDEV, although the latter is better suited for divergent noise types.

- **References**

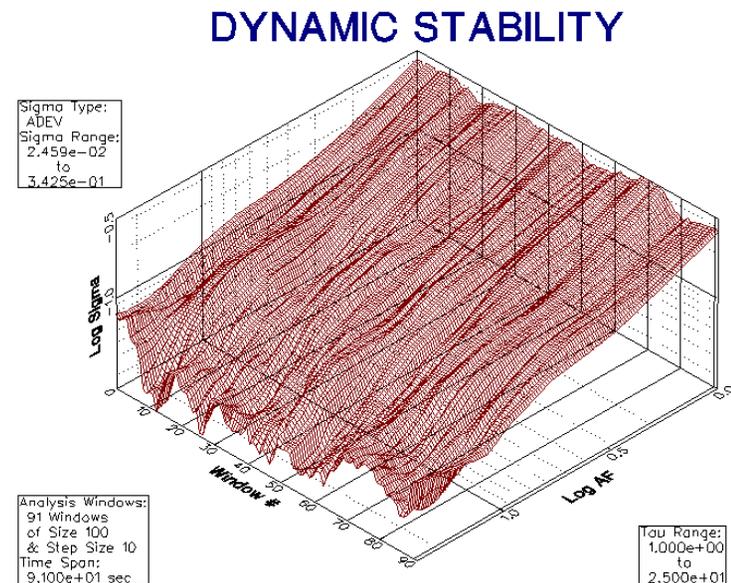
The following reference applies to the TIE<sub>rms</sub> statistic:

- S. Bregni, "Clock Stability Characterization and Measurement in Telecommunications", *IEEE Trans. Instrum. Meas.*, Vol. 46, No. 6, pp. 1284-1294, Dec. 1997.

## Dynamic Stability

- **Dynamic Stability**

A dynamic stability analysis uses a sequence of sliding time windows to perform an Allan (DAVAR) or Hadamard (DHVAR) analysis, thereby showing changes (nonstationarity) in clock behavior versus time. It is able to detect variations in clock stability (noise bursts, changes in noise level or type, etc.) that would be difficult to see in an ordinary overall stability analysis, and it can also show the portion of a run that has the best or worst stability. The results of a dynamic stability analysis are presented as a 3D surface plot of log sigma versus log tau (averaging factor) as a function of time (window #). Stable32 implements the DAVAR and DHVAR dynamic stability analyses in its Dynamic Stability function. An example of a dynamic stability plot is shown below:



- **References**

1. L. Galleani and P. Tavella, "The Characterization of Clock Behavior with the Dynamic Allan Variance", Proc. 2003 Joint FCS/EFTF Meeting, pp. 239-244.
2. L. Galleani and P. Tavella, "Tracking Nonstationarities in Clock Noises Using the Dynamic Allan Variance", Proc. 2005 Joint FCS/PTTI Meeting (to be published).

## Confidence Intervals

- **Confidence Intervals**

Stable32 uses several different methods for setting the confidence intervals and error bars in its stability analysis and plotting functions. The confidence interval of an Allan variance estimate depends not only on the Allan variance type (normal, overlapping, modified, or time) and nominal value, and on the averaging factor and number of data points, but also on the statistical confidence factor desired and the type of noise.

- **Simple Confidence Intervals**

The simplest confidence interval approximation, with no consideration of the noise type, sets the  $\pm 1\sigma$  (68%) error bars at  $\pm \sigma_y(\tau)/\sqrt{N}$ , where N is the number of frequency data points used to calculate the Allan deviation.

- **Kn Noise Factor**

A more accurate determination of this confidence interval can be made by considering the noise type, which can be estimated by the B1 bias function (the ratio of the standard variance to the Allan variance). That noise type is then be used to determine a multiplicative factor, Kn, to apply to the confidence interval. These are the error bars shown by the Stable32 Variance and Run functions for the normal Allan deviation.

<u>Noise Type</u>	<u>Kn</u>
Random Walk FM	0.75
Flicker FM	0.77
White FM	0.87
Flicker PM	0.99
White PM	0.99

- **Overlapping Samples**

Better statistical confidence can be obtained by making maximum use of the available data by forming all possible frequency differences for a given averaging time. Although these fully overlapping samples are not all statistically independent, they nevertheless help to improve the confidence of the resulting Allan deviation estimate. Furthermore, Chi-squared statistics can be applied to calculate single and double-sided confidence intervals at any desired confidence factor. These calculations, based on a determination of the number of degrees of freedom for the estimated noise type, are the confidence intervals used by the Stable32 Variance function for the overlapping Allan deviation. The same method is used by the Run function to show  $\pm 1\sigma$  error bars for its overlapping Allan deviation plot.

The error bars for the modified Allan and time variances are also determined by Chi-squared statistics, using the number of MVAR degrees of freedom for the particular noise type, averaging factor, and number of data points. During the Run function, noise type estimates are made at each averaging factor (except the last, where the noise type of the previous averaging factor is used).

• **Noise Identification**

It is necessary to identify the spectral characteristics of the dominant noise process in order to set the confidence interval, show the error bars and, where appropriate, apply a bias correction to the results of a time domain frequency stability analysis. Stable32 uses two methods for power law noise identification, based respectively on the lag 1 autocorrelation and the B1 bias factor. The former method is preferred, and is used when there are at least 30 analysis data points.

The lag 1 autocorrelation is an effective method for identifying power law noises based on the properties of discrete-time fractionally integrated noises having spectral densities of the form  $(2 \sin \pi f)^{-2\delta}$  [1]. For  $\delta < 1/2$ , the process is stationary and has a lag 1 autocorrelation equal to  $\rho_1 = \delta / (1-\delta)$  [6], and the noise type can therefore be estimated from  $\delta = r_1 / (1+r_1)$ . For frequency data, white PM noise has  $\rho_1 = -1/2$ , flicker PM noise has  $\rho_1 = -1/3$ , and white FM noise has  $\rho_1 = 0$ . For the more divergent noises, first differences of the data are taken until a stationary process is obtained as determined by the criterion  $\delta < 0.25$ . The noise identification method therefore uses  $p = \text{round}(\delta) - 2d$ , where  $\text{round}(\delta)$  is  $\delta$  rounded to the nearest integer and  $d$  is the number of times that the data is differenced to bring  $\delta$  down to  $< 0.25$ . If  $z$  is a  $\tau$ -average of frequency data  $y(t)$ , then  $\alpha = p$ ; if  $z$  is a  $\tau$ -sample of phase data  $x(t)$ , then  $\alpha = p + 2$ , where  $\alpha$  is the usual power law exponent  $f^\alpha$ , thereby determining the noise type at that averaging time. When there are a sufficient number of data points, this method has excellent discrimination for all common power law noises for both phase and frequency data, including difficult cases with mixed noises

The dominant power law noise type can also be estimated by comparing the ratio of the N-sample (standard) variance to the 2-sample (Allan) variance of the data (the B1 bias factor, see below) to the value expected of this ratio for the pure noise types (for the same averaging factor) [2]. This method of noise identification is reasonably effective in most cases. The main limitations are (1) its inability to distinguish between white and flicker PM noise, and (2) its limited precision at large averaging factors where there are few analysis points. The former limitation can be overcome by supplementing the B1 ratio test with one based on R(n), the ratio of the modified Allan variance to the normal Allan variance (see below). That technique is applied to members of the modified family of variances (MVAR, TVAR, and TOTMVAR). The second limitation is avoided by using the previous noise type estimate at the longest averaging time of an analysis run. One further limitation is that the R(n) ratio is not meaningful at a unity averaging factor. A description of the power law noise identification method used in Stable32 can be found in Reference [2].

• **AVAR EDF**

The equivalent # of  $X^2$  degrees of freedom (edf) for the fully overlapping Allan variance (AVAR) is determined by the following approximation formulae for each power law noise type.

Power Law Noise Type	AVAR edf, where N = # phase data points, m = averaging factor = $\tau/\tau_0$	
W PM	$\frac{N+1}{2} \frac{N-2m}{N-m}$	
F PM	$\exp \ln \frac{N-1}{2m} \ln \frac{2m+1}{4} \frac{N-1}{4}^{1/2}$	
W FM	$\frac{3}{2m} \frac{N-1}{N} - \frac{2}{4m^2+5} \frac{N-2}{N}$	
F FM	$\frac{2}{2.3N-4.9} \frac{N-2}{N}$	For m = 1
	$\frac{5N^2}{4mN+3m}$	For m > 1
RW FM	$\frac{N-2}{m} \frac{N-1}{N-3} - \frac{2-3m}{N-1} \frac{N-1+4m^2}{N-3}$	

• **MVAR EDF**

The edf for the modified Allan variance (MVAR) is determined by the same expression as the overlapping Hadamard variance (see below) with the arguments changed as follows (valid for  $-2 \leq \alpha \leq 2$ ):

MVAR and TVAR edf for N, m and  $\alpha =$  MVAR edf for N+1, m and  $\alpha-2$ .

• **TOTVAR EDF**

The edf for the total variance (TOTVAR) is given by the formula  $b(T/\tau) - c$ , where T is the length of the data record,  $\tau$  is the averaging time, and b & c are coefficients that depend on the noise type as shown in the following table:

Power Law Noise Type	TOTVAR edf coefficients	
	b	c
W FM	1.50	0
F FM	1.17	0.22
RW FM	0.93	0.36

• **TOTMVAR EDF**

The edf for the total modified variance (TOTMVAR) is given by the same formula  $b(T/\tau) - c$ , where T is the length of the data record,  $\tau$  is the averaging time, and b & c are coefficients that depend on the noise type as shown in the following table:

Power Law Noise Type	TOTMVAR edf coefficients	
	b	c
W PM	1.90	2.10
F PM	1.20	1.40
W FM	1.10	1.20
F FM	0.85	0.50
RW FM	0.75	0.31

• **HVAR EDF**

The edf for the fully overlapping Hadamard variance (HVAR) can be found by an algorithm developed by C.A. Greenhall based on its generalized autocovariance function. The HVAR edf is found either as a summation (for small m cases with a small number of terms) or from a limiting form for large m where  $1/edf = (1/p)(a0-a1/p)$  with the coefficients as follows:

Power Law Noise Type	HVAR edf coefficients	
	a0	a1
W FM	7/9	1/2
F FM	1.00	0.62
RW FM	31/30	17/28
FW FM	1.06	0.53
RR FM	1.30	0.54

• **TOTMVAR EDF**

The edf for the total modified variance (TOTMVAR) is given by the formula  $b(T/\tau) - c$ , where T is the length of the data record,  $\tau$  is the averaging time, and b & c are coefficients that depend on the noise type as shown in the following table:

Power Law Noise Type	TOTMVAR edf coefficients	
	b	c
W PM	1.90	2.10
F PM	1.20	1.40
W FM	1.10	1.20
F FM	0.85	0.50
RW FM	0.75	0.31

• **Thêo1 EDF**

See the Thêo1 section of this User Manual for a discussion of the Thêo1 edf and confidence intervals.

• **# Analysis Points**

The # in the Run stability table, and the # Analysis Pts in the detailed Sigma dialog box, is the number of analysis points (the # of 2nd or 3rd differences summed) in the sigma calculation. This number is used in determining the confidence intervals. Without gaps, the #s are equal to the following:

<u>Sigma Type</u>	#
Normal Allan	M/m -1
Overlapping Allan	M-2m+1 = N-2m
Modified & Time	M-3m+2 = N-3m+1
Total	M-m = N-m-1
Mod & Time Total	M-3m+2 = N-3m+1
Hadamard	M/m -2
Overlapping Hadamard	M-3m+1 = N-3m

where: N = # phase data points = M+1  
M = # frequency data points  
m = AF = averaging factor

For the normal Allan and Hadamard variances, the rounding down associated with successive averaging may reduce the #. With gaps, the # can depend on exactly where the gaps are. In all cases, the # is counted dynamically during the calculation as the statistical sums are accumulated.

• **References**

1. W.J. Riley and C.A. Greenhall, "Power Law Noise Identification Using the Lag 1 Autocorrelation", *Proc. 18<sup>th</sup> European Frequency and Time Forum*, April 2004.
2. D. Howe, R. Beard, C. Greenhall, F. Vernotte and W. Riley, "A Total Estimator of the Hadamard Function Used for GPS Operations", *Proceedings of the 32nd Annual Precise Time and Time Interval (PTTI) Systems and Applications Meeting*, pp. 255-268, November 2000.

## Bias Functions

- **Bias Functions**

Several bias functions are defined and used in the analysis of frequency stability. These bias functions are defined below. The Stable32 program uses the B1, the standard variance to Allan variance ratio, and R(n), the modified Allan variance to normal Allan variance ratio bias functions for the identification of noise types.

- **B1 Bias Function**

The B1 bias function is the ratio of the N-sample (standard) variance to the 2-sample (Allan) variance with dead time ratio  $r = T/\tau$ , where T = time between measurements,  $\tau$ =averaging time, and  $\mu$ =exponent of  $\tau$  in Allan variance for a certain power law noise process:

$$B1(N, r, \mu) = \sigma^2(N, T, \tau) / \sigma^2(2, T, \tau)$$

- **B2 Bias Function**

The B2 bias function is the ratio of the 2-sample (Allan) variance with dead time ratio  $r = T/\tau$  to the 2-sample (Allan) variance without dead time ( $r = 1$ ):

$$B2(r, \mu) = \sigma^2(2, T, \tau) / \sigma^2(2, \tau, \tau)$$

- **B3 Bias Function**

The B3 bias function is the ratio of the N-sample (standard) variance with dead time ratio  $r = T/\tau$  at multiples  $M = \tau/\tau_0$  of the basic averaging time  $\tau_0$  to the N-sample variance with the same dead time ratio at averaging time  $\tau$ :

$$B3(N, M, r, \mu) = \sigma^2(N, M, T, \tau) / \sigma^2(N, T, \tau)$$

- **R(n) Function**

The R(n) function is the ratio of the modified Allan variance to the normal Allan variance for  $n = \#$  phase data points. Note: R(n) is also a function of  $\alpha$ , the exponent of the power law noise type:

$$R(n) = \text{Mod } \sigma_y^2(\tau) / \sigma_y^2(\tau)$$

- **TOTVAR Bias Function**

The TOTVAR statistic is an unbiased estimator of the Allan variance for white and flicker PM noise, and for white FM noise. For flicker and random walk FM noise, TOTVAR is biased low as  $\tau$  becomes significant compared with the record length. The ratio of the expected value of TOTVAR to AVAR is given by the expression:

$$B(\text{TOTAL}) = 1 - a(\tau/T), \quad 0 < \tau \leq T/2$$

where  $a = 1/3 \cdot \ln 2 = 0.481$  for flicker FM noise,  $a = 3/4 = 0.750$  for random walk FM noise, and T is the record length. At the maximum allowable value of  $\tau = T/2$ , TOTVAR is biased low by about 24% for RW FM noise.

The Stable32 program applies this bias function automatically to correct the reported TOTVAR result.

- **TOTMVAR Bias Function**

The TOTMVAR statistic is a biased estimator of the modified Allan variance. The TOTMVAR bias factor (the ratio of the expected value of TOTMVAR to MVAR) depends on the noise type but is essentially independent of the averaging factor and # of data points, as shown in the following table:

Noise	Bias Factor	Note: The Stable32 program applies this bias factor automatically to correct the reported TOTMVAR result.
W PM	1.06	
F PM	1.17	
W FM	1.27	
F FM	1.30	
RW FM	1.31	

- **Théo1 Bias**

See the Théo1 section of this User Manual for a discussion of the Théo1 bias.

- **Summary of Stable32 EDF and Bias Functions**

The following table summarizes the use of the equivalent degrees of freedom, confidence intervals and bias corrections in the variance calculations of the Stable32 Sigma and Run functions:

Stable32 Noise ID, EDF, CI & Bias Methods				
Variance Type	Noise ID Method *	EDF Calc	CI/Error Bars	Bias Calc
Allan (AVAR)	Lag 1 ACF B1 ratio	N/A	$\sigma/\sqrt{N}$ and $K_n$ Can't set CI	N/A
Overlapping Allan (AVAR)	Lag 1 ACF B1 ratio	CDF [1]	$\chi^2$	N/A
Modified Allan (MVAR)	Lag 1 ACF B1 plus R(n) for $\alpha = 1,2$	HEDF [2, 11] w/ modified arguments	$\chi^2$	N/A
Time (TVAR)	Lag 1 ACF B1 plus R(n) for $\alpha = 1,2$	HEDF [2, 11] w/ modified arguments	$\chi^2$	N/A
Hadamard (HVAR)	Lag 1 ACF	N/A	Sigma: None Run: $\sigma/\sqrt{N}$ and $K_n$ Can't set CI	N/A
Overlapping Hadamard (HVAR)	Lag 1 ACF B1 ratio	HEDF [3]	$\chi^2$	N/A
Total Allan (TOTAVAR)	Lag 1 ACF B1 ratio	TOTEDF [4]	$\chi^2$	TOTVAR Bias [6]
Total Modified (TOTMVAR)	Lag 1 ACF B1 plus R(n) for $\alpha = 1,2$	TOTMVAR EDF for $m > 8$ , else EDF [5]	$\chi^2$	TOTMVAR Bias [7]
Total Time (TOTTVAR)	Lag 1 ACF B1 plus R(n) for $\alpha = 1,2$	TOTMVAR EDF for $m > 8$ , else EDF [5]	$\chi^2$	TOTMVAR Bias [7]
Théo1	Lag 1 ACF B1 plus R(n) for $\alpha = 1,2$	Théo1 EDF [17]	$\chi^2$	Théo1 Bias [17] or ThéoBR [18]
ThéoH	Lag 1 ACF B1 plus R(n) for $\alpha = 1,2$	Théo1 EDF [17]	$\chi^2$	ThéoBR [18]
TIE rms	N/A	N/A	None	N/A
MTIE	N/A	N/A	None	N/A

\* The Lag 1 ACF noise ID method is used when the # of analysis points is  $\geq 30$

- **References**

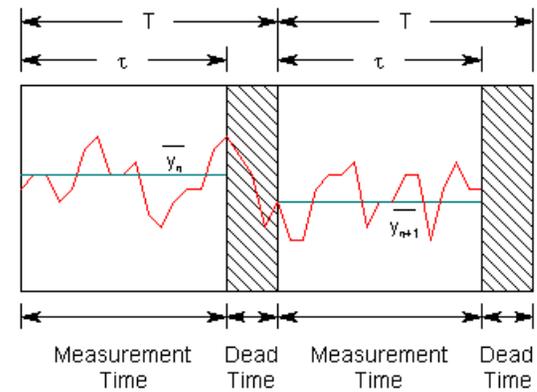
The following references apply to the Stable32 edf and bias functions:

1. "IEEE Standard Definitions of Physical Quantities for Fundamental Frequency and Time Metrology - Random Instabilities", *IEEE Std 1139-1999*, July 1999.
2. C.A. Greenhall, "Estimating the Modified Allan Variance", *Proc. IEEE 1995 Freq. Contrl. Symp.*, pp. 346-353, May 1995.
3. D.A. Howe & C.A. Greenhall, "Total Variance: A Progress Report on a New Frequency Stability Characterization", *Proc. 1997 PTTI Meeting*, pp. 39-48, December 1997.
4. C.A. Greenhall, private communication, May 1999.
5. D.A. Howe, private communication, March 2000.
6. D.A. Howe, "Total Variance Explained", *Proc. 1999 Joint Meeting of the European Freq. and Time Forum and the IEEE Freq. Contrl. Symp.*, pp. 1093-1099, April 1999.
7. D.A. Howe, private communication, March 2000.
8. C.A. Greenhall, "Recipes for Degrees of Freedom of Frequency Stability Estimators", *IEEE Trans. Instrum. Meas.*, Vol. 40, No. 6, pp. 994-999, December 1991.
9. D.A. Howe, "Methods of Improving the Estimation of Long-Term Frequency Variance", *Proc. 11th European Freq. and Time Forum*, pp. 91-99, March 1997.
10. J.A. Barnes and D.W. Allan, "Variances Based on Data with Dead Time Between the Measurements", *NIST Technical Note 1318*, 1990.
11. C.A. Greenhall, private communication, May 2000.
12. D. Howe, R. Beard, C. Greenhall, F. Vernotte and W. Riley, "A Total Estimator of the Hadamard Function Used for GPS Operations", *Proceedings of the 32nd Annual Precise Time and Time Interval (PTTI) Systems and Applications Meeting*, pp. 255-268, November 2000.
13. C. Greenhall and W. Riley, "Uncertainty of Stability Variances Based on Finite Differences", *Proc. 2003 PTTI Meeting*, December 2003.
14. D.A. Howe and T.K. Pepler, "Very Long-Term Frequency Stability: Estimation Using a Special-Purpose Statistic", *Proceedings of the 2003 IEEE International Frequency Control Symposium*, pp. 233-238, May 2003.
15. W.J. Riley and C.A. Greenhall, "Power Law Noise Identification Using the Lag 1 Autocorrelation", *Proc. 18th European Frequency and Time Forum (EFTF)*, April 2004, Session 6B.
16. D.A. Howe, R.L. Beard, C.A. Greenhall, F. Vernotte, W.J. Riley and T.K. Pepler, "Enhancements to GPS Operations and Clock Evaluations Using a "Total" Hadamard Deviation", *IEEE Trans. UFFC*, Vol. 52, No. 8, August 2005.
17. T.N. Tasset, D.A. Howe and D.B. Percival, "Théo1 Confidence Intervals", Unpublished paper, 2004.
18. T.N. Tasset, "ThéoH", Unpublished private communication, July 2004.

## Dead Time

- **Dead Time**

Dead time can occur in frequency measurements because of instrumentation delay between successive measurements, or because of a deliberate wait between measurements. It can have a significant effect on the results of a stability analysis, especially for the case of large dead time (e.g. frequency data taken for 100 seconds once per hour). Stable32 includes provisions for making dead time corrections in the Sigma and Run functions for normal and overlapping Allan deviations where the noise type is known. These corrections use the B2 and B3 bias functions to handle single and multiple (distributed) measurement dead time respectively. The dead time is entered as the ratio  $r=T/t$ , where  $t$  is the measurement time and  $T$  is the (longer) time between measurements ( $r=1$  for no dead time).



The corrections are applied by dividing the calculated sigma by the square root of the product of B2 and B3. These corrections are particularly important for non-white FM noise with a large dead time ratio. Restricting the dead time corrections to Allan deviations is a conservative approach based on the B2 and B3 definitions. These bias functions depend critically on the power law noise type. Requiring manual noise selection avoids the problem of noise identification for biased data having the wrong sigma-tau slope. Dead time correction is problematic for data having multiple noise types. In addition to introducing bias, measurement dead time reduces the confidence in the results, lowers the maximum allowable averaging factor, and prevents proper frequency to phase conversion. Moreover, no information is available about the behavior of the device under test during the dead time. It is recommended that these issues be avoided by making measurements with zero dead time.

- **References**

1. J.A. Barnes, "Tables of Bias Functions, B1 and B2, for Variances Based on Finite Samples of Processes with Power Law Spectral Densities", *NBS Technical Note 375*, January 1969.
1. J.A. Barnes and D.A. Allan, "Variances Based on Data with Dead Time Between the Measurements", *NIST Technical Note 1318*, 1990.

## Autocorrelation

- **Autocorrelation**

The theoretical autocorrelation function is defined as:

$$\rho_k = E\{(z_t - \mu)(z_{t+k} - \mu)\} / \sigma^2$$

where:  $E\{ \}$  is the expectation operator,  $z$  is the time series,  $m$  is its mean and  $s^2$  is its variance. A common estimate of the autocorrelation function is:

$$r_k = \frac{\frac{1}{N} \sum_{t=1}^{N-k} (z_t - \bar{z})(z_{t+k} - \bar{z})}{\frac{1}{N} \sum_{t=1}^N (z_t - \bar{z})^2}$$

where:

$$\bar{z} = \frac{1}{N} \sum_{t=1}^N z_t$$

and where the lags are  $k=0, 1, 2 \dots K$ , and  $K$  is  $\leq N-1$ .

An equivalent (and faster) estimate can be made for the summation in the numerator as the product of the Fourier transforms of the two terms, based on the fact that convolution in the time domain is equivalent to multiplication in the frequency domain.

- **Noise Identification**

The lag 1 autocorrelation can provide an estimate of the power law noise type, as described in Reference 5 below.

- **References**

1. W.H. Press, B.P. Flannery, S.A. Teukolsky and W.T. Vetterling, *Numerical Recipes in C*, Cambridge University Press, 1988, ISBN 0-521-35465-X, Section 13.2.
2. G. Box and G. Jenkins, *Time Series Analysis*, Holden-Day, Inc., 1976, ISBN 0-8162-1104-3.
3. D.B. Percival and A.T. Walden, *Spectral Analysis for Physical Applications*, Cambridge University Press, 1993, ISBN 0-521-43541-2.
4. C.A. Greenhall, "Another Power-Law Identifier That Uses Lag-1 Autocorrelation", private communication, August 28, 2003.
5. W.J. Riley and C.A. Greenhall, "Power Law Noise Identification Using the Lag 1 Autocorrelation", *Proc. 18th European Frequency and Time Forum (EFTF)*, April 2004, Session 6B.

## Frequency Domain Stability

- **Frequency Domain Stability**

Stable32 uses the following frequency domain measures of frequency stability:

<u>Formula</u>	<u>Units</u>	<u>Description</u>
$S_y(f)$	1/Hz	PSD of fractional frequency fluctuations
$S_x(f)$	sec <sup>2</sup> /Hz	PSD of time fluctuations
$S_\phi(f)$	rad <sup>2</sup> /Hz	PSD of phase fluctuations
$\mathcal{L}(f)$	dBc/Hz	SSB phase noise to carrier power ratio

where: PSD = Power Spectral Density  
SSB = Single Sideband  
dBc = Decibels with respect to carrier power

The most common frequency domain stability measure is  $\mathcal{L}(f)$ , dBc/Hz, the SSB phase noise to signal ratio in a 1 Hz BW as a function of sideband frequency,  $f$ . Other frequency domain stability measures are:  $S_y(f)$ , 1/Hz, the power spectral density of the fractional frequency fluctuations;  $S_x(f)$ , sec<sup>2</sup>/Hz, the power spectral density of the time fluctuations; and  $S_\phi(f)$ , rad<sup>2</sup>/Hz, the power spectral density of the phase fluctuations. The Stable32 program can plot any of these from the appropriate phase or frequency data.

## Noise Processes

- **Power Law Noise Processes**

The spectral characteristics of the power law noise processes commonly used to describe the performance of frequency sources are shown in the following table:

### Spectral Characteristics of Power Law Noise Processes

<u>Noise Type</u>	<u>a</u>	<u>b</u>	<u>m</u>	<u>m̄</u>	<u>h</u>
White PM	2	0	-2	-3	-1
Flicker PM	1	-1	-2	-2	0
White FM	0	-2	-1	-1	1
Flicker FM	-1	-3	0	0	2
Random Walk FM	-2	-4	1	1	3

- **Power Spectral Densities**

Four types of power spectral density (PSD) are commonly used to describe the stability of a frequency source:

#### PSD of Frequency Fluctuations $S_y(f)$

The power spectral density (PSD) of the fractional frequency fluctuations  $y(t)$  in units of 1/Hz is given by  $S_y(f) = h(\alpha) \cdot f^{-\alpha}$ , where  $f$  = sideband frequency, Hz.

#### PSD of Phase Fluctuations $S_\phi(f)$

The PSD of the phase fluctuations in units of  $\text{rad}^2/\text{Hz}$  is given by  $S_\phi(f) = (2\pi\nu_0)^2 \cdot S_x(f)$ , where  $\nu_0$  = carrier frequency, Hz.

#### PSD of Time Fluctuations $S_x(f)$

The PSD of the time fluctuations  $x(t)$  in units of  $\text{sec}^2/\text{Hz}$  is given by  $S_x(f) = h(\beta) \cdot f^{-\beta} = S_y(f)/(2\pi f)^2$ , where:  $\beta = \alpha - 2$ . The time fluctuations are related to the phase fluctuations by  $x(t) = \phi(t)/2\pi\nu_0$ . Both are commonly called "phase" to distinguish them from the independent time variable,  $t$ . Stable32 uses  $x(t)$  and  $y(t)$  arrays as its phase and frequency variables.

#### SSB Phase Noise $\mathcal{L}(f)$

The SSB phase noise in units of dBc/Hz is given by  $\mathcal{L}(f) = 10 \cdot \log[1/2 \cdot S_\phi(f)]$ . These are the most common engineering units to specify phase noise.

- **Variations**

Several types of variance are commonly used to describe the stability of a frequency source:

**Standard Variance  $S^2(t)$**

The N-sample or standard variance is not recommended as a general-purpose measure of frequency stability because it is not convergent for some types of noise commonly found in frequency sources. It is used primarily in the calculation of the B1 ratio for noise recognition.

**Allan Variance  $S^2_y(t)$**

The Allan variance is given by  $\sigma_y^2(\tau) = A(\mu) \cdot \tau^\mu$ , where  $\tau$  = averaging time, seconds. It is the most common measure of the fluctuations of the fractional frequency  $y(t) = [v(t) - v_0]/v_0 = \Delta v/v_0 = x'(t)$ . It may be calculated by the normal or overlapping method.

**Modified Allan Variance Mod  $S^2_y(t)$**

The modified Allan variance employs phase averaging and is given by Mod  $\sigma_y^2(\tau) = B(\mu) \cdot \tau^\mu$ , where  $\mu' = -\alpha - 1$ .

**Total Allan Variance  $S^2_{total}(t)$**

The total Allan variance is similar to the Allan variance but uses a doubly reflected method that improves its statistical confidence at large averaging factors.

**Total Modified Allan Variance Mod  $S^2_{total}(t)$**

The total modified Allan variance is similar to the modified Allan variance but uses an uninverted even reflection method that improves its statistical confidence at large averaging factors.

**Time Variance  $S^2_x(t)$**

The time variance is based on the modified Allan variance. It has units of seconds is given by  $\sigma_x^2(\tau) = C(\eta) \cdot \tau^\eta = (\tau^2/3) \cdot \text{Mod } \sigma_y^2(\tau)$ , sec, where  $\eta = \beta - 1$ .

**Total Time Variance  $S^2_x(t)$**

The total time variance in units of seconds is given by  $\sigma_{xtotal}^2(\tau) = C(\eta) \cdot \tau^\eta = (\tau^2/3) \cdot \text{Mod } \sigma_{total}^2(\tau)$ , sec, where  $\eta = \beta - 1$ .

**Hadamard Variance  $HS^2_y(t)$**

The 3-sample (Hadamard) variance is used primarily as an alternative to the Allan variance for a frequency source having linear frequency drift.

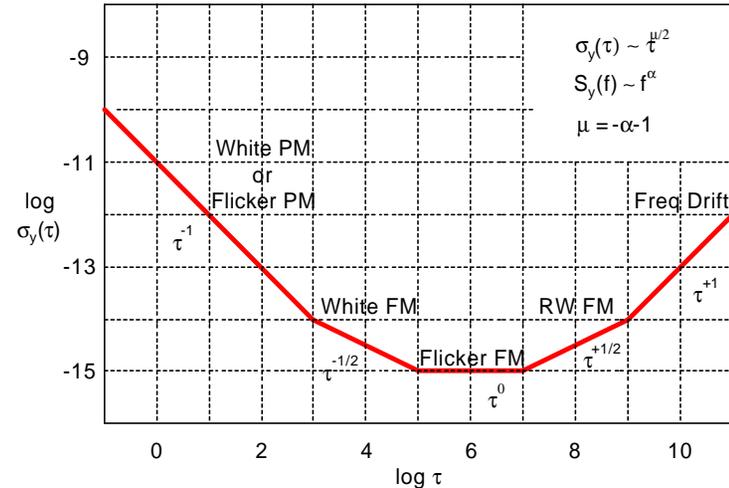
**Total Hadamard Variance  $HS^2_{total}(t)$**

The total Hadamard variance is similar to the Hadamard variance but uses a doubly reflected method that improves its statistical confidence at large averaging factors.

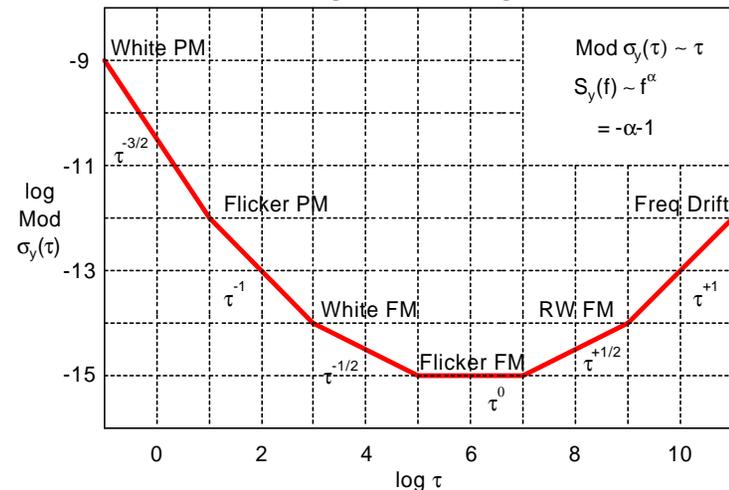
- **Sigma Tau Diagrams**

The following two diagrams show the relationship between the log  $\sigma$  versus log  $\tau$  slopes for various power law noise processes for the normal and modified Allan variances. Note that Mod  $\sigma_y(\tau)$  is able to distinguish between white and flicker PM noise.

**Sigma Tau Diagram**



**Mod Sigma Tau Diagram**



## Noise Spectra

- **Noise Spectra**

The random phase and frequency fluctuations of a frequency source can be modeled by power law spectral densities of the form:

$$S_y(f) = h(\alpha)f^\alpha$$

where:  $S_y(f)$  = one-sided power spectral density of the fractional frequency fluctuations, 1/Hz  
 $f$  = Fourier or sideband frequency, Hz  
 $h(\alpha)$  = intensity coefficient  
 $\alpha$  = exponent of the power law noise process.

The most commonly encountered noise spectra are:

White ( $f^0$ )	Random Walk ( $f^{-2}$ )
Flicker ( $f^{-1}$ )	Flicker Walk ( $f^{-3}$ )

For examples showing the appearance of these noise types, use the Play program to display the file NOISEPIX.TKF. Power law spectral models can be applied to both phase and frequency power spectral densities (PSD). Phase is the time integral of frequency, so the relationship between them is as  $1/f^2$ :

$$S_x(f) = S_y(f)/(2\pi f)^2$$

where:  $S_x(f)$  = PSD of the time fluctuations,  $\text{sec}^2/\text{Hz}$ .

Two other quantities are also commonly used to measure phase noise:

$S_\phi(f)$  = PSD of the phase fluctuations,  $\text{rad}^2/\text{Hz}$  and its logarithmic equivalent  $\mathcal{L}(f)$ ,  $\text{dBc}/\text{Hz}$ .

The relationship between these is:

$$S_\phi(f) = (2\pi v_0)^2 \cdot S_x(f)$$

$$\mathcal{L}(f) = 10 \cdot \log[1/2 \cdot S_\phi(f)]$$

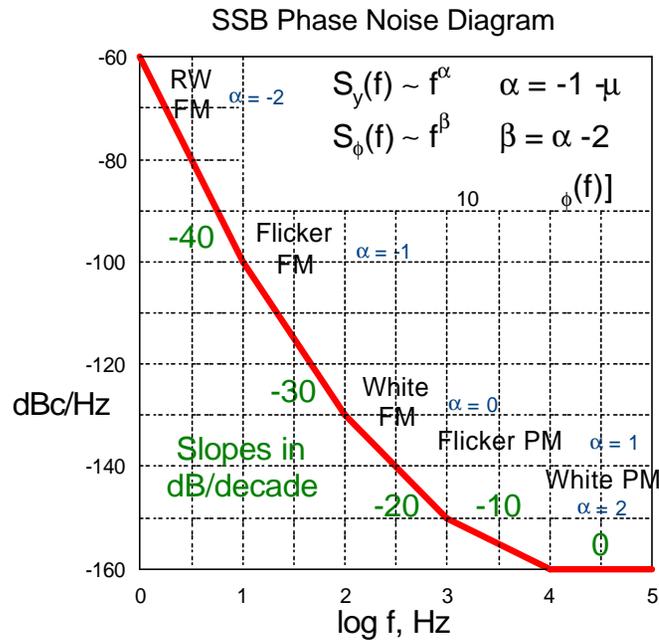
where  $v_0$  is the carrier frequency, Hz.

The power law exponent of the phase noise power spectral densities is  $\beta = \alpha - 2$ . These frequency-domain power law exponents are also related to the slopes of the following time domain stability measures:

Allan variance	$\sigma_y^2(\tau)$	$\mu = \alpha - 1, \alpha < 2$
Modified Allan variance	Mod $\sigma_y^2(\tau)$	$\mu' = \alpha - 1, \alpha < 3$
Time variance	$\sigma_x^2(\tau)$	$\eta = \alpha - 1, \alpha < 3$

- Phase Noise Diagram

The following diagram shows the slope of the SSB phase noise,  $\epsilon(f)$ , dBc/Hz versus  $\log f$ , Fourier frequency, Hz for various power law noise processes.



## Domain Conversions

- Time and Frequency Domains

The stability of a frequency source can be specified and measured in either the time or frequency domain. Examples of these stability measures are the Allan variance,  $\sigma_y^2(\tau)$ , in the time domain and the spectral density of the fractional frequency fluctuations,  $S_y(f)$ , in the frequency domain. Conversions between these domains may be made by an analytical method based on a power law spectral model for the noise processes involved. This method, the one used by the Stable32 Domain function, can be applied only when the dominant noise process decreases toward higher sideband frequencies. Otherwise a more fundamental method based on numerical integration must be used. The general time to frequency domain conversion is not unique because white and flicker phase noise have the same Allan variance  $\tau$  dependence. When performing any of these conversions, it is necessary to choose a reasonable range for  $\sigma$  and  $\tau$  in the domain being converted to. The main lobe of the  $\sigma_y(\tau)$  and Mod  $\sigma_y(\tau)$  responses occur at the Fourier frequency  $f = 1/2\tau$ .

- Domains Conversions

Domain conversions may be made for power law noise models by using the following conversion formulae:

Noise Type	$s_y^2(t)$	$S_y(f)$
RW FM	$A \cdot f^2 \cdot S_y(f) \cdot \tau^1$	$A^{-1} \cdot \tau^{-1} \cdot \sigma_y^2(\tau) \cdot f^{-2}$
F FM	$B \cdot f^1 \cdot S_y(f) \cdot \tau^0$	$B^{-1} \cdot \tau^0 \cdot \sigma_y^2(\tau) \cdot f^{-1}$
W FM	$C \cdot f^0 \cdot S_y(f) \cdot \tau^{-1}$	$C^{-1} \cdot \tau^1 \cdot \sigma_y^2(\tau) \cdot f^0$
F PM	$D \cdot f^{-1} \cdot S_y(f) \cdot \tau^{-2}$	$D^{-1} \cdot \tau^2 \cdot \sigma_y^2(\tau) \cdot f^1$
W PM	$E \cdot f^{-2} \cdot S_y(f) \cdot \tau^{-2}$	$E^{-1} \cdot \tau^2 \cdot \sigma_y^2(\tau) \cdot f^2$

where:

$$A = 4\pi^2/6$$

$$B = 2 \cdot \ln 2$$

$$C = 1/2$$

$$D = 1.038 + 3 \cdot \ln(2\pi f_h \tau_0) / 4\pi^2$$

$$E = 3f_h / 4\pi^2$$

and  $f_h$  is the upper cutoff frequency of the measuring system in Hz, and  $\tau_0$  is the basic measurement time. This factor applies only to white and flicker PM noise.

## File Functions

- **Purpose**

Stable32 offers several functions for setting, opening, adding, saving and reading data files. The Open function applies to phase, frequency, and archive data files, the Add and Save functions apply to phase and frequency data files, and the Read function is used to read a stability file.

- **Controls**

The Stable32 Open, Add and Save file functions use the standard Windows file open and save dialog boxes, except that the Open and Add dialog boxes contain the following additional controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
Data Type	Group	Select data type.
Phase	Radiobutton	Use phase data array.
Frequency	Radiobutton	Use frequency data array.

The Open function also has a radiobutton to select archive data, and an edit control to set the averaging factor to be used when reading the data. The Save function has a checkbox to save timetags along with the phase or frequency data.

- **Operation**

Select **Phase**, **Frequency** or **Archive** data type, enter the desired **Avg By** factor (if applicable), and perform the usual Windows file dialog box operations. Opening a data file with multiple columns will invoke the Multicolumn Data function. Attempting to open an oversize data file (as determined by the Max Data File Size configuration value) will invoke the Large Data File function, and attempting to add excessive data will be trapped with an error message. The results of an Open or Add operation will be displayed in a File Opened box if this feature is activated with the Show File Opened option of the Configure function.

## FileNames Function

- **Purpose**

Use the FileNames function to set the names of the various Stable32 data and plot files. It is good practice to preset all the relevant filenames within this function before beginning an analysis.



- **Controls**

The FileNames dialog box contains the following controls:

<b>Control</b>	<b>Type</b>	<b>Description</b>
Plot Filenames	Group Box	Controls to set plot filenames.
<u>P</u> hase	Edit	Enter phase plot filename.
<u>F</u> req	Edit	Enter frequency plot filename.
<u>S</u> igma	Edit	Enter sigma plot filename.
<u>H</u> isto	Edit	Enter histogram plot filename.
<u>S</u> pectrum	Edit	Enter power spectrum plot filename.
<u>A</u> CF (1)	Edit	Enter autocorrelation plot filename.
<u>D</u> AVAR	Edit	Enter dynamic stability plot filename.
Data Filenames	Group Box	Controls to set data filenames.
<u>P</u> hase	Edit	Enter phase data filename.
<u>F</u> req	Edit	Enter frequency data filename.
<u>S</u> tability	Edit	Enter stability data filename.
<u>P</u> SD	Edit	Enter power spectrum data filename.
<u>A</u> CF (2)	Edit	Enter autocorrelation data filename.
<u>P</u> ath	Edit	Enter the filename path.
<u>O</u> K	Pushbutton	Apply filenames & close.
<u>S</u> et	Pushbutton	Set all filenames to new path.
<u>R</u> eset	Pushbutton	Reset all filenames to default path.

Cancel      Pushbutton    Abort Filenames function & close.  
Help        Pushbutton    Invoke the Filenames help topic.

- **Operation**

Enter the path for the filenames and press the **Set** button. Then enter the individual plot and data filenames. Press **OK** to apply the new filenames, **Reset** to use the default path, or **Cancel** to restore the original filenames.

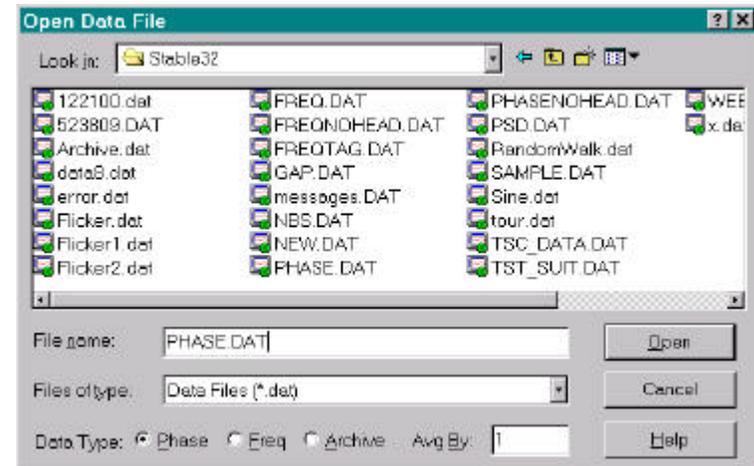
- **Note**

A valid filename cannot be blank or contain the characters ? " / \ < > \* | or :.

## Open Function

- **Purpose**

Use the Open function to read a phase or frequency data file from disk.



- **Controls**

The Open dialog box contains the standard Windows file open controls plus the following custom controls:

<u>C</u> ontrol	<u>T</u> ype	<u>D</u> escription
Data Type	Group Box	Controls to choose the data type.
<u>P</u> hase	Radiobutton	Read data into the phase array.
<u>F</u> requency	Radiobutton	Read data into the frequency array.
<u>A</u> rchive	Radiobutton	Read TSC archive file.
Avg <u>B</u> y	Edit	Averaging factor to apply when reading data.

- **Operation**

Select **Phase**, **Frequency** or **Archive** data type, and perform the usual file dialog box operations. Opening a data file with multiple columns will invoke the Multicolumn Data function. The data may be averaged as it is read by entering a non-unity **Avg By** value. Attempting to open an oversize data file (as set by the Max Data File Size configuration parameter) will invoke the Large Data File function. The results of an Open operation will be displayed in a File Opened box if this feature is activated with the Show File Opened option of the Configure function.

- **Filename Extensions**

The Open function supports \*.DAT (general data), \*.PHD (phase data), \*.FRD (frequency), and \*.\* (all) filename extensions. Selecting \*.PHD or \*.FRD file types automatically selects the corresponding data type.

- **Notes**

The data are read twice, first to determine their size and layout, and then to read the desired portion, averaging and columns. Commas are converted to spaces, and the resulting whitespace (spaces and tabs) is used to divide the lines into columns. The data are scanned by lines using an array of 8 "%le" fields. Non-numeric input characters are ignored, as are characters in excess of 8 fields. Using a non-unity averaging factor does not shorten the file reading process, but will make the subsequent analysis faster. Several configuration options can be set to alter the behavior of the file reading operation. See the Configuration section of this User Manual for more information about these settings.

**Configuration Item**

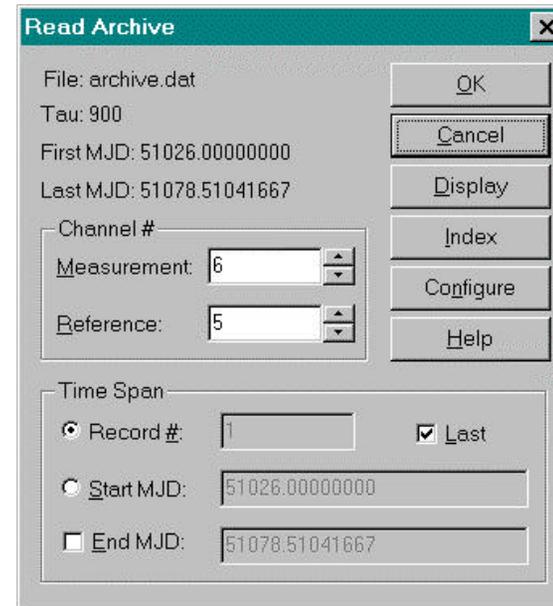
**Description**

Show File Opened	Show data plot and other information when file opened.
Status Plots	Show mini data plots in status bar.
Max Data FileSize	Set maximum data file size to read without averaging.
# Non-Numerics to Skip	Set # non-numeric characters to skip at beginning of each row of data.
Use Comma as Decimal Point	Use comma instead of decimal point for input data.
Warn for Zero Freq Data	Display warning if two adjacent phase data points are equal.

## Read Archive Function

- **Purpose**

Use the Read Archive function to extract clock data from a TSC archive file. This function will read the information produced by a Timing Solutions Corporation clock measuring system and convert it into time-tagged phase and frequency data for analysis by the Stable32 program. Controls are available for selecting the desired data file, measurement channels and time span for analysis.



- **Controls**

The Read Archive dialog box contains the following controls:

<b><u>Control</u></b>	<b><u>Type</u></b>	<b><u>Description</u></b>
File	Text	Archive filename.
Tau	Text	Measurement interval.
First MJD	Text	First MJD in archive file.
Last MJD	Text	Last MJD in archive file.
Channel #	Group	Channel # controls.
Measurement	Edit	Enter measurement channel #.
Reference	Edit	Enter reference channel #. (0=LO reference).
Time Span	Group	Time span controls.
Record #	Radiobutton	Select record # time span.
Record #	Edit	Enter record # to read.

<u>L</u> ast	Checkbox	Select last record.
<u>S</u> tart MJD	Radiobutton	Select MJD time span.
Start MJD	Edit	Enter start MJD.
<u>E</u> nd MJD	Checkbox	Select optional end MJD (instead of all data).
End MJD	Edit	Enter optional end MJD.
<u>O</u> K	Pushbutton	Perform Read Archive & close.
<u>C</u> ancel	Pushbutton	Abort Read Archive function & close
<u>D</u> isplay	Pushbutton	Display the archive file.
<u>I</u> ndex	Pushbutton	Show an index of the archive file.
<u>C</u> onfigure	Pushbutton	Configure the Read Archive function.
<u>H</u> elp	Pushbutton	Invoke the Read Archive help topic.

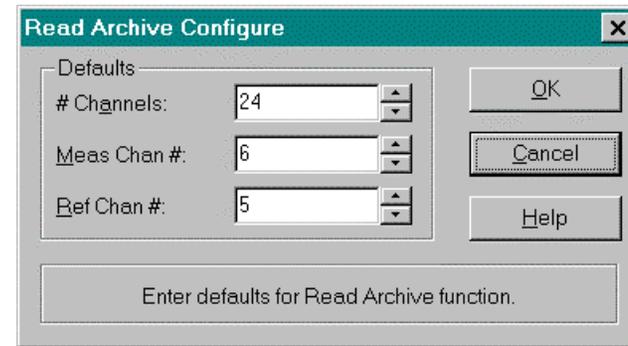
- **Operation**

Choose the desired TSC archive file with the Archive option of the Open function. Then observe the resulting **File**, **Tau** and **Start MJD**. Use the **Display** button to display the contents of the archive file, or the **Index** button to show a listing of all clock transactions in the archive file. Enter the desired **Measurement** and **Reference** channel #s that correspond to the clock to be analyzed. Entering a **Reference** channel # of 0 will select the offset LO as the reference. Entering a negative **Measurement** channel # will cause the sense of the measurement and reference channels to be reversed (this is useful in the case of a LO that is above rather than the below the nominal frequency as is normal). Reversal of the **Measurement** and **Reference** channel #s will also accomplish this, but the ID information displayed after processing the data will be that of the other channel. Then select either the **Record #** or **Start MJD** button as the method for defining the time span for the data. If **Record #** is selected, either enter the desired **Record #** or check the **Last** record checkbox. If **Start MJD** is selected as the time span method, enter the **Start MJD** value in the corresponding edit box. This will extract the data for the selected clock from that time to the end of the file. If it is desired to also set the **End MJD**, then check the **End MJD** checkbox and enter that value into its corresponding edit box. Then press **OK** to extract the clock data, or **Cancel** to abort the Read Archive function. The **Configure** button brings up another dialog box that allows setting certain defaults for the Read Archive function.

## Read Archive Configure

- **Purpose**

Use the Read Archive Configure dialog box to set defaults for the Read Archive function.



- **Controls**

The Read Archive Configure dialog box contains the following controls:

<u>C</u> ontrol	<u>T</u> ype	<u>D</u> escription
Defaults	Group	Read Archive default values.
# Channels	Edit	Set the maximum # of channels.
<u>M</u> meas Chan #	Edit	Set the default measurement channel.
<u>R</u> ef Chan #	Edit	Set the default reference channel
<u>O</u> K	Pushbutton	Perform the Read Archive Configure function & close dialog box.
<u>C</u> ancel	Pushbutton	Abort the Read Archive Configure function & close dialog box.
<u>H</u> elp	Pushbutton	Invoke the Read Archive Configure help topic.

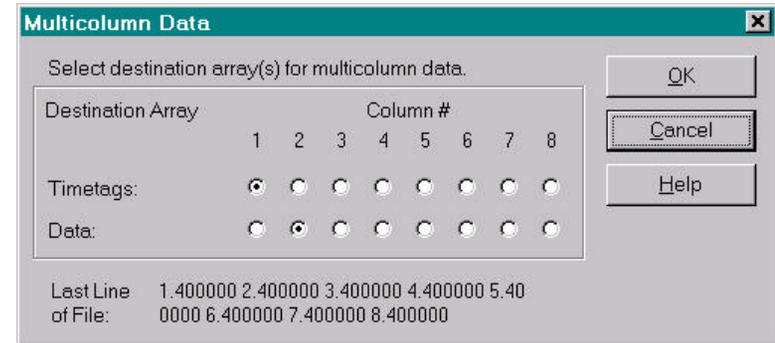
- **Operation**

Use the **# Channels**, **Meas Ch #** and **Ref Chan #** edit controls to set default values for the Read Archive function. Then press **OK** to accept or **Cancel** to reject these values. The new # of channels will take effect immediately in the Read Archive function.

## Multicolumn Data

- **Purpose**

The Multicolumn Data function supports the reading of data files with multiple columns. It is a sub-function of the Open and Add functions. Use the Multicolumn Data function to select the desired columns of phase or frequency and (optional) timetag data. This dialog box opens automatically during an Open or Add operation whenever multicolumn data is detected. The contents of the last line of the data file are displayed at the bottom of the dialog box to assist in selecting the desired timetags and data column.



- **Controls**

The Multicolumn Data dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
Timetags	Radiobutton	Choose column of timetag data to be read.
Data	Radiobutton	Choose column of phase or frequency data to be read.
1st Line of File	Text	Display of the 1st line of the data file.
<u>OK</u>	Pushbutton	Perform the read operation & close dialog boxes.
<u>Cancel</u>	Pushbutton	Abort the read function & close dialog boxes.
<u>Help</u>	Pushbutton	Invoke Multicolumn Read help topic.

- **Operation**

Select the desired column for the (optional) Timetags and Data, then press OK to read the data or Cancel to abort.

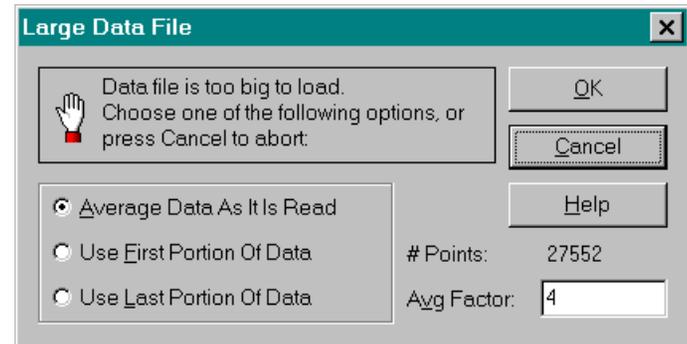
- **Limits**

Up to 8 columns may be processed from the data file.

## Large Data File

- **Purpose**

The Large Data File function supports the reading of large data files. It is a sub-function of the Open function. Use the Large Data File function to select the desired option for handling a large data file. This dialog box opens automatically during an Open operation whenever such a data file is detected. The user then has the options of (1) averaging the data as it is read, (2) using only the first portion of the data, (3) using only the last portion of the data.



- **Controls**

The Large Data File dialog box contains the following controls:

<u>C</u> ontrol	<u>T</u> ype	<u>D</u> escription
<u>A</u> verage	Radiobutton	Average the data as it is read.
<u>F</u> irst	Radiobutton	Use the first portion of the data.
<u>L</u> ast	Radiobutton	Use the last portion of the data.
# Points	Text	Display the # of points in the data file.
<u>A</u> vg Factor	Edit	Enter the averaging factor to be used to read the data file.
<u>O</u> K	Pushbutton	Accept selected option & continue with Open.
<u>C</u> ancel	Pushbutton	Abort the Open operation.
<u>H</u> elp	Pushbutton	Invoke the Large Data File help topic.

- **Operation**

Choose one of the **Average**, **First** or **Last** options and press **OK**, or press **Cancel** to abort the Open operation. With the **Average** option, the minimum **Avg Factor** required to read the entire data file is displayed. It may be increased if another (larger) averaging factor is desired. The **Avg Factor** is always 1 for the **First** and **Last** options.

- **Averaging**

Averaging of phase data is accomplished by simply eliminating the intermediate data points, while averaging of frequency data uses the mean value of the points being averaged. In either case, the averaging factor is displayed as the default  $\tau$  value.

- **Limits**

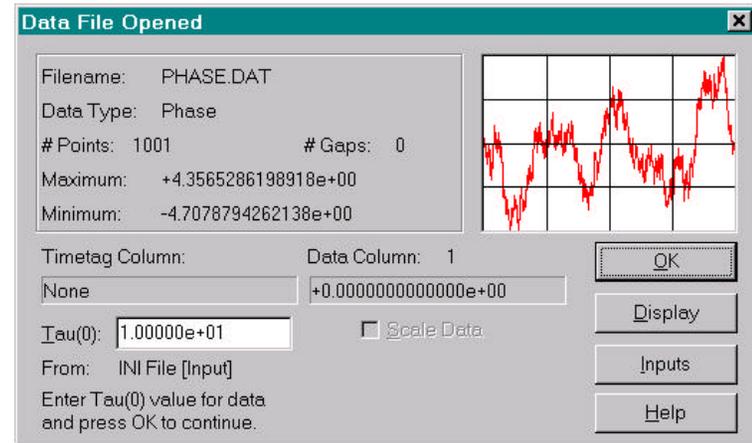
The maximum size data file that can be read is set by the Max Data File Size item in the Configure function. The purpose of this setting is to automatically activate the data averaging dialog box when a large data file is read. Set this parameter to a value appropriate to your computer's speed and RAM.

With the Average option, there is essentially no limit to the size of the data file that can be read.

## Data File Opened

- **Purpose**

The Data File Opened dialog box confirms the opening of a phase or frequency data file. It displays the name and size of the file, the # of gaps, and the maximum and minimum values of the data. This dialog box also shows the first line of the selected file column(s), and a simple plot of the data.



- **Controls**

Besides displaying information about the data file that has just been opened, the Data File Opened dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
<u>T</u> au	Edit	Enter Tau value for data.
<u>O</u> K	Pushbutton	Accept tau value and close dialog box.
<u>D</u> isplay	Pushbutton	Display data as text in Notepad.
<u>I</u> nputs	Pushbutton	Open Inputs dialog box.
<u>H</u> elp	Pushbutton	Invoke the Data File Opened help topic.

- **Operation**

Observe the properties of the data file that has been opened, enter the correct **Tau(0)** value, and then press **OK** to continue.

- **Tau**

The Tau(0) value must be set to correspond to the basic measurement interval of the data (in seconds) to obtain correct analysis results. It is preset to a value determined by the following order of precedence: (1) a command line tau entry, (2) a data file header tau entry, (3) a Stable32.ini [Inputs] value, (4) the first timetag interval, and (5) the default=1.0.

- **Data Scaling**

The data scaling option uses phase or frequency data addend and multiplier values from the [Inputs] section of the Stable32.ini configuration file. Those values can be entered manually by editing the configuration file with a text editor (such as Notepad), or by pressing the Inputs button. They default to 0 and 1 respectively, and, in that case, the scaling option is disabled.

- **Timetag Scaling**

The timetag scaling option uses the tag scale and tag offset values from the [Inputs] section of the Stable32.ini configuration file. Those values can be entered manually by editing the configuration file with a text editor (such as Notepad), or by pressing the Inputs button. They default to 1 and 0 respectively. Timetag scaling is done as the data file is read.

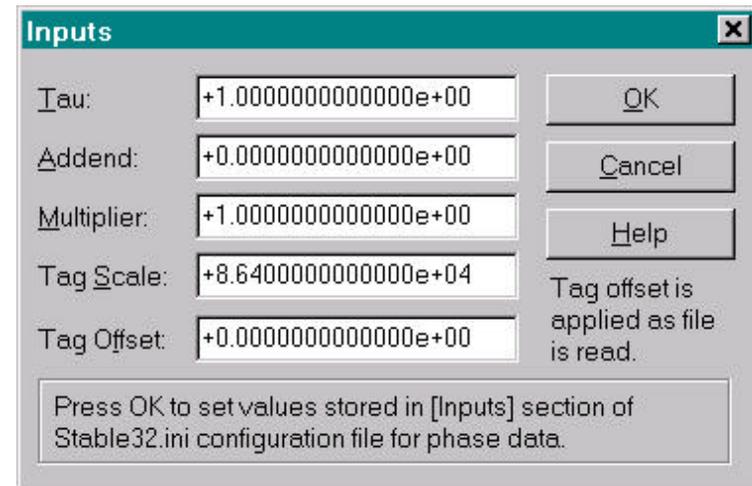
## Inputs

- **Purpose**

Use the Inputs function to set the tau, data addend and multiplier values, and the timetag offset and scale factors, for the current data type.

- **Controls**

The Inputs dialog box contains the following controls:



<u>Control</u>	<u>Type</u>	<u>Description</u>
<u>T</u> au	Edit	Enter the tau value.
<u>A</u> ddend	Edit	Enter the addend scaling value.
<u>M</u> ultiplier	Edit	Enter the multiplier scaling value.
<u>T</u> ag <u>S</u> cale	Edit	Enter the timetag scale multiplier.
<u>T</u> ag <u>O</u> ffset	Edit	Enter the timetag offset value.
<u>O</u> K	Pushbutton	Store values & close dialog box.
<u>C</u> ancel	Pushbutton	Abort & close dialog box.
<u>H</u> elp	Pushbutton	Invoke this help topic.

- **Operation**

Invoking the Inputs dialog box displays the current **Tau**, **Addend**, **Multiplier**, **Tag Scale** and **Tag Offset** values from the [Inputs] section of the Stable32.ini configuration file for the current phase or frequency data type. These values may be edited and saved by using the corresponding edit control(s) and pressing **OK**, or any changes made may be aborted by pressing the Cancel button.

- **inputs Values**

There are separate Tau, Addend, Multiplier, Tag Scale and Tag Offset inputs values for phase and frequency data. The Tau value should be set to the measurement interval of the data in seconds. The Multiplier value

should be set to scale the data for analysis (generally to seconds and fractional frequency). The Addend value is then added to the scaled values.

- **Tag Scale**

Use the tag scale multiplier to convert the timetag units to seconds. For example, if the timetags are MJD in whole and fractional days, the Tag Scale value should be set to 86400. The tag scale does not affect the timetag itself, but only its interpretation as a tau value.

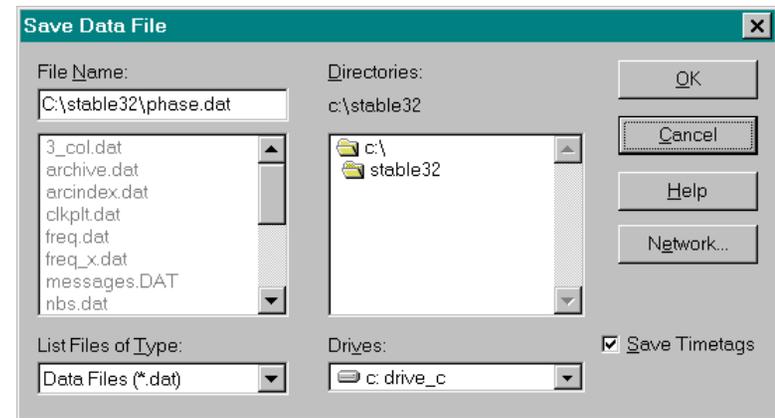
- **Tag Offset**

Use the tag offset to make time zone corrections or to convert general timetag dates to MJD. For example, if the timetags are the default dates used by Microsoft Excel® that start at zero at midnight on December 31, 1899, the Tag Offset value should be set to 15018 to convert them to MJD. Note: Excel incorrectly treats 1900 as a leap year. The tag offset is applied as the timetags are read.

## Save Function

- **Purpose**

Use the Save function to save phase or frequency data to disk.



- **Controls**

The Save dialog box contains the standard Windows file save controls, plus a Save Timetags checkbox to choose whether or not to also save the timetags.

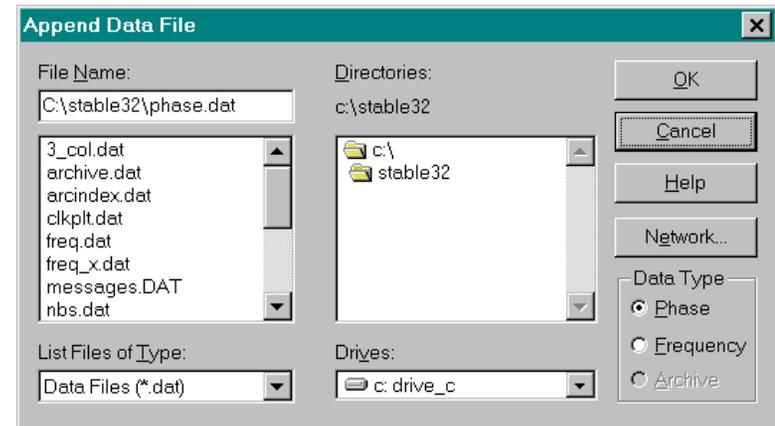
- **Operation**

Perform the usual Windows file save dialog box operations.

## Add Function

- **Purpose**

Use the Add function to append phase or frequency data from disk to the end of the current data in memory.



- **Controls**

The Add dialog box contains the standard Windows file open controls plus the following custom controls:

<u>C</u> ontrol	<u>T</u> ype	<u>D</u> escription
Data Type	Group Box	Controls to choose the data type.
<u>P</u> hase	Radiobutton	Read additional data into the phase array.
<u>F</u> requency	Radiobutton	Read additional data into the frequency array.

- **Operation**

Select **Phase** or **Frequency** data type, and perform the usual Windows file dialog box operations. Attempting to add excessive data will be trapped with an error message. The results of an Add operation will be displayed in a File Opened box if this feature is activated with the Show File Opened option of the Configure function.

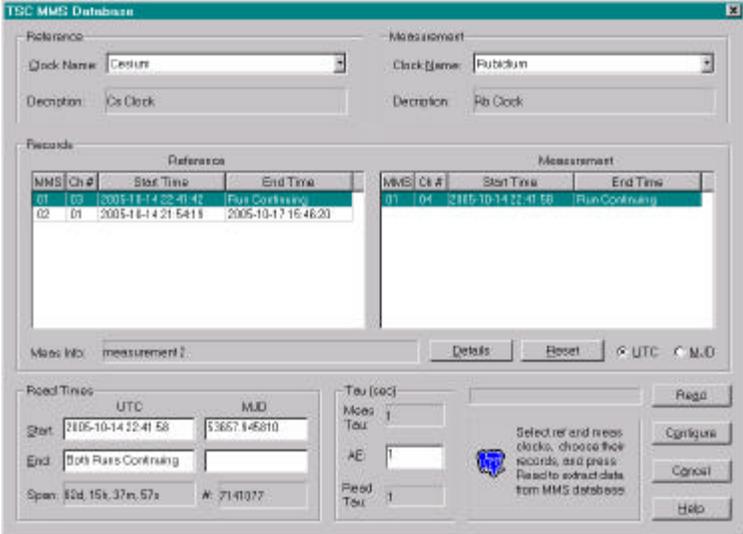
- **Caution**

Overlength data files cannot be added.

# Database Function

- Purpose**

Use the Database function to select, read and save clock data from a Timing Solutions Corporation (TSC) Multichannel Measuring System (MMS) database.



- Controls**

The Database function main dialog box contains the following controls:

<u>Dialog Box</u>	<u>ALT</u>	<u>Accel</u>	<u>Control</u>	<u>Control</u>
<u>Control</u>	<u>Key</u>	<u>Key</u>	<u>Type</u>	<u>Description</u>
Reference			Group Box	Reference clock info
Clock Name	C		Combo	Shows list of clocks
Description			Text	Shows description of selected reference clock
Measurement			Group Box	Measurement clock info
Clock Name	N		Combo	Shows list of clocks
Description			Text	Show description of selected measurement clock
Records			Group Box	Clock measurement records
Reference			Data Table	List of ref clock records
Measurement			Data Table	List of meas clock records
Meas Info			Text	Information about measurement record
Details	D		Pushbutton	Open Details box with database info
Reset	R		Pushbutton	Reset clock & record lists

UTC	U	Radio button	Show date/time as timestamp	
MJD	M	Radio button	Show date/time as MJD	
Read Times		Group Box	Start & stop times of data to be read	
Start UTC	S	Edit	Read start date/time (or message)	
Start MJD		Edit	Read start MJD	
End UTC	E	Edit	Read end date/time (or message)	
End MJD		Edit	Read end MJD	
Span		Text	Read span (days, hrs, mins, secs)	
#		Text	# data points in read span	
Tau (sec)		Group Box	Measurement & sampling interval	
Meas Tau		Text	MMS measurement interval	
AF	F	Edit	Averaging factor applied during read	
Read Tau		Text	Sampling interval of read data	
(No Name)		Progress Bar	Read progress indication	
(No Name)		Text	Program instructions & messages	
Read	A	Pushbutton	Read data from database Abort button during read	
Configure	O	Pushbutton	Open Configure dialog box	
Stable32	T	Pushbutton	Launch Stable32	
Close	L	Esc	Pushbutton	Close program
Help	H	F1	Pushbutton	Invoke this help topic

In addition, the right & left arrow keys will move the focus up and down through the pushbuttons, and the spacebar can be used to press them.

### • Operation

Begin by selecting the **Reference** and **Measurement** clocks, which will display their descriptions and load their measurement records into the tables under their names. Then select the desired reference and measurement clock **Records**. Measurement clock records shown in red are from a different MMS system than the reference record and cannot be read. The time format for the start and end time in the clock records table can be chosen as a **UTC** timestamp or **MJD**. Details about the clock database records can be displayed with the **Details** button. Measurement record pairs without overlapping time spans cannot be read. The **Start** and **End** times of measurement record pairs with overlapping time spans will be shown in the **Read Times** section along with their time **Span** and **#** of data points. Records that are continuing are denoted by "Run Continuing", and if both records are continuing the end time will be the current time. Enter the desired **AF** (averaging factor) for the read data. Then press **Read** to read the selected data from the MMS database and write it to a file. The read operation can

take a considerable time if the span is long and/or the database connection is slow, and can be aborted with the **Abort** button during a read. The data read prior to the abort is valid, and may be analyzed by any of the Stable32 analysis functions.

The parameters for the database access can be set by pressing the **Configure** button (see the Database Configure Help topic). These settings will take effect the next time the Database function is opened. The clock names and records can be reset with the **Reset** button (useful if the database may have changed since opening the Database function).

### • TSC MMS System

The Timing Solutions Corporation Multi-Channel Measuring System (MMS) is a multiple mixer clock measurement system. It measures the phase differences between the clocks under test and an offset reference that is common to all measurement channels. These phase differences and the time of the measurement are stored in the MMS database. When data is read from the database, reference and measurement channels must be selected that were measured on the same system over the same time span. Timetagged clock data is then written to a file, where it may be used with Stable32 to perform an analysis.

### • MMS Data

The data output from a MMS, in the form of time differences,  $x$ , are obtained by dividing the measured phase differences of the measurement and reference channels by their respective nominal frequencies, and then subtracting them to remove the effect of the common offset reference:

$$x = (\Delta\phi_{\text{meas}}/v_{\text{meas}}) - (\Delta\phi_{\text{ref}}/v_{\text{ref}})$$

The  $x$  values have units of time in seconds. The  $x(t)$  time fluctuations are related to the phase fluctuations by  $\phi(t) = x(t) \cdot 2\pi v_0$ , where  $v_0$  is the carrier frequency in Hz. Both  $x(t)$  and  $\phi(t)$  are commonly called "phase" to distinguish them from the independent time variable,  $t$ . These data are taken at a uniform measurement time,  $\tau_0$ . The phase data may be sampled at a multiple  $\tau = n \cdot \tau_0$  by using every  $n$ th point (decimation). During subsequent analysis, the  $x(t)$  may be converted to  $y(t)$  values of (dimensionless) fractional frequency,  $\Delta f/f$ .

The Stable32 Database function reads and stores timetags and phase ( $x$ ) values in the phase timetags and data arrays. The measurements are equally spaced at an averaging time (tau) that can be chosen within the database function as any integer multiple of the basic MMS measurement interval. The timetags are formatted as Modified Julian Date (MJD). The MJD is based on the astronomical Julian Date, the # of days since noon on January 1, 4713 BC. The MJD is the Julian Date - 2,400,000.5; it starts at zero at midnight on November 17, 1858. Stable32 handles MJDs over the span of the 21st century, and it has provisions for displaying them in several formats.

The MMS system usually takes data on each clock at a rate of once per second, and each resulting row in the measurements table has a source\_id INT, a pps\_count INT, a utc TIMESTAMP, and a meas NUMERIC(30,10) as shown in character format below:

```
3 1129343209 2005-10-15 02:26:16 9814320.4886926900
```

These rows require 42 bytes. The database for a 12-channel system taking data a once per second grows in size by about 1 million points/day or 42 MB/day. This is a large amount of data to store, read and analyze. The meas value, which represents phase in cycles, has a resolution of 1e-8 cycle. When divided by a nominal frequency of 5e6, this provides a resolution of 2e-15 sec. The actual system resolution is limited by white phase noise to about 0.5 psec (5e-13 sec) rms for a 1 second measurement interval.

The Database function automatically generates a phase data filename based on the measurement clock name with a .phd extension.

### • MMS Database Connection

The computer on which the Stable32 program is run must be connected to the TSC MMS database server via a network before the Database function can be used. Follow the directions for the MMS database system to establish this connection, which should be verified before running the Stable32 Database function. When it is run for the first time, a Database Configure dialog box will appear to enter the necessary access parameters (host IP address, database name, user name, and password). Those settings will be saved for subsequent use by the program (the password is encrypted).

### • Database Queries

The MMS database is queried when the program is opened to gather a list of all the clocks it contains, which is put into both the reference and measurement combo boxes. This represents a snapshot of the clocks in the database which is not updated again unless the **Reset** button is pushed or the program is restarted. When the user selects a clock from one of these boxes, all of the records for that clock are put into the corresponding table. Again, this represents a snapshot of the database - if a continuing run is subsequently ended, the table contents will not dynamically reflect that change. Records for MMS systems different from the one selected as the reference are flagged in red and cannot be read. Finally, after the user has selected the records and the start & end times to be read, the corresponding data is read from the database. For a continuing run for both clocks, the current date/time is used for the end of the record, which will change each time the **Read** button is pushed. If data collection on one or both of the clocks has stopped without an end time having been entered, the read operation will end when the last pair of measurement & reference points is read. In that case, the actual number of points read will be less than indicated for the time span.

### • Reference

*Timing Solutions Measurement Database Hardware and Software User Manual*, Timing Solutions Corp., 4775 Walnut St., Boulder, CO 80303.

## Database Details

### • Database Details

Additional information about the selected reference and measurement clock records can be obtained from the Database Details dialog box. While this information is not necessary to use the database function, it can be helpful for SQL queries to directly access the database. The database entries associated with these items are shown in the table below:

Item	Reference	Measurement
Clock Name:	Cesium	Rubidium
Clock ID:	1	2
System ID:	1	1
Sys Name:	Test	Test
Frequency:	5000000.0000000000	5000000.0000000000
Channel:	3	4
Source ID:	3	4
Meas ID:	2	3
Begun By:		
Ended By:		

Details of selected reference and measurement clock records

OK

<u>Details Item</u>	<u>Database Table(s)</u>	<u>Database Item</u>	<u>Remarks</u>
Clock ID	clock_names	clock_id	Unique key
	measurement_list	clock_id	Foreign key
System ID	measurement_systems	system_id	Unique key
	meas_system_data	system_id	Primary key
	measurement_sources	system_id	Foreign key
	timer_groups	system_id	Primary key
Frequency	measurement_list	frequency	
Channel	measurement_sources	channel	

Source ID	measurement_sources	source_id	Primary key
	measurement_list	source_id	Foreign key
	measurements	source_id	
Meas ID	measurement_list	meas_id	Primary key
Begun By	measurement_list	begun_by	
Ended By	measurement_list	ended_by	

## Database Configure

- **Database Configure**

Configuration information for the database function is stored in the Stable32.ini initialization file in standard Windows format comprising [Section] headings and Item= string values. These items may be edited directly with a text editor, or (preferably) by using the Database Configure dialog box from within the Stable32 program, which is invoked by pressing the Configure button. The Database Configure dialog box is shown below:

<u>Dialog Box</u>	<u>ALT</u>	<u>Control</u>	<u>Control</u>
<u>Control</u>	<u>Key</u>	<u>Type</u>	<u>Description</u>
Database Access		Group Box	Database connection parameters
Host IP Address	I	Edit	Dot address or blank for localhost
Database Name	D	Edit	tsc-postgres is standard MMS database name
User	U	Edit	tsc is standard MMS database user name
Password	W	Edit	tsc is standard MMS database password

- **Initial or Failed Launch**

When the Database function is launched the first time, it is unlikely that the default database access parameters will be correct, and the connection will fail. This will bring up an error message, followed by the Database Configure dialog that will allow the user to enter the correct access parameters. Then, after another error message, the function can be restarted, and the connection should succeed. The same procedure can be followed any time a change is required in the access parameters.

- **Configuration File**

A typical Stable32 configuration file contains the following database items:

```
[Database]
Host=192.168.1.100
Database=tsc
User=Bill
Password=A1B2C3D4
```

The password is encrypted in the Stable32 configuration file (each password ASCII character becomes two hex characters). The [Database] section of the Stable32.ini file contains four parameters associated with database access:

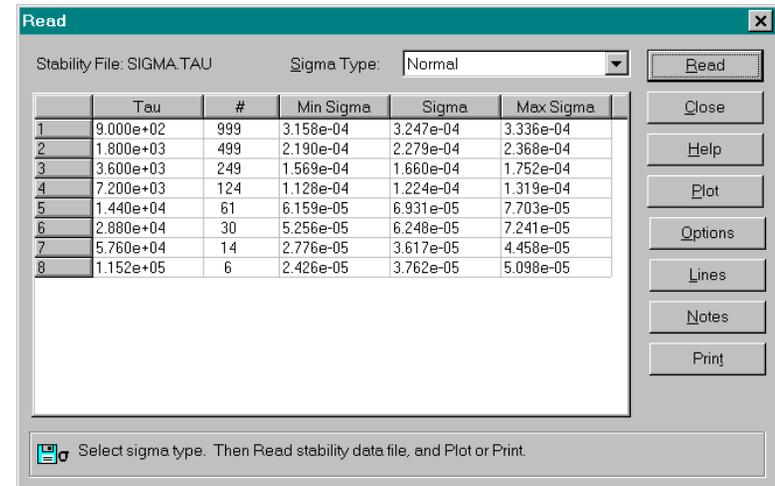
<u>Name</u>	<u>Description</u>	<u>Default Value</u>	<u>Remarks</u>
Host	IP address	1	Blank for localhost
Database	Database name	tsc-postgres	Required
User	User name	tsc	Required
Password	Password	tsc	Not always required Encrypted

These items of the Stable32 configuration file may be manually edited with a text editor, but the preferred method is to use the Database Configure dialog within the Stable32 program. If no Stable32.ini file or [Database] item exists in the Stable32.exe program folder, the program will (almost certainly) follow the failed connection process, and a [Database] configuration file item will be created. The Stable32.ini configuration file is rewritten each time the program closes.

## Read Function

- **Purpose**

Use the Read function to read, print and plot stability data from disk. This stability data may be a table of (possibly edited) stability results from one or more previous Run operations. The main purpose of the Read function is to allow stability data from multiple runs with different basic averaging times to be combined onto a composite stability plot.



- **Controls**

The Open dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
Stability File	Text	Stability data filename.
Sigma Type	Combo	Select sigma type.
Stability Data	Table	Display of stability data.
Read	Pushbutton	Read stability data from disk.
Close	Pushbutton	Close the Read dialog box.
Help	Pushbutton	Invoke the Open help topic.
Plot	Pushbutton	Plot the stability data.
Options	Pushbutton	Enter plot options.
Lines	Pushbutton	Enter plot lines.
Notes	Pushbutton	Enter plot annotations.
Print	Pushbutton	Print the stability data.

- **Operation**

Choose the **Sigma Type** to be read, and press the **Read** button. Use the standard file open dialog box to open the stability file. The stability filename and data will then be displayed. Use the **Plot** or **Print** buttons to plot or print the stability data, or the **Close** button to close the dialog box.

- **Stability File Format**

The format of the stability files written after a Run operation and read by the Read function is slightly different from the screen display. The screen columns are Tau, #, Min Sigma, Sigma, and Max Sigma (where # is the number of analysis points, Sigma is the nominal sigma value, and the Min & Max Sigma values are used to define the plot error bars). The order of the stability file columns is Tau, #, Sigma, Min Sigma, Max Sigma and DF. The Min & Max Sigma and DF (number of X<sup>2</sup> degrees of freedom) columns are optional, and, if they are omitted, the plot will show the Sigma values without error bars. An example of a stability data file is shown below (the column headings are not included in the file):

<u>Tau</u>	<u>#</u>	<u>Sigma</u>	<u>Min Sigma</u>	<u>Max Sigma</u>	<u>DF</u>
1.000e+01	999	2.922e-01	2.846e-01	3.005e-01	6.651e+02
2.000e+01	997	2.010e-01	1.953e-01	2.072e-01	5.693e+02
4.000e+01	993	1.448e-01	1.396e-01	1.506e-01	3.456e+02
8.000e+01	985	1.057e-01	1.006e-01	1.117e-01	1.818e+02
1.600e+02	969	6.191e-02	5.780e-02	6.706e-02	9.121e+01
3.200e+02	937	4.808e-02	4.372e-02	5.409e-02	4.478e+01
6.400e+02	873	3.624e-02	3.177e-02	4.334e-02	2.141e+01
1.280e+03	745	2.767e-02	2.307e-02	3.691e-02	9.710e+00

The stability file items for the Both sigma case are Tau, #, Std Dev, Sigma, Mod Sigma and DF. No DF information is available for Normal and Hadamard variances, or for MTIE and TIErms, and the DF value is set to zero. The stability file contains no header or other text. It is comprised of space-delimited columns of ASCII characters in any numeric format. Examine the files written by the Run function for more examples. These stability files may be edited and combined as required to produce a composite stability plot. The Tau values should be in numerical order.

The TIE rms and MTIE stability files do not contain Min & Max Sigma or DF columns.

- **Multiple Data Sets**

The Stable32 Read function includes provisions for reading and plotting multiple sets of stability data. A new data set is started whenever the tau value is less than the previous one. The points between multiple data sets are not connected. One application for multiple data sets is to show both stability data and its corresponding specification limit.

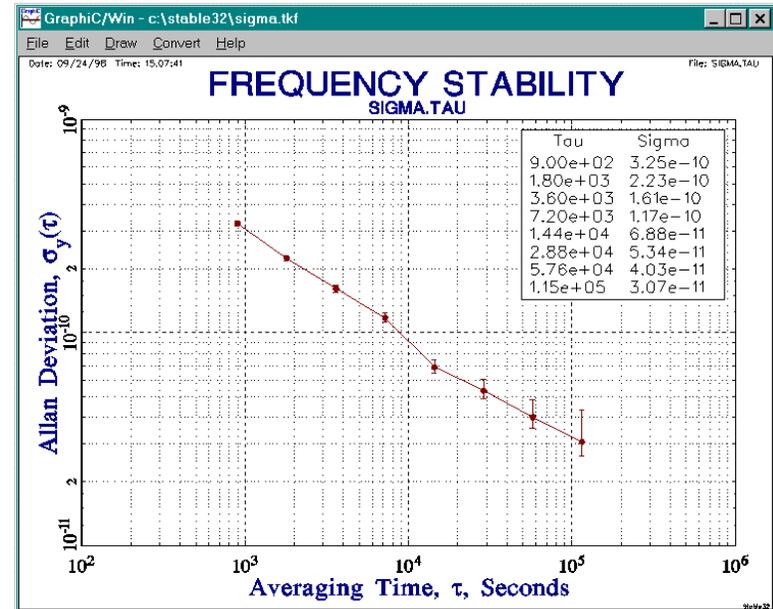
- **Other Options**

By default, all data points are shown with symbols, and are included in the data table. If, however, the # of analysis points is made zero for the first row of a set, that data will be plotted without symbols. This may be desirable for showing spec limits. In addition, all data rows having # = 0 are excluded from the data table. Thus spec limits can be excluded from the table by setting the # analysis points to zero and placing them at the end of the data file.

## Read Plot Function

- **Purpose**

Use the Read Plot function to generate a stability plot.



## Read Options Function

- **Purpose**

Use the Read Options function to control the attributes of the stability plot.

- **Controls**

The Read Options dialog box contains the following controls:

<b>Control</b>	<b>Type</b>	<b>Description</b>
<u>T</u> itle	Edit	Enter plot title.
<u>S</u> ub Title	Edit	Enter plot subtitle.
<u>M</u> essage	Edit	Enter plot message.
Message <u>P</u> osition	Combo	Choose message position.
Table <u>P</u> osition	Combo	Choose table position.
Read <u>C</u> onfig	Pushbutton	Read options from INI file.
Save <u>C</u> onfig	Pushbutton	Save options in INI file.
Stability <u>F</u> ile	Text	Display name of stability file.
X-Axis	Group	X-Axis parameters.
<u>L</u> abel	Edit	Enter x-axis label.
<u>T</u> au Max:	Text	Maximum tau value.
<u>S</u> cale Max	Edit	Enter x-axis (tau) scale maximum.
<u>T</u> au Min	Text	Minimum tau value.
<u>S</u> cale <u>M</u> in	Edit	Enter x-axis (tau) scale minimum.
Y-Axis	Group	Y-Axis parameters
<u>L</u> abel	Edit	Enter y-axis label.
<u>S</u> igma Max	Text	Maximum sigma value.
<u>S</u> cale <u>M</u> ax	Edit	Enter y-axis (sigma) scale maximum.

Sigma Min	Text	Minimum sigma value.
Scale Min	Edit	Enter y-axis (sigma) scale minimum.
Options	Group	Plot options.
Date	Checkbox	Show date on plot.
Box	Checkbox	Draw box around plot.
Bars 1	Checkbox	Show error bars on plot.
Table 2	Checkbox	Draw stability table on plot.
Fit 3	Checkbox	Show fit parameters on plot.
Plot File	Edit	Enter plot filename.
Reset All	Pushbutton	Reset all plot attributes to defaults.
OK	Pushbutton	Set plot attributes & close dialog box.
Cancel	Pushbutton	Abort changes & close dialog box.
Help	Pushbutton	Invoke the Read Options help topic.

• **Operation**

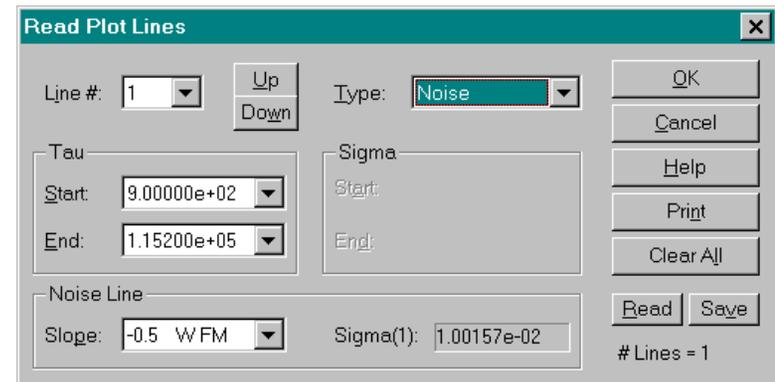
Enter the desired plot attributes and press **OK** to accept them, or **Cancel** to abort. Press **Clear All** to reset all attributes to their default values.

## Read Lines Function

• **Purpose**

Use the Read Lines function to add lines to a stability plot. The possible line types are as follows:

<u>Line Type</u>	<u>Description</u>	<u>Parameter</u>
None	No line	None
Endpoints	Line between certain endpoint values	None
Horizontal	Horizontal line at certain sigma value	None
Vertical	Vertical line at certain tau value	None
Noise	Fit to certain power-law noise model	Sigma(1)



• **Controls**

The Read Lines dialog box contains the following controls. Some parameters do not apply to all line types.

<u>Control</u>	<u>Type</u>	<u>Description</u>
Line #	Edit	Enter # of line to edit.
Up	Pushbutton	Increment line #.
Down	Pushbutton	Decrement line #.
Type	Combo	Select line type.
Tau	Group	X-Axis line parameters.
Start	Edit/Combo	Enter x-axis start of line.
End	Edit/Combo	Enter x-axis end of line.
Sigma	Group	Y-axis line parameters.
Start	Edit	Enter y-axis start of line.
End	Edit	Enter y-axis end of line.
Slope:	Combo	Noise type for fit
Sigma(1)	Text	Noise line fit parameter.
# Lines	Text	# of lines entered.

<u>O</u> K	Pushbutton	Perform Read Lines function & close dialog box.
<u>C</u> ancel	Pushbutton	Abort Read Lines function & close dialog box.
<u>H</u> elp	Pushbutton	Invoke the Read Lines help topic.
<u>P</u> rint	Pushbutton	Print line parameters.
<u>C</u> lear <u>A</u> ll	Pushbutton	Clear all lines.
<u>R</u> ead	Pushbutton	Read line information from INI file.
<u>S</u> ave	Pushbutton	Save line information in INI file.

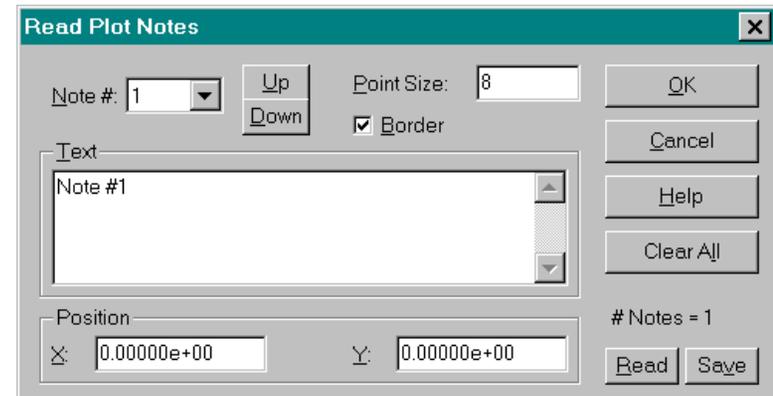
• **Operation**

Select the desired **Line #** and **Type**. Enter the applicable **Tau Start**, **Tau End**, **Sigma Start** and **Sigma End** parameters. Choose the applicable **Slope** for a **Noise Line**, and press **OK** to accept the lines, or **Cancel** to abort. Press **Clear All** to remove all lines.

## Read Notes Function

• **Purpose**

Use the Read Notes function to add annotations to a stability plot.



• **Controls**

The Read Notes dialog box contains the following controls:

<u>C</u> ontrol	<u>T</u> ype	<u>D</u> escription
<u>N</u> ote #	Edit	Enter the # of the note to edit.
<u>U</u> p	Pushbutton	Increment the note #.
<u>D</u> own	Pushbutton	Decrement the note #.
<u>P</u> oint Size	Edit	Enter the point size for the note.
<u>T</u> ext	Edit	Enter the text of the note.
<u>P</u> osition	Group	Note position values.
<u>X</u>	Edit	Enter the x-position of the note.
<u>Y</u>	Edit	Enter the y-position of the note.
<u>#</u> Notes	Text	# notes entered.
<u>B</u> order	Checkbox	Draw a border around the note.
<u>R</u> ead	Pushbutton	Read line information from INI file.
<u>S</u> ave	Pushbutton	Save line information in INI file.
<u>O</u> K	Pushbutton	Accept the settings & close dialog box.
<u>C</u> ancel	Pushbutton	Abort the changes & close dialog box.
<u>H</u> elp	Pushbutton	Invoke the Read Notes help topic.
<u>C</u> lear <u>A</u> ll	Pushbutton	Clear all notes.

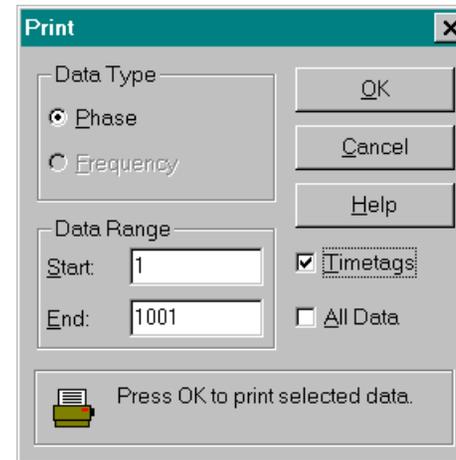
• **Operation**

Enter the desired plot annotations and press **OK** to accept them, or **Cancel** to abort. Press **Clear All** to clear all annotations.

## Print Function

- **Purpose**

Use the Print function to print a listing of all or a portion of the current phase or frequency data.



- **Controls**

The Print dialog box contains the following controls:

Control	Type	Description
Data Type	Group	Data type.
Phase	Radiobutton	Select phase data.
Frequency	Radiobutton	Select frequency data.
Data Range	Group	Data range.
Start	Edit	Enter first point to print.
End	Edit	Enter last point to print.
Timetags	Checkbox	Print timetag data.
All Data	Checkbox	Print all data.
OK	Pushbutton	Perform the Print function & close dialog box.
Cancel	Pushbutton	Abort the Print function & close dialog box.
Help	Pushbutton	Invoke the Print help topic.

- **Operation**

Choose **Phase** or **Frequency** data and enter the **Start** and **End** points to print, or check **All Data** to print all the data. Check **Timetags** to also print the corresponding timetag data. Then press **OK** to print or **Cancel** to abort.

## Edit Function

- **Purpose**

Use the Edit function to display, search and edit the current phase or frequency data. The primary purpose of this function is to examine phase or frequency data for gaps, outliers and other anomalous values, relating them to their point number or timetag. For example, this function can be used to quickly find the date and time of an outlier in a set of frequency data (outlier identification is not available for phase data). The Edit function is not intended for manual data entry or heavy editing tasks. Use Windows Notepad or another ASCII text editor for that purpose. Notepad (or the substitute defined in the STABLE32.INI configuration file) can be invoked from within Stable32 from the Utility/Notepad menu or the Pad toolbutton.

If timetag information is available, check the Show Timetags box to display it. It is also helpful to check the Color Gaps and Color Outliers boxes to emphasize those points. Several Timetag Formats are available including Modified Julian Date (MJD), Day of Year (DOY), Date and Exponential (Exp). The MJD format also serves to display any decimal numeric format.

The Go To group allows easy access to the data by point number, while the Find group allows easy access to any gaps and outliers. The Detailed Search group provides more detailed Find and Replace functions as described in the next sections of this User Manual.

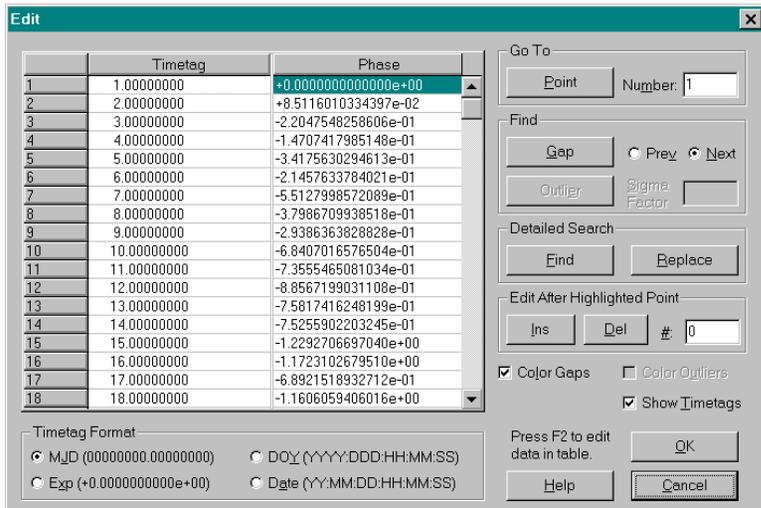
The Edit After Highlighted Point group allows inserting or deleting a certain number of points. The inserted points are 0's (gaps), and this can be a convenient way to correct the data for missing points.

Data editing is accomplished by highlighting a data point with the left mouse button or moving to it with the up or down arrow keys. A new value (such as 0 for a gap) can then be inserted with a numeric key, or the existing value can be edited by pressing the F2 key (which will activate the EDIT MODE annunciator). Timetag values are not editable.

- **Controls**

The Edit dialog box contains the following controls:

<b><u>Control</u></b>	<b><u>Type</u></b>	<b><u>Description</u></b>
Go To	Group	Go To controls.
<u>P</u> oint	Pushbutton	Go to selected point #.
<u>N</u> umber	Edit	Enter point # to go to.
<u>F</u> ind	Group	Find controls.
<u>G</u> ap	Pushbutton	Find gap.
<u>P</u> re <u>v</u>	Radiobutton	Find previous occurrence.
<u>N</u> ext	Radiobutton	Find next occurrence.
<u>O</u> utlier	Pushbutton	Find outlier.



<u>S</u> igma Factor	Edit	Enter outlier criterion.
Detailed Search	Group	Detailed search & replace controls.
<u>F</u> ind	Pushbutton	Invoke Find function.
<u>R</u> eplace	Pushbutton	Invoke Replace function.
Edit After	Group	Insertion and deletion controls.
Highlighted Pt		
<u>I</u> ns	Pushbutton	Insert gap(s).
<u>D</u> el	Pushbutton	Delete point(s).
<u>#</u>	Edit	# points to insert/delete.
<u>C</u> olor Gaps	Checkbox	Highlight gaps in blue.
<u>C</u> olor <u>O</u> utliers	Checkbox	Highlight outliers in red.
<u>S</u> how <u>T</u> imetags	Checkbox	Show timetags along with data.
Timetag Format	Group	Timetag format choices.
<u>M</u> JD	Radiobutton	Show timetags in MJD format.
<u>D</u> OY	Radiobutton	Show timetags in DOY format.
<u>E</u> xp	Radiobutton	Show timetags in exponential format.
<u>D</u> ate	Radiobutton	Show timetags in date format.
<u>O</u> K	Pushbutton	Accept editing changes & close dialog box.
<u>C</u> ancel	Pushbutton	Abort editing changes & close dialog box.
<u>H</u> elp	Pushbutton	Invoke the Edit help topic.

• **Operation**

The Edit function can be used for several purposes. The simplest use is to just display the current phase or frequency data. Check the **Show Timetags** control to also display the corresponding timetag data, and use the vertical scroll bars to scroll through the data. The data point #s are shown in the left column of the data display. Use the **Go To Point #** controls to move to a particular data point. Check the **Color Gaps** and/or **Color Outliers** to highlight those data points. The outlier criterion can be set with the **Sigma Factor** entry box. The Color Gaps option highlights timetag intervals that differ by more than  $\pm 1\%$  from the starting value, and data points with a value of zero (except the 1st and last points of phase data).

The **Previous** and **Next Gaps** and **Outliers** can be found using the **Find** controls. More elaborate searching is supported by the **Find** and **Replace** functions of the **Detailed Search** controls.

Simple editing can be done with the usual editing keys directly in the data display, including changing, adding and deleting data. (No adjustments are made in the timetags when data is added or deleted, and the timetag column is not editable.) **Insertion** or **Deletion** of a certain # of points is facilitated by the **Edit After Highlighted Point** controls. The editing changes can be accepted with the **OK** button, or aborted by pressing **Cancel**.

• **Navigation**

A cell may be selected with a mouse click, the scroll bars, or the arrow keys. A cell may also be selected by using one of the **Go To**, **Find** or **Search** functions. If status bar plots are enabled, the corresponding plot is dynamically updated during editing, and a small cursor appears above the plot to indicate the position of the currently selected point.

• **Editing Data**

A new value can be entered directly into the selected cell with the numeric keys. The existing data may be edited by first pressing the F2 key. It is necessary to delete a portion of the existing data to make room for the new entries. The usual editing keys apply. The original value may be restored with the Esc key, and the changes are accepted with the Enter key.

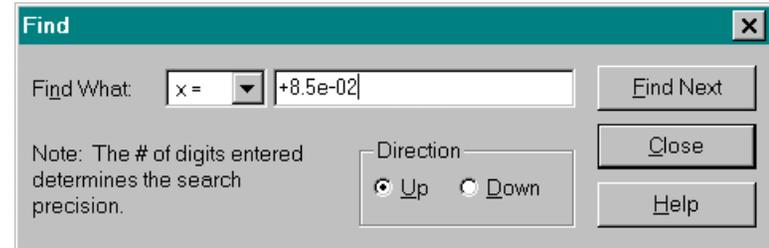
• **Caution**

Editing changes in phase and frequency data must be made with discretion so that the integrity of the data is not lost.

## Find Function

- **Purpose**

The Find function is a Detailed Search sub-function of the Edit function. Use the Find function to find phase or frequency data points that meet a particular criterion. The Replace function is also available to find and replace phase or frequency data values.



- **Controls**

The Find function uses a modeless dialog box that can remain open while the Edit function is active. It contains the following controls:

<u>C</u> ontrol	<u>T</u> ype	<u>D</u> escription
<u>F</u> ind What	Combo	Search type.
<u>E</u> dit	Edit	Search value.
<u>D</u> irection	Group	Search direction controls.
<u>U</u> p	Radiobutton	Search upwards.
<u>D</u> own	Radiobutton	Search downwards.
<u>F</u> ind Next	Pushbutton	Perform a Find operation.
<u>C</u> lose	Pushbutton	Abort the Find function & close dialog box.
<u>H</u> elp	Pushbutton	Invoke the Find help topic.

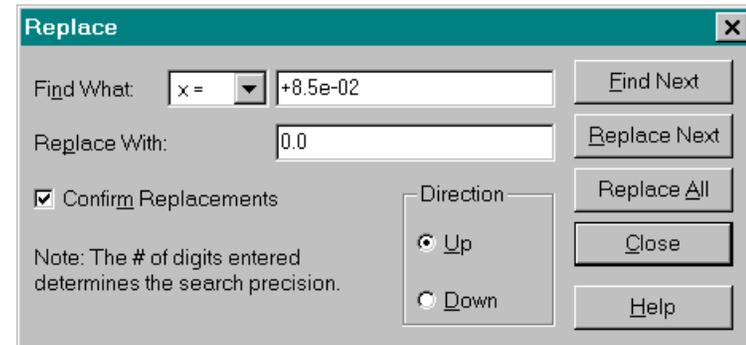
- **Operation**

Select the search criterion with the two **Find What** controls. The numeric value also determines the search precision. Set the search direction with the **Up** and **Down** buttons, and press the **Find Next** button. The result of a successful search is displayed in the Edit dialog box. Additional search may be conducted with the **Find Next** button. The Find dialog box can be closed with the **Cancel** button or by closing the Edit dialog box.

## Replace Function

- **Purpose**

The Replace function is a Detailed Search sub-function of the Edit function. Use the Replace function to find and replace data phase or frequency data points that meet a particular criterion. The Find function is also available to simply find certain phase or frequency data values.



- **Controls**

The Replace function uses a modeless dialog box that can remain open while the Edit function is active. It contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
<u>F</u> ind What	Combo	Search type.
	Edit	Search value.
<u>R</u> eplace With	Edit	Replacement value.
<u>D</u> irection	Group	Search direction controls.
<u>U</u> p	Radiobutton	Search upwards.
<u>D</u> own	Radiobutton	Search downwards.
<u>F</u> ind Next	Pushbutton	Perform a Find operation.
<u>R</u> eplace Next	Pushbutton	Perform a Replace operation for one point.
<u>R</u> eplace <u>A</u> ll	Pushbutton	Perform a Replace operation for all points.
<u>C</u> onfirm	Checkbox	Confirm each replacement.
<u>C</u> lose	Pushbutton	Abort Replace function & close dialog box.
<u>H</u> elp	Pushbutton	Invoke the Replace help topic.

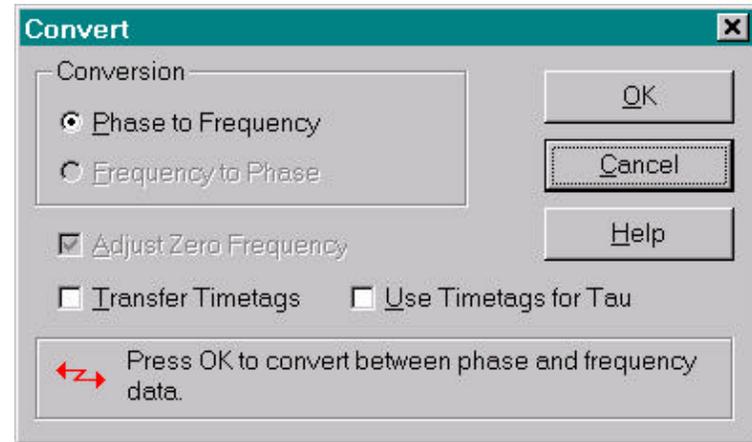
- **Operation**

Select the search criterion with the two **Find What** controls. The numeric value also determines the search precision. Enter the value that it is to be **Replaced With**. Set the search direction with the **Up** and **Down** buttons, and press one of the **Find Next**, **Replace Next** or **Replace All** buttons. The result of a successful search is displayed in the Edit dialog box. Additional searches may be conducted with the **Find Next** button, and additional replacements with the **Replace Next** or **Replace All** buttons. Close the Replace dialog with the **Cancel** button or by closing the Edit dialog box.

## Convert Function

- **Purpose**

Use the Convert function to convert between phase and frequency data.



- **Controls**

The Convert dialog box contains the following controls:

<u>C</u> ontrol	<u>T</u> ype	<u>D</u> escription
Conversion	Group	Choose conversion type.
Phase to Frequency	Radiobutton	Select $\phi \rightarrow f$ conversion.
Frequency to Phase	Radiobutton	Select $f \rightarrow \phi$ conversion.
<u>N</u> ormalize Frequency	Checkbox	Remove average frequency before $f \rightarrow \phi$ conversion.
<u>A</u> dj Zero Frequency	Checkbox	Adjust zero freq to $1e-99$ in $\phi \rightarrow f$ conversion.
<u>T</u> ransfer Timetags	Checkbox	Transfer timetags along with data.
<u>U</u> se Timetags for Tau	Checkbox	Use timetags as tau for conversion.
<u>O</u> K	Pushbutton	Perform conversion & close box.
<u>C</u> ancel	Pushbutton	Abort conversion & close dialog box.
<u>H</u> elp	Pushbutton	Invoke the Convert help topic.

- **Operation**

Select desired **Conversion** type, and desired **Normalize Frequency** or **Adjust Zero Frequency**, **Transfer Timetags**, and **Use Timetags for Tau** options, then press **OK**. To abort, press **Cancel**.

- **Conversions**

Phase to frequency conversion is done by dividing the first differences of the phase points by the averaging time:  $y = x_{i+1} - x_i / \tau$ . The inverse frequency to phase conversion is done by piecewise integration using the averaging time as the integration interval:  $x_{i+1} = x_i + y_i \cdot \tau$ . Any gaps in the frequency data are filled to obtain phase continuity. Two equal adjacent phase values will

result in zero frequency (which is treated as a gap) unless the **Adjust Zero Frequency** option is used (recommended – see below).

- **Zero Frequency**

The conversion of phase data having two equal adjacent values results in a fractional frequency value of zero. While this is correct, it will subsequently be treated as a gap in the frequency data. That problem (for which there is an optional warning during the reading of phase data) can be avoided by using the Adjust Zero Frequency option during phase to frequency conversion, which will adjust the frequency value to 1e-99.

- **Gaps**

Gaps in frequency data are filled in the converted phase data with values based on the average frequency.

- **Timetags as Tau**

The Use Timetags for Tau option allows the time difference between adjacent timetags to be used as the tau for each point's phase to frequency conversion. This is not normally used except for non-uniformly spaced data. The tau units must be in seconds. Timetags in other units (e.g. MJD fractional days) must be scaled to seconds (e.g. 86400) by the PhaseTagScale multiplier in the Stable32.ini, which can be entered from the Inputs function (accessed via the Data File Opened dialog box or the Utilities menu. The average timetag interval is used as the tau value for the converted frequency data.

- **Notes**

The conversion of phase data having two equal adjacent values results in a fractional frequency value of zero. While this is correct, it would subsequently be treated as a gap in the frequency data. That problem can be avoided by using the Adjust Zero Frequency option during the phase to frequency conversion, which will adjust the frequency value to 1e-99.

- **Cautions**

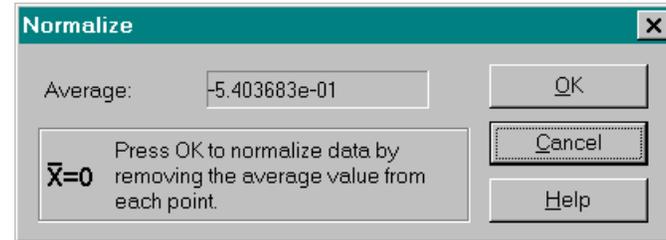
Gaps in frequency data are converted to phase data by linear interpolation using the average frequency value, which can cause a discontinuity in the resulting phase data.

The Use Timetags for Tau option will fail (generating an error message) if two adjacent timetags are equal. Frequency data converted from unequally spaced phase data using the Use Timetags for Tau option will be correct but must have approximately the same tau to support subsequent analysis.

## Normalize Function

- **Purpose**

Use the Normalize function to remove the average value from phase or frequency data. This normalizes the data to have a mean value of zero.



- **Controls**

The Normalize dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
Average	Text	Display the mean value.
<u>OK</u>	Pushbutton	Perform the normalize function & close dialog box.
<u>Cancel</u>	Pushbutton	Abort the normalize function & close dialog box.
<u>H</u> elp	Pushbutton	Invoke the Normalize help topic.

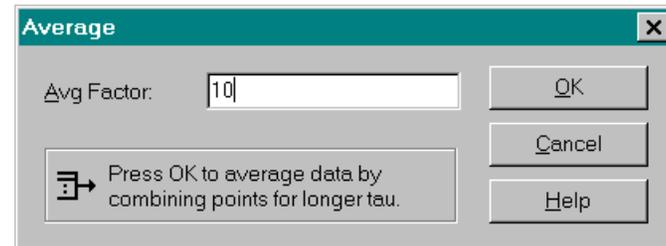
- **Operation**

Press **OK**. To abort, press **Cancel**.

## Average Function

- **Purpose**

Use the Average function to combine groups of phase or frequency data into values corresponding to a longer averaging time,  $\tau$ . Note: The entire phase or frequency data array is averaged regardless of the analysis limits.



- **Controls**

The Average dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
<u>A</u> vg Factor	Edit	Enter averaging factor (default = 1).
<u>O</u> K	Pushbutton	Perform the Average function & close dialog box.
<u>C</u> ancel	Pushbutton	Abort the Average function & close dialog box.
<u>H</u> elp	Pushbutton	Invoke the Average help topic.

- **Operation**

Enter the desired **Avg Factor**, and press **OK**. To abort, press **Cancel** (or **OK** with default averaging factor = 1).

- **Phase Averaging**

Phase data averaging (decimation), which is done by simply eliminating the intermediate data points to form data at a longer tau, which corresponds to frequency averaging. Do not confuse this process with actual averaging of the phase data (such as is done when computing the modified Allan variance).

- **Frequency Averaging**

Frequency averaging is done by finding the mean value of the data points being averaged.

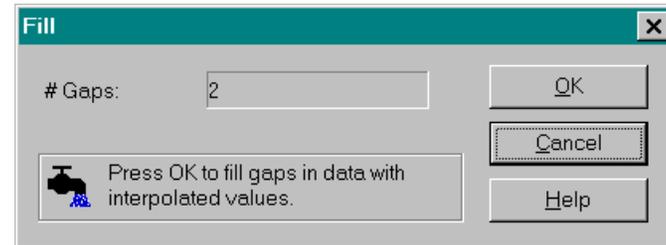
- **Gaps**

Gaps (0s) are ignored, and the average will be a gap if all points in the group being averaged are gaps.

## Fill Function

- **Purpose**

Use the Fill function to fill gaps in the selected portion of phase or frequency data with interpolated values. Leading and trailing gaps are removed, the # of data points is adjusted, and the analysis limits are reset.



- **Controls**

The Fill dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
# Gaps	Text	Display # gaps in data.
<u>O</u> K	Pushbutton	Perform the Fill function & close dialog box.
<u>C</u> ancel	Pushbutton	Abort the Fill function & close dialog box.
<u>H</u> elp	Pushbutton	Invoke the Fill help topic.

- **Operation**

Press **OK**. To abort, press **Cancel** (or **OK** if # gaps=0).

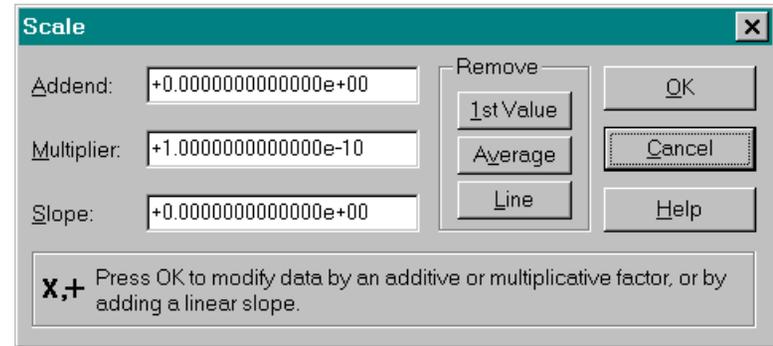
- **Caution**

The data inserted are simply interpolated values using the closest non-gap data points at either side of each gap. No attempt is made to simulate noise, and the resulting statistics are not necessarily valid. It is generally better practice to leave gaps unfilled.

## Scale Function

- **Purpose**

Use the Scale function to modify the selected phase or frequency data by an additive or multiplicative factor, or by adding a linear slope.



- **Controls**

The Scale dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
<u>A</u> ddend	Edit	Enter addend (default = 0).
<u>M</u> ultiplier	Edit	Enter multiplier (default = 1).
<u>S</u> lope	Edit	Enter slope per $\tau$ interval (default = 0).
<u>R</u> emove	Group	Insert values to be removed from data.
<u>1</u> st Value	Pushbutton	Insert -1st value into Addend.
<u>A</u> verage	Pushbutton	Insert -average value into Addend.
<u>L</u> ine	Pushbutton	Insert -linear slope value into Slope.
<u>O</u> K	Pushbutton	Perform the Scale function & close dialog box.
<u>C</u> ancel	Pushbutton	Abort the Scale function & close dialog box.
<u>H</u> elp	Pushbutton	Invoke the Scale help topic.

- **Operation**

Enter the value of the **Addend**, **Multiplier**, and/or **Slope** to be added to the data. Or automatically insert the negative of the **1st Value** or **Average** into the **Addend**, or the negative of the **Linear** slope into the **Slope**. Then Press **OK** to scale the data, or **Cancel** (or **OK** with defaults) to abort.

- **Scaling Equation**

The data is scaled according to the following equation:

$$\text{Scaled Value} = (\text{Original Value} \cdot \text{Multiplier}) + \text{Addend} + (\text{Slope} \cdot \text{Data Point \#})$$

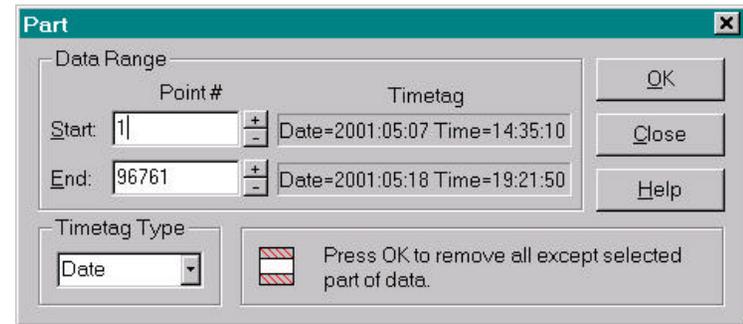
- **Caution**

It is possible to scale only a portion of a data array, but this is unlikely to produce a satisfactory overall result.

## Part Function

- **Purpose**

Use the Part function to clear all except a portion of the current phase or frequency data. The Part function changes the data in memory. Use the analysis limits on the status bar to temporarily change the range of the data to be analyzed.



- **Controls**

The Part dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
Data Range	Group	Controls for entering data range to select.
Point #	Static	Start and end point #s.
<u>S</u> tart	Edit	Start data point #.
+/-	Pushbuttons	Increment/decrement start point #.
<u>E</u> nd	Edit	End data point #.
+/-	Pushbuttons	Increment/decrement end point #.
Timetag	Static	Start and end timetag data.
Timetag Type	Combo	Choose timetag type
<u>O</u> K	Pushbutton	Perform the Part function & close dialog box.
<u>C</u> ancel	Pushbutton	Abort the Part function & close dialog box.
<u>H</u> elp	Pushbutton	Invoke the Part help topic.

- **Operation**

Enter the **Start** and **End** data points to be selected. If timetags are available, choose the desired **Timetag Type** to help select the desired data points. Use the **OK** to delete the data outside these limits, or **Cancel** to abort.

- **Timetag Display**

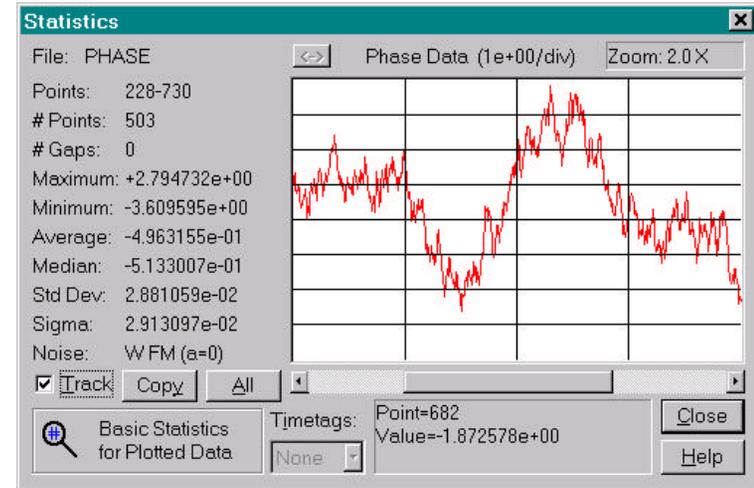
Timetags can be displayed for the start and end data points in MJD, DOY, Date or general exponential format.

## Statistics Function

- **Purpose**

Use the Statistics function to display basic statistics for, and a simple plot of, phase or frequency data. The full filename and the following statistics are displayed:

Point Range	# Points	# Gaps	Minimum	Maximum
Average	Median	Std Dev	Sigma	Noise Type



- **Controls**

The Statistics dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
File	Text	Data filename.
<->	Pushbutton	Swap data type.
Plot Title	Text	Plot type and scale factor.
Statistics	Text	Basic statistics for data (see above).
Plot	Graphic	Data plot.
Scrollbar	Scrollbar	Data selection.
Point	Text	Data display.
Zoom	Text Box	Zoom factor.
Track	Checkbox	Enable scroll tracking.
Copy	Pushbutton	Copy plot to clipboard.
Timetags	Combo	Select timetag type.
All	Pushbutton	Select all data.
Close	Pushbutton	Close the Statistics dialog box.
Help	Pushbutton	Invoke the Statistics help topic.

- **Operation**

Use the mouse cursor to select data points for numeric display, and the scrollbar to select them for statistics and plotting. Zoom and un-zoom the

plot with the R and L mouse buttons. The **Track** checkbox determines if the plot updates during scrolling. Disable tracking for large data sets and/or slow computers. Use the **Timetags** box to select the timetag display for data that has them. The **All** button selects all the data and disables the scrollbar, and the **<->** button switches between phase and frequency data (if equivalent data exists). Press the **Close** button to close the Statistics dialog box.

**Zooming**

The plot and statistics can be zoomed in and out by clicking the R and L mouse buttons while the cursor is in the plot. The zooming is centered on the mouse position. The width of the scroll box (thumb) indicates the relative span of the plot, and the zoom factor is shown in its text box. The status bar plots are not affected by Statistics plot zooming, and, if set to the full range, can serve as a guide to the overall data.

**Scrolling**

The usual horizontal scrolling operations apply. Dragging the scroll box scrolls the plot. Clicking on the L and R arrows, or pressing the left/up or right/down arrow keys, moves the plot one point in the corresponding direction. Clicking on the L and R scrollbar shaft regions, or pressing the page up or page down keys, moves the plot one plot width in the corresponding direction. Pressing the home or end keys moves the plot to the first or last point. The scrollbar must have the input focus (be blinking) for the keyboard inputs to work. During scrolling, the point range is shown in the text box below the scrollbar.

**Plot Scale**

The plot is scaled to a standardized value at the top of the plot. During scrolling, the plot scale and/or # of divisions may change.

**Average**

A green horizontal line is drawn to show the average value of a frequency plot.

**Data and Timetag Readout**

Data point #s and their corresponding values are shown in the text box below the plot when the mouse cursor is in the plot region. Phase data is read at the points, while freq data is read on the horizontal lines that show the averaging time. Timetags can also be displayed for the selected data point in MJD, DOY, Date or general exponential formats.

**Data Type Switching**

Use the **<->** button to switch between equivalent phase and frequency data at the same zoom factor.

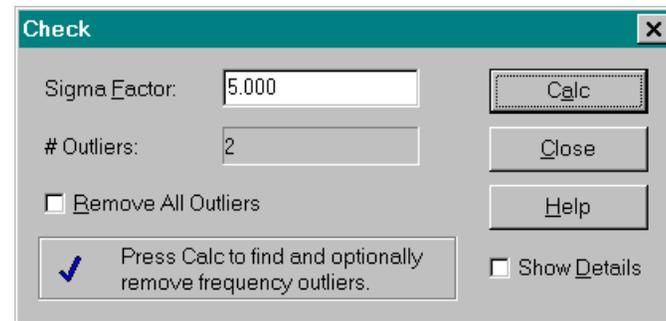
**Caution**

The Statistics function can be slow when processing frequency data having many repeated values because of the function used to find the median value.

## Check Function

**Purpose**

Use the Check function to check for and remove outliers from frequency data.

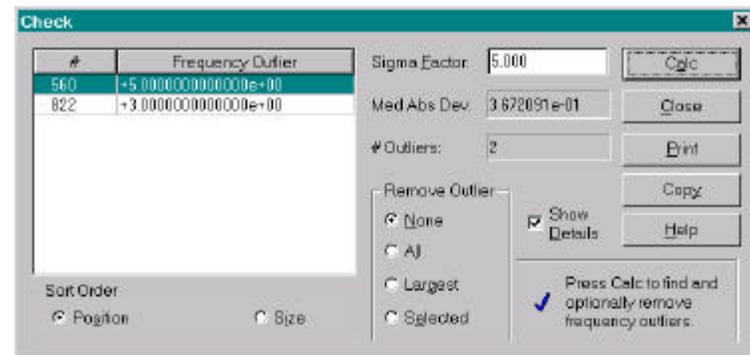


**Controls**

The Check dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
Sigma Factor	Edit	Enter desired outlier criterion
# Outliers	Text	Display the # of outliers
Remove All	Checkbox	Remove all frequency outliers
Show Details	Checkbox	Show detailed Check dialog box
Calc	Pushbutton	Perform Check function & close dialog box.
Close	Pushbutton	Abort Check function & close dialog box.
Help	Pushbutton	Invoke the Check help topic.

Activating the Show Details option adds the following controls to the Check dialog box:



<u>Control</u>	<u>Type</u>	<u>Description</u>
Sort Order	Group	Choose sort order.
Position	Radiobutton	Sort outliers by position.
Size	Radiobutton	Sort outliers by size.
Sigma Factor	Edit	Enter sigma factor.
Med Abs Dev	Text	Display MAD value.
Remove Outliers	Group	Choose type of outlier removal.
None	Radiobutton	Don't remove any outliers.
All	Radiobutton	Remove all outliers.
Largest	Radiobutton	Remove the largest outlier.
Selected	Radiobutton	Remove the selected outlier.
Print Pushbutton	Print outliers.	
Copy	Pushbutton	Copy outliers to clipboard.

### • Operation

Observe the # of **Outliers** that meet the default  $5\sigma$  criterion. Enter the desired outlier criterion in the **Sigma Factor** box, and press **Calc** to find the # of outliers that meet this criterion. Optionally check the **Remove All Outliers** checkbox and press **OK** to replace them with gaps, or press **Close** to abort the Check function. The detailed Check dialog box also presents a list of the outliers, allows a choice of sorting by **Position** or **Size** (absolute deviation from median), and removing **None**, **All**, the **Largest** or the **Selected** outliers. Pressing **Print** will print a list of the outliers and **Copy** will copy them to the clipboard.

### • Outlier Recognition

Stable32 uses the median absolute deviation (MAD) as its means of outlier recognition. The MAD is a robust statistic based of the median of the data. It is the median of the scaled absolute deviations of the data points from their median value, defined as  $MAD = \text{Median} \{ |y(i) - m| / 0.6745 \}$ , where  $m = \text{Median} \{ y(i) \}$ , and the factor 0.6745 makes the MAD equal to the standard deviation for normally distributed data. Each frequency data point,  $y(i)$ , is compared with the median value of the data set,  $m$ , plus or minus the desired multiple of the MAD.

### • Limits

The Check function does not work with phase data, which must be converted to frequency data to apply the Check function.

### • Auto Calc

A check can be done automatically when the Check function is invoked by setting the AutoCalc item in the Configure function.

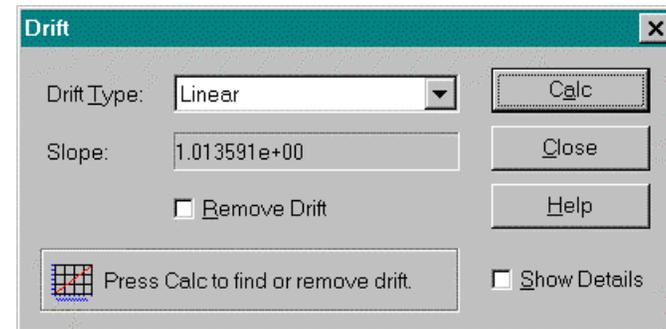
### • Caution

Judgment is required when removing outliers. Automatic outlier removal is convenient for quickly plotting and analyzing frequency data, which may have to be removed to obtain meaningful results. While it provides a degree of objectivity, the analyst is still responsible for decisions regarding outlier removal. It is particularly important to explain all phase discontinuities.

## Drift Function

### • Purpose

Use the Drift function to analyze phase or frequency data for frequency drift, or find frequency offset in phase data. It is common to remove the deterministic frequency drift from phase or frequency data before analyzing the noise with Allan variance statistics. It is sometimes useful to remove only the frequency offset from phase data.

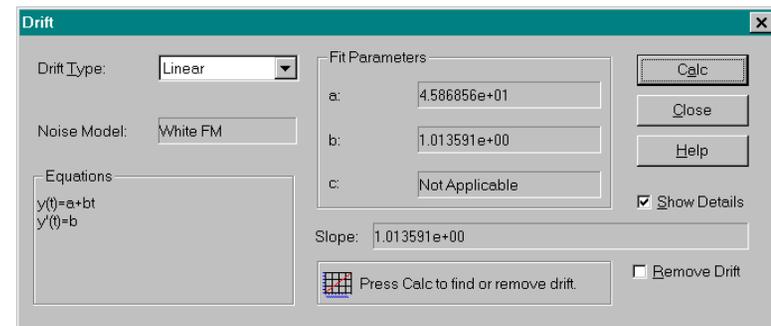


### • Controls

The basic Drift dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
Drift Type	Combo	Select drift analysis type.
Slope	Text	Display slope value.
Remove Drift	Checkbox	Remove drift from data.
Calc	Pushbutton	Perform the Drift function.
Close	Pushbutton	Abort the Drift function & close dialog box.
Help	Pushbutton	Invoke the Drift help topic.
Show Details	Checkbox	Show detailed Drift dialog box.

Activating Show Details adds the following controls to the Drift dialog box:



<u>Control</u>	<u>Type</u>	<u>Description</u>
Noise Model	Text	Noise model for the drift type.
Equations	Group	Equations for the drift fit.
Fit Parameters	Group	Results of drift fit.
a, b and c:	Text	Fit parameters.

- **Operation**

Select the desired **Drift Type**, and optionally check **Remove Drift**. Then press **OK** to find (and optionally remove) the drift, or **Cancel** to abort. The **Show Details** checkbox changes the dialog box to a detailed format, and the **Help** button invokes a Help screen. A changed **Drift Type** clears the results.

- **Detailed Drift Dialog Box**

The detailed Drift dialog box provides additional information about the drift analysis. The **Noise Model** is the power law noise process corresponding to the selected drift type that provides white residuals. The **Equations** box shows the equations for the noise fit, while the **Fit Parameters** box shows the values of the all the fit parameters.

- **Drift Analysis Methods**

Several drift methods are available for both phase or frequency data as described below. The best method depends on the quality of the fit, which can be judged by the randomness of the residuals.

<u>Data</u>	<u>Method</u>	<u>Noise Model</u>
Phase	Quadratic Fit	W PM
Phase	Avg of 2nd Diff's	RW FM
Phase	3-Point Fit	W & RW FM
Phase	Linear	Frequency Offset
Phase	Avg of 1st Diff's	Frequency Offset
Phase	Endpoints	Frequency Offset
Freq	Linear Fit	W FM
Freq	Bisection Fit	W & RW FM
Freq	Log Fit	Stabilization
Freq	DiffusionFit	Diffusion

- **Phase Data**

Three drift methods are available to analyze frequency drift in phase data:

1. The first is a least squares quadratic fit to the phase data:

$$x(t) = a + bt + ct^2, \text{ where } y(t) = x'(t) = b + 2ct, \text{ slope} = y'(t) = 2c.$$

This method is optimum for white PM noise. It is the default, and is also the method used for drift analysis when plotting phase data.

2. The second method is the average of the 2nd differences of the phase data:

$$y(t) = [x(t+\tau) - x(t)]/\tau, \text{ slope} = [y(t+\tau) - y(t)]/\tau = [(t+2\tau) - 2x(t+\tau) + x(t)]/\tau^2.$$

This method is optimum for random walk FM noise.

3. The third method uses the 3 points at the start, middle and end of the phase data:

$$\text{slope} = 4[x(\text{end}) - 2x(\text{mid}) + x(\text{start})]/(M\tau)^2, \text{ where } M = \# \text{ data points.}$$

It is the equivalent of the bisection method for frequency data.

Frequency offset may be calculated in phase data by either of three methods:

1. The first (optimum for white PM noise) uses a least squares linear fit to the phase data:

$$x(t) = a + bt, \text{ where slope} = y(t) = b.$$

2. The second method (optimum for white FM noise) uses the average of the first differences of the phase data:

$$y(t) = \text{slope} = [x(t+\tau) - x(t)]/\tau.$$

3. The third method simply uses the difference between the first and last points of the phase data:

$$\text{slope} = y(t) = [x(\text{end}) - x(\text{start})] / (M-1), \text{ where } M = \# \text{ phase data points.}$$

This method is used mainly to match the two endpoints.

- **Frequency Data**

Four drift methods are also available for frequency data:

1. The first, the default, is a least squares linear regression to the frequency data:

$$y(t) = a+bt, \text{ where } a = \text{intercept, } b = \text{slope} = y'(t).$$

This is the optimum method for white FM noise.

- The second method computes the drift from the frequency averages over the first and last halves of the data:

$$\text{slope} = 2 [ y(2\text{nd half}) - y(1\text{st half}) ] / (N\tau), \text{ where } N = \# \text{ points.}$$

This bisection method is optimum for white and random walk FM noise.

- The third method, a log model of the form (see MIL-O-55310B):

$$y(t) = a - \ln(bt+1), \text{ where slope} = y'(t) = ab/(bt+1)$$

which applies to frequency stabilization. It is available in the Plot function for frequency data.

- The last frequency drift method is a diffusion ( $\sqrt{t}$ ) model of the form:

$$y(t) = a + b(t+c)^{1/2}, \text{ where slope} = y'(t) = \frac{1}{2} \cdot b(t+c)^{-1/2}.$$

It is also available in the Plot function for frequency data.

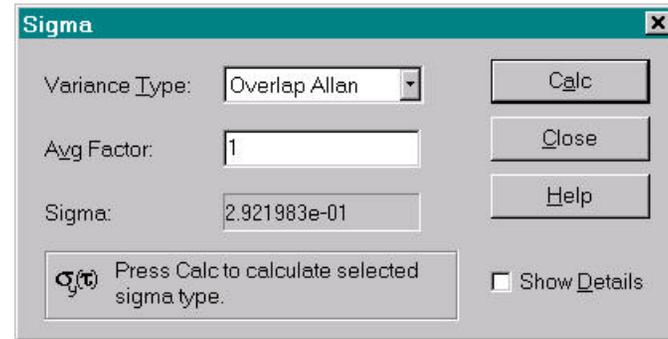
- Auto Calc**

A drift analysis can be done automatically when the Drift function is invoked by setting the AutoCalc item in the Configure function.

## Sigma Function

- Purpose**

Use the Sigma function to calculate Allan or Hadamard variance statistics for the selected portion of the current phase or frequency data at a single averaging time. (The Run function is available for performing these calculations over a range of averaging times.)



- Controls**

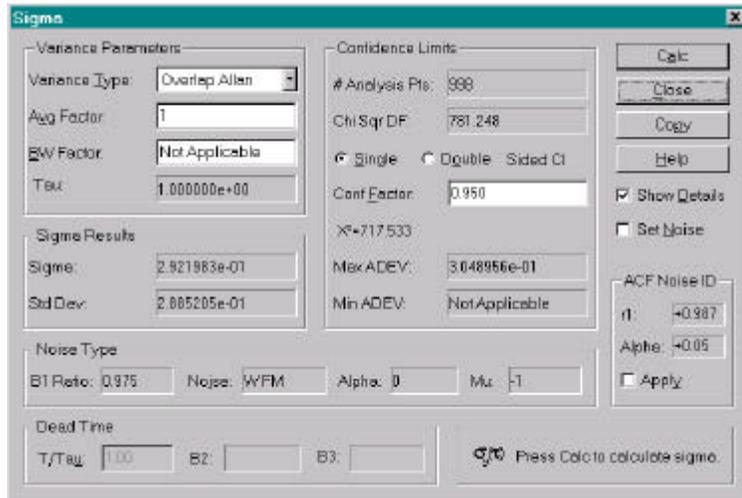
The basic Sigma dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
Variance <u>T</u> ype	Combo	Choose desired variance type.
Avg Factor	Edit	Enter averaging factor.
Sigma	Text	Display Sigma result.
Show <u>D</u> etails	Checkbox	Select detailed Sigma dialog box format.
<u>C</u> alc	Pushbutton	Calculate sigma.
<u>C</u> lose	Pushbutton	Close Sigma dialog box & restore previous data.
<u>S</u> top	Pushbutton	Abort TOTMDEV, TOTTDEV or MTIE calc.
<u>H</u> elp	Pushbutton	Invoke the Sigma help topic.

Activating the Show Details option adds the following controls to the Sigma dialog box:

<u>Control</u>	<u>Type</u>	<u>Description</u>
Var Params	Group	Variance parameters
<u>B</u> W Factor	Edit	Enter bandwidth factor.
Tau	Text	Display tau.
Sigma Results	Group	Sigma values
Std Dev	Text	Display standard deviation.
Noise Type	Group	Noise ID parameters
B1/R(n) Ratio	Text	Display B1 or R(n) ratio.
Noise	Text	Display estimated noise type.
Alpha	Text	Display estimated $\alpha$ value.
Mu	Text	Display estimated $\mu$ value.
Kn	Text	Display Kn value.

Dead Time	Group	Dead time parameters
T/Tau	Edit	Enter deadtime ratio.
B2	Text	Display B2 value.
B3	Text	Display B3 value.
Conf Limits	Group	Confidence limit parameters
# Analysis Pts	Text	Display # analysis points.
Max Sigma	Text	Display maximum sigma value.
Min Sigma	Text	Display minimum sigma value.
Copy	Pushbutton	Copy analysis results to clipboard.
Show Details	Checkbox	Deselect detailed Sigma dialog box format.
Set Noise	Checkbox	Set noise to user-selected type.
ACF Noise ID	Group	Lag 1 autocorrelation noise ID params
r1	Text	Lag 1 autocorrelation value
Alpha	Text	Estimated power law noise type
Apply	Checkbox	Use ACF noise ID method
Noise	Combo	Select noise type when Set Noise is checked.



### • Variance Types

The Allan and Hadamard variances are the principal method for describing the stability of a frequency source in the time domain. Those variances and other statistics may be calculated as shown in the table below:

Type	Variance	Abbreviation	Symbol
Allan	Allan	AVAR	$\sigma_y^2(\tau)$
Overlap Allan	Overlapping Allan	AVAR	$\sigma_y^2(\tau)$
Mod Allan	Modified Allan	MVAR	Mod $\sigma_y^2(\tau)$
Time	Time	TVAR	$\sigma_x^2(\tau)$
Hadamard	Hadamard	HVAR	$H\sigma_y^2(\tau)$
Overlap Had	Overlapping Hadamard	HVAR	$H\sigma_y^2(\tau)$

Total	Total Allan	TOTAVAR	$\sigma_{total}^2(\tau)$
Mod Total	Total Modified Allan	TOTMVAR	Mod $\sigma_{total}^2(\tau)$
Time Total	Total Time	TOTTVAR	$\sigma_x^2_{total}(\tau)$
TIE rms	RMS Time Interval Error	TIE rms	
MTIE	Maximum Time Interval Error	MTIE	
Thèo1	Theoretical Variance #1	Thèo1	

The Type column shows the variance types of the Sigma function list. These short original Stable32 names are retained to avoid breaking batch files. The currently accepted names and abbreviations for these variance types are shown in the Variance and Abbreviation columns. These are all functions of the averaging time,  $\tau$ , and are usually expressed as their square-roots (the deviation or sigma),  $\sigma_y(\tau)$ , Mod  $\sigma_y(\tau)$ ,  $\sigma_x(\tau)$ ,  $H\sigma_y(\tau)$ ,  $\sigma_{total}(\tau)$ , Mod  $\sigma_{total}(\tau)$ , and abbreviated as ADEV, MDEV, TDEV, HDEV, TOTADEV, TOTMDEV, and TOTTDEV respectively. The TIE rms, MTIE, and Thèo1 statistics apply only to phase data.

### • Operation

The Sigma function begins with the selection of the **Variance Type** and the **Avg Factor**, the # of  $\tau_0$  measurement intervals averaged to form the averaging time  $\tau$ . Entering an 'm' or 'M' will give the maximum allowable averaging factor. The **Calc** button will then calculate the desired **Sigma** value. The **Show Details** checkbox will change the dialog box to a more detailed format as described below. The **Help** button invokes this Help screen, while the **Close** button closes the dialog box. A change in either the **Variance Type** or **Avg Factor** clears the results.

### • Detailed Sigma Dialog Box

The detailed Sigma dialog box provides additional options and results. The **Variance Parameters** group box adds a **BW Factor** entry and a display of the **Tau** value. The **BW Factor** is equal to  $2\tau f_h \tau_0$ , where  $f_h$  is the upper cutoff frequency of the measuring system in Hz, and applies only to flicker PM noise. The **Sigma Results** group box adds a display of the **Std Dev** (standard deviation). The **Noise Type** group box shows several parameters associated with the power law noise type, based on the **B1** or **R(n) Ratio**. This ratio is used to estimate the Noise type and corresponding **Alpha**, **Mu**, and (for normal AVAR) **Kn** values as described below. The  $\alpha$  value is the exponent of the  $S_y(f)$  noise process, and the  $\mu$  is the  $\sigma_y(\tau)$  log-log slope. See the Noise Processes and Noise Spectra Help screens for more information about those subjects. **Confidence Interval** information is displayed in the center group box. The Kn value is used to determine the confidence limits for the non-overlapping sigma. Chi-squared statistics are used for the overlapping, modified, time, and total sigmas, and the  $X^2$  value is displayed. A bias correction is applied automatically as needed to the total sigma and Thèo1 value. Activating the **Set Noise** checkbox opens a **Noise** combo box that allows selection of a particular noise type, or the **r1** and **Alpha** lag 1 ACF noise ID parameters are shown, and may be **Appl(y)**ied to determine the noise type.

- **Allan Variance**

The non-overlapping Allan variance function calculates the original  $\sigma_y(\tau)$  and the standard deviation, and uses their ratio to estimate the noise type and the confidence interval of the Allan deviation.

- **Overlapping Allan Variance**

The overlapping Allan variance function also calculates and displays  $\sigma_y(\tau)$ , using fully-overlapping samples. It also estimates the noise type, and uses that to determine the # of degrees of freedom. It then establishes single or double-sided confidence intervals for the Allan deviation based on a certain confidence factor.

- **Modified Allan Variance**

The modified Allan variance function requests the averaging factor and bandwidth factor for the measuring system,  $BW = 2\pi f_h \tau_0$ , where  $f_h$  is the upper cutoff frequency in Hz. It then calculates  $\sigma_y(\tau)$  and Mod  $\sigma_y(\tau)$ , and uses their ratio to estimate the noise type. This function then continues with the determination of single or double-sided confidence intervals for the modified Allan deviation for the estimated noise type and desired confidence factor.

- **Time Variance**

The time variance is a measure of time stability based on the modified Allan variance.

- **Hadamard Variance**

The non-overlapping Hadamard variance uses 3 (rather than 2) fractional frequency samples, which allows it to converge for  $\alpha = -3$  (Flicker Walk FM) and  $\alpha = -4$  (Random Run FM) noises, and causes it to be unaffected by linear frequency drift.

- **Overlapping Hadamard Variance**

The overlapping Hadamard variance uses fully overlapping samples to calculate the Hadamard variance.

- **Total Allan Variance**

The total variance uses a double reflection method to form a  $\sigma_y(\tau)$  estimate with better confidence at long averaging times.

- **Total Modified Variance**

The total modified variance also uses a double reflection method to form a mod  $\sigma_y(\tau)$  estimate with better confidence at long averaging times.

- **Total Time Variance**

The total time variance is a measure of time stability based on the total modified variance.

- **TIE rms**

The rms time interval error, TIE rms, is the rms value of the time deviations. It can be calculated for phase data only.

- **MTIE**

The maximum time interval error, MTIE, is maximum value of the time deviation over a moving window. It can be calculated for phase data only.

- **Thêo1**

Thêo1 is a special purpose statistic used to estimate the Allan variance at large averaging factors.

- **Power Law Noise Type Estimates**

The power law noise type is estimated for each of the variance types by using either the lag 1 autocorrelation, the B1 ratio (the ratio of the standard variance to the Allan variance for zero dead time), or R(n) (the ratio of the modified to normal Allan variances). The bandwidth factor  $BW=2\pi f_h \tau_0$  (where  $f_h$  is the measuring system bandwidth in Hz) applies to the R(n) calculation for flicker PM noise.

- **Bias Correction**

Bias corrections are applied to the total variances and Thêo1 based on the estimated noise type.

- **Confidence Intervals**

Single or double-sided confidence intervals are determined using the # of analysis points, estimated noise type and the equivalent # of  $X^2$  degrees of freedom for the particular variance type. These are then used to calculate the corresponding maximum (and optionally) minimum sigma values.

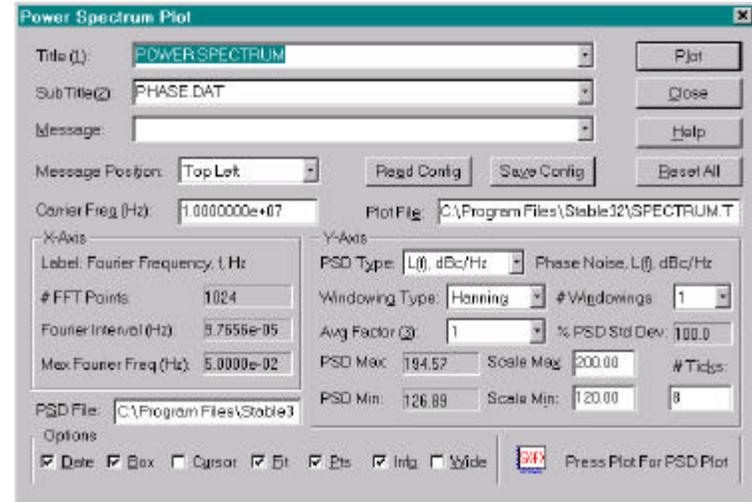
- **Auto Calc**

A sigma calculation can be done automatically when the Sigma function is invoked by setting the AutoCalc item in the Configure function.

## Power Function

- **Purpose**

Use the Power function to plot the power spectral density of the current phase or frequency data.



- **Controls**

The Power dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
<u>T</u> itle	Combo	Edit or select plot title. Default is POWER SPECTRUM.
<u>S</u> ubTitle	Combo	Edit or select plot subtitle. Default is filename.
<u>M</u> essage	Combo	Edit or select plot message.
<u>M</u> sg <u>P</u> osition	List	Select message position. Default is top left.
<u>C</u> arrier Freq	Edit	Enter carrier frequency. Default is 10 MHz.
<u>X</u> -Axis	Group	X-axis parameters.
<u>L</u> abel	Text	X-axis label (not editable).
<u>#</u> FFT Points	Text	# Fourier frequency points. Always the power-of-2 $\leq n/2$ , where $n = \#$ data points.
<u>F</u> ourier Interval	Text	Spacing between frequency points $= 1/n\tau$ .
<u>M</u> ax Fourier Freq	Text	Highest frequency point $= 1/2\tau$ .
<u>R</u> ead Config	Pushbutton	Read plot configuration data.
<u>S</u> ave Config	Pushbutton	Save plot configuration data.

Plot File	Edit	Enter file plot name. Default is SPECTRUM.TKF
Y-Axis	Group	Y-axis parameters
PSD Type	List	Select PSD type: $S_x(f)$ , $S_y(f)$ or $\mathcal{E}(f)$ for phase data, or $S_y(f)$ for freq data.
Windowing Type	List	Select FFT windowing type: None, (Rectangular), Hanning or Hamming.
# Windowings	List	Select # windowings (0-3).
PSD File	Edit	Enter PSD filename.
Options	Group	Plot options.
Date	Checkbox	Show date on plot.
Box	Checkbox	Draw box around plot.
Cursor	Checkbox	Activate mouse cursor for plot.
Fit	Checkbox	Draw least-squares power law line through octave band PSD averages.
Pts	Checkbox	Show FFT points on plot.
Info	Checkbox	Show windowing and carrier frequency information on plot.
Wide	Checkbox	Draw wide lines on plot.
Plot	Pushbutton	Plot power spectrum.
Close	Pushbutton	Close Power Spectrum dialog box.
Help	Pushbutton	Invoke the Power help topic.
Reset All	Pushbutton	Reset all plot attributes to defaults.

- **Operation**

Enter the desired **Title**, **Sub Title**, **Message** and message **Position**. Select the desired **PSD Type**, **Windowing Type** and **# Windowings**. Enter the **Carrier Frequency** value for  $S_y(f)$  or  $\mathcal{E}(f)$  PSD types. Use the **Date**, **Box**, **Cursor**, **Fit**, **Pts** and **Info** checkboxes to select the desired options. Then press **Plot** to plot the power spectrum, or **Close** to close the Power Spectrum dialog box.

Display of the FFT points and octave-band fits are controlled by the Pts and Fit boxes, and the Info box controls the display of the FFT information. Checking the Wide option causes the stability data to be plotted with wide lines. This can be useful for producing plots with better visibility for presentations. The Cursor option will activate the mouse during a plot to draw lines and display plot coordinates. Use the Esc key to deactivate it before closing the plot.

Zero padding is used, if necessary, to raise the # of time series data points to the next larger power-of-two.

The PSD results (Fourier frequency and Log PSD) are written to a data file with the default filename PSD.dat.

See the Special Characters help topic for information about adding subscripts, superscripts, Greek letters, etc. to power spectrum plot titles and subtitles.

- **PSD Windowing**

Data windowing is the process of applying a weighting function that falls off smoothly at the beginning and end to avoid spectral leakage in an FFT analysis. Without windowing, bias will be introduced that can severely restrict the dynamic range of the PSD result. The Stable32 program offers a choice of four windowing types: None, Hanning, Hamming and Multitaper. The classic Hanning and Hamming windows can be applied for 1 to 3 times.

- **PSD Averaging**

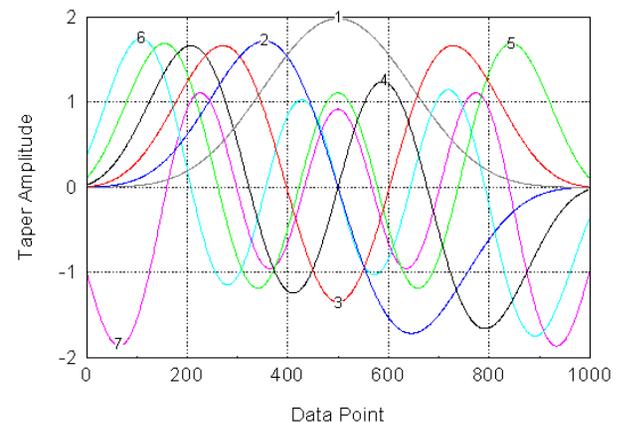
Without filtering or averaging, the variances of the PSD results are always equal to their values regardless of the size of the time domain data set. More data provides finer frequency resolution, not lower noise (while the data sampling time determines the highest Fourier frequency). Without averaging, for white noise, each spectral result has only 2 degrees of freedom. Some sort of filtering or averaging is usually necessary to provide less noise in the PSD results. This is accomplished in Stable32 by dividing the data into sections, performing an FFT analysis on each section separately, and then averaging those values to obtain the final PSD result.

The averaging factor can be selected in integer power-of-2 increments, which improves the PSD standard deviation by the square root of the averaging factor. The tradeoff in this averaging process is that each section of the data is shorter, yielding a result with coarser frequency resolution that does not extend to as low a Fourier frequency.

- **Multitaper PSD Analysis**

The Stable32 Power function also includes a multitaper PSD analysis method that offers a better compromise between bias, variance and spectral resolution. Averaging is accomplished by applying a set of orthogonal windowing (tapering) functions called discrete prolate spheroidal sequences (DPSS) or Slepian functions to the entire data array. An example of seven of these functions for order J=4 is shown below.

Slepian DPSS Taper Functions, J=4, #=7



The 1st function resembles a classic window function, while the others sample other portions of the data. The higher windows have larger amplitude at the ends that compensates for the denser sampling at the center. These multiple tapering functions are defined by two parameters, the order of the function, J, which affects the resolution bandwidth, and the # of windows, which affects the variance. A higher J permits the use of more windows without introducing bias, which provides more averaging (lower variance) at the expense of lower spectral resolution. This is shown in the following table:

Order J	# Windows
2.0	1-3
2.5	1-4
3.0	1-5
3.5	1-6
4.0	1-7
4.5	1-8
5.0	1-9

The resolution BW is given by  $2J/Nt$  where N is the # of data points sampled at time interval t. An adaptive algorithm is used to weight the contributions of the individual tapers for lowest bias. The multitaper PSD has a flat-topped response for discrete spectral components that is nevertheless narrower than an averaged periodogram with the same variance. It is therefore particularly useful for examining discrete components along with noise.

• **PSD Data File**

A data file comprising rows of 2 columns of Fourier frequency and PSD values is written each time a power spectrum plot is generated. This file, which has a default name PSD.DAT, must be manually renamed before the next plot to save it. An example of the first 5 lines of a PSD.dat file is shown below:

```
2.44140625e-04 -2.44245834e+01
4.88281250e-04 -2.40925655e+01
7.32421875e-04 -2.40541973e+01
9.76562500e-04 -2.39492950e+01
1.22070313e-03 -2.39989624e+01
```

• **Notes**

The carrier frequency parameter applies only to the Sf(f) and £(f) PSD types. The # of Fourier frequency points is always the power of 2 greater than or equal to 1/2 of the # of time domain data points, n. The spacing between Fourier frequency points is  $1/nt$ , and the highest Fourier frequency is  $1/2t$ . If averaging is done, the value of n is reduced by the averaging factor. The PSD fit is a least-squares power law line through octave-band PSD averages.

The Stable32 Power function is intended primarily for the analysis of noise (not discrete components), and includes the quantitative display of power law noise in common PSD units, with fits to integer power law noise processes. Amplitude corrections are made for the noise response of the windowing

functions. The amplitude of discrete components should be increased by the log of the BW (Fourier frequency spacing in Hz), which is a negative number for typical sub-Hz bandwidths.

The cursor can be used to help identify the frequency and amplitude of discrete components on the PSD plot. Press the ESC key to deactivate it before closing the plot.

• **Caution**

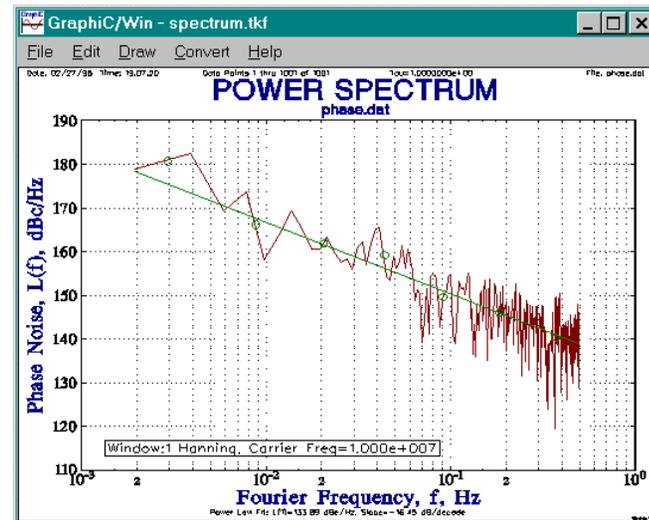
The PSD results are based on time units of seconds.

• **References**

1. W.H. Press, B.P. Flannery, S.A. Teukolsky and W.T. Vetterling, *Numerical Recipes in C*, Cambridge University Press, 1988, ISBN 0-521-35465-X, Chapter 12.
2. D.J. Thomson, "Spectrum Estimation and Harmonic Analysis", *Proc. IEEE*, Vol. 70, No. 9, Sept. 1982, pp. 1055-1096.
3. D.B. Percival and A.T. Walden, *Spectral Analysis for Physical Applications*, Cambridge University Press, 1993, ISBN 0-521-43541-2.
4. J. M. Lees and J. Park, "A C-Subroutine for Computing Multi-Taper Spectral Analysis", *Computers in Geosciences*, Vol. 21, 1995, pp. 195-236.
4. Help File for the *AutoSignal* program for spectral analysis, AISN Software, Inc., 1999.

• **Example**

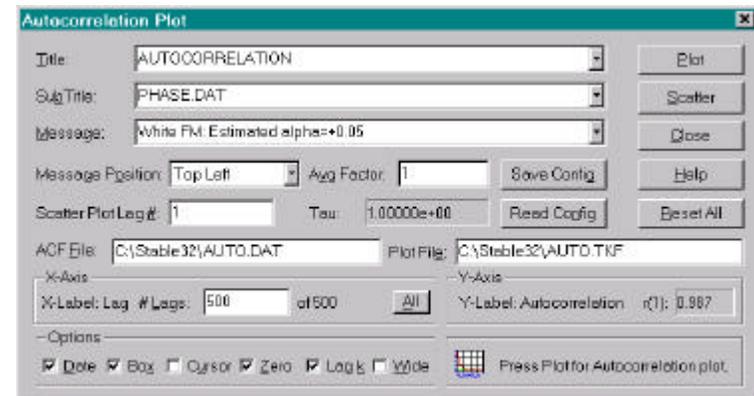
An example of a power spectrum plot is shown below:



## Autocorrelation Function

- **Purpose**

Use the Autocorrelation function to plot the autocorrelation function (ACF) of the current phase or frequency data. The primary purpose of the Autocorrelation function is to provide insight into the degree of correlation or non-whiteness of the phase or frequency fluctuations, and to provide an estimate of the power law noise type.



- **Controls**

The Autocorrelation dialog box contains the following controls:

Dialog Box Control	ALT Key	Control Type	Control Description
Plot	P	Pushbutton	Plot autocorrelation.
Scatter	S	Pushbutton	Plot lag scatter plot.
Title	T	Combo	Edit or select plot title.
SubTitle	B	Combo	Edit or select plot subtitle.
Message	M	Combo	Edit or select plot message.
Position	O	List	Select message position.
Avg Factor	V	Edit	Avg factor for autocorrelation.
Tau		Text	Averaging time for autocorrelation.
Read Config	N	Pushbutton	Read config data from INI file.
Save Config	G	Pushbutton	Save configuration data to INI file.
ACF File	F	Edit	Enter ACF data filename.
Plot File	E	Edit	Enter ACF plot filename.
Lag #	#	Edit	Scatter plot lag #.
X-Axis		Groupbox	X-axis parameters.
X-Label		Text	X-axis label = Lag.
# Lags	L	Edit	# of autocorrelation lag points.
Y-Axis		Groupbox	Y-axis parameters.
Y-Label		Text	Y-axis label = Autocorrelation.
Options		Groupbox	Plot options.

Date	D	Checkbox	Show date on plot.
Box	X	Checkbox	Draw box around plot.
Cursor	U	Checkbox	Activate mouse cursor for plot.
Zero	Z	Checkbox	Draw horizontal line at zero.
Lag k	K	Checkbox	Show lag k scatter plot.
Wide	W	Checkbox	Use wide lines for spectrum plot.
Reset All	R	Pushbutton	Reset all parameters to defaults.
Help	H	Pushbutton	Invoke this help topic
Close	C	Pushbutton	Close ACF dialog box

- **Operation**

Enter the desired plot **Title**, **SubTitle**, **Message**, **Message Position** and **Options**, and select the desired **Avg Factor** and **# Lags**. Then press **Plot** to plot the autocorrelation sequence or **Scatter** to plot a lag scatter plot for the selected lag #. When done, press **Close** to close the Autocorrelation dialog box.

Autocorrelation plot configurations may be read, saved and reset with the **Read Config**, **Save Config** and **Reset All** buttons.

The lag # to be used for the scatter plot (normally 1) can be entered in the **Scatter Plot Lag #** box.

Display of the zero line is controlled by the **Zero** checkbox, and the **Lag k** checkbox selects the insertion of a lag k scatter plot. Checking the **Wide** option causes the stability data to be plotted with wide lines. This can be useful for producing plots with better visibility for presentations. The **Cursor** option will activate the mouse during a plot to draw lines and display plot coordinates. Use the Esc key to deactivate it before closing the plot.

- **Noise Identification**

The Stable32 Autocorrelation function provides an estimate of the power law noise type (white, flicker, random walk, flicker walk, or random run) for the particular data type (phase or frequency) that is displayed as a message on the plot. This estimate includes both the name of the closest power law noise type and the estimated alpha value, based on the lag 1 autocorrelation value, and is available for data sets of 30 and larger. The estimated alpha value is particularly helpful for mixed noises because it indicates the approximate proportions of the two dominant noise types (e.g. alpha=1.50 indicates an equal mixture of white and flicker PM noise).

- **Lag Scatter Plot**

The **Lag k** checkbox inserts a scatter plot of the phase or frequency data plotted against itself with a lag of k. The data at time t+k is plotted on the y-axis versus the value at time t on the x-axis. This plot is another way of showing the degree of correlation in the data, and the slope of a linear fit to these points is closely related to the lag k autocorrelation.

The lag scatter plot is a cluster of uncorrelated points for white noise, and has a linear shape for the more correlated flicker and random walk noises. It will assume a circular form for noise having a sinusoidal component when the lag # is at or near one-quarter of its period. Boxes showing the one, two and three sigma error bounds are drawn on the lag scatter plot with dashed lines. Not all of these boxes may be visible. The lag scatter plot can take a considerable time to draw, and the normal filled dot symbol is replaced with a cross when the # of analysis points is 10,000 and higher. The subtitle of the lag scatter plot is the same as the autocorrelation plot.

- **ACF Data File**

A data file comprising rows of 2 columns of lag # and ACF values is written each time a dynamic stability plot is generated. This file, which has a default name AUTO.DAT, and must be manually renamed before the next plot to save it. An example of the first 5 lines of an AUTO.dat file is shown below:

```
0  9.99654114e-01
1 -1.59891486e-01
2  5.67231514e-02
3 -4.17460054e-02
4 -4.83987387e-03
```

- **Notes**

The autocorrelation sequence is calculated using the Fast Fourier Transform (FFT), which produces autocorrelation points at lags up to one-half of the data length. The phase or frequency data is averaged by the entered averaging factor to the corresponding tau before the autocorrelation is calculated. Gaps in the data are filled with interpolated values before the analysis, which can result in linear features on the lag scatter plot for a group of gaps. The autocorrelation plot can be restricted to fewer points to better show values at smaller lags. The autocorrelation sequence is most useful for determining the non-whiteness of data or residuals, for detecting periodic components, and for identifying the dominant power law noise type. The autocorrelation results are available in a data file for further analysis. Deselect the inserted lag scatter plot to print or convert an autocorrelation plot, or use the Play program to replay and plot or convert an autocorrelation plot with the inserted lag scatter plot.

## Histogram Function

- **Purpose**

Use the Histogram function to plot a histogram of the current phase or frequency data.

- **Controls**

The Histogram dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
Plot	Pushbutton	Plot histogram.
<u>T</u> itle	Combo	Enter or select plot title. Default is HISTOGRAM.
<u>S</u> ub Title	Combo	Enter or select plot subtitle. Default is filename.
<u>M</u> essage	Combo	Enter or select plot message. Default is Mean Removed value.
<u>M</u> sg <u>P</u> osition	List	Select message location. Default is top left.
<u>P</u> lot <u>F</u> ilename	Edit	Enter plot filename. Default is HISTO.TKF.
X-Axis	Group	X-axis parameters.
Label	Text	X-axis label (not editable).
Y-Axis	Group	Y-axis parameters.
Label	Text	Y-axis label (not editable).
Options	Group	Plot options.
<u>D</u> ate	Checkbox	Show date on plot.
<u>B</u> ox	Checkbox	Draw box around plot.
<u>C</u> ursor	Checkbox	Activate mouse cursor for plot.
<u>F</u> it	Checkbox	Draw normal curve fit on plot.
<u>F</u> ill	Checkbox	Fill histogram bars.
<u>N</u> ormalize	Checkbox	Normalize data for zero mean. If grayed, normalization is required.

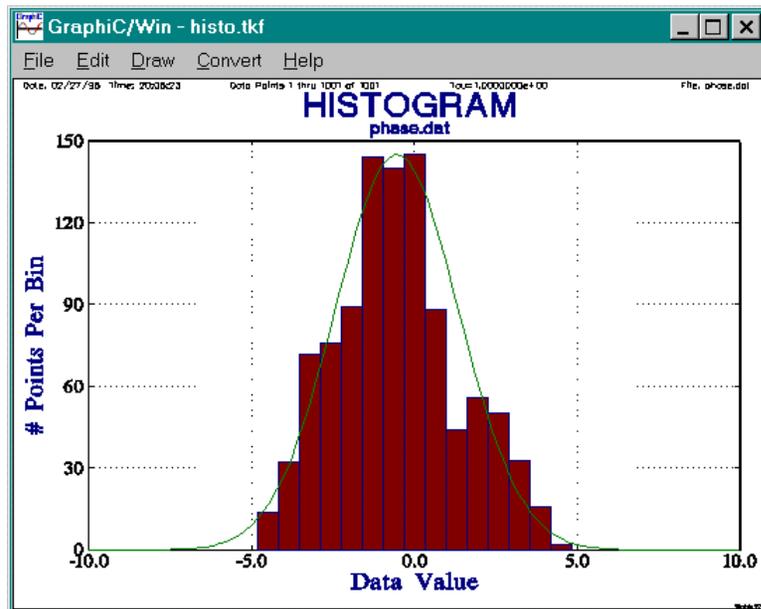
Mean Removed	Text	Value of average value removed from data.
Read Config	Pushbutton	Read plot configuration data from STABLE32.INI.
Save Config	Pushbutton	Save plot configuration data in STABLE32.INI.
Reset All	Pushbutton	Reset all plot attributes to defaults.
Close	Pushbutton	Close the Histogram dialog box.
Help	Pushbutton	Invoke the Histogram help topic.

- **Operation**

Enter the desired **Title**, **Sub Title**, **Message**, message **Position** and **Options**. Check **Normalize** to remove the mean value from the data. This may be required if the data has relatively small variations around a large average value. Normalization does not permanently affect the data in memory. If the Normalize checkbox is grayed, normalization is required and will be done automatically and the mean value removed will be displayed as the default message. The original data is not affected. The Fill control can be used to eliminate the filled histogram bars for better printouts. Then press **Plot** to plot the histogram, or **Close** to close the Histogram dialog box.

- **Example**

An example of a histogram plot is shown below:



## Plot Function

- **Purpose**

Use the Plot function to control the plotting of the current phase or frequency data.



- **Controls**

The Plot dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
<u>P</u> lot	Pushbutton	Plot the current phase or frequency data.
<u>O</u> ptions	Pushbutton	Select plot options.
<u>L</u> ines	Pushbutton	Add lines to the plot.
<u>N</u> otes	Pushbutton	Add notes to the plot.
<u>C</u> lose	Pushbutton	Abort the Plot function & close dialog box.
<u>H</u> elp	Pushbutton	Invoke the Plot help topic.

- **Operation**

Simply press the button for the desired plotting function.

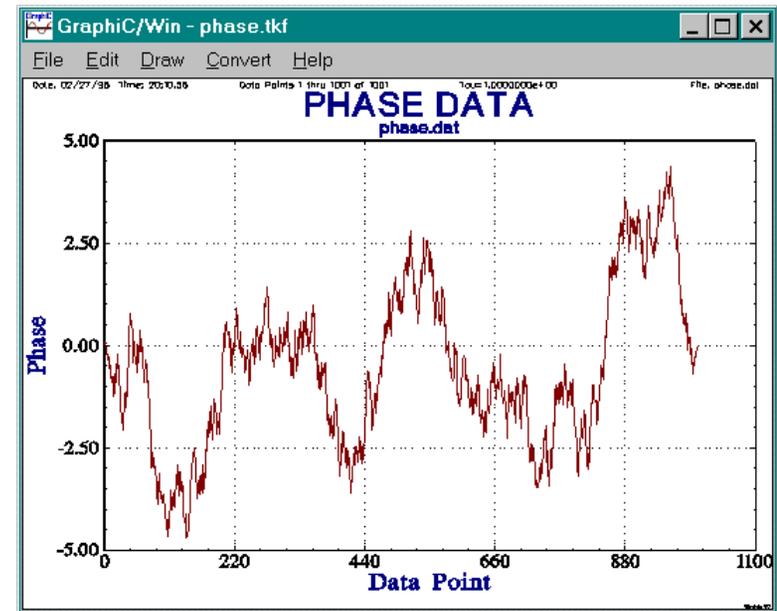
- **Auto Plot**

Plotting can be done automatically, bypassing the Plot function dialog box, by setting the AutoPlot item in the Configure function.

## Data Plot Function

- **Purpose**

Use the Data Plot function to generate a plot of the current phase or frequency data.



## Plot Options Function

- **Purpose**

Use the Plot Options function to control the attributes of the data plot.

- **Controls**

The Plot Options dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
Title	Edit	Enter plot title.
SubTitle	Edit	Enter plot subtitle.
Message	Edit	Enter plot message.
Position	Combo	Choose message position.
X-Axis	Group	X-Axis parameters.
Label	Edit	Enter x-axis label.
Keep 8	Checkbox	Keep X-axis label for next plot.
Offset	Edit	Enter x-axis scale offset.
Multiplier	Edit	Enter x-axis scale multiplier.
# X Divs	Edit	Enter # x-axis major scale divisions
Start 3	Edit	Enter starting point #
End 4	Edit	Enter ending point #
Y-Axis	Group	Y-Axis parameters
Label	Edit	Enter y-axis label.
Offset	Edit	Enter y-axis scale offset.
Multiplier	Edit	Enter y-axis scale multiplier.
Data Max	Text	Display maximum data value.
Scale Max	Edit	Enter y-axis scale maximum.
Data Min	Text	Display minimum data value.
Scale Min	Edit	Enter y-axis scale minimum.
# Y Divs	Edit	Enter # y-axis major scale divisions.
# Ticks	Edit	Enter # y-axis minor scale ticks.

Options	Group	Plot options (curve fits).
Type	Combo	Choose line options.
Date	Checkbox	Show date on plot.
Box <u>5</u>	Checkbox	Draw box around plot.
Fit <u>6</u>	Checkbox	Show fit parameters at bottom of plot.
Wide	Checkbox	Use wide lines for data plot
Cursor <u>7</u>	Checkbox	Activate mouse cursor.
Plot File	Edit	Enter plot filename.
Read <u>1</u>	Pushbutton	Read plot configuration data
Save <u>2</u>	Pushbutton	Save plot configuration data
Reset All	Pushbutton	Reset all plot attributes to defaults.
OK	Pushbutton	Set plot attributes & close dialog box.
Cancel	Pushbutton	Abort changes & close dialog box.
Help	Pushbutton	Invoke the Plot Options help topic.

### • Operation

Enter the desired plot attributes and press **OK** to accept them, or **Cancel** to abort. Press **Reset All** to reset all attributes to their default values.

Both of the Stable32 phase and frequency data plots have defaults that give reasonable plots automatically. Common changes are to modify the **Subtitle**, **Title**, to add a **Message**, and to Scale and re **Label** the axes. Standard and previous titles, subtitles and messages may be selected from their combo box controls. Several curve fits are available from the **Options** box, as described in the Drift function. The x-axis is rescaled automatically when new time units are selected. Since the x-axis (time) scale is determined automatically, use the **# X Divs** to adjust an awkward x-axis scale. It is common to adjust the y-axis **Multiplier** (e.g. 1e+12) when the units (e.g. pp10<sup>12</sup>) are changed. When this done for frequency data, entering a pp10 label automatically adjusts the multiplier value. The plot options, when accepted with OK, “stick” between dialog box openings. User-entered titles, subtitles and messages are pushed onto their respective list boxes with a CR, Tab or other focus change. Up to six such options are automatically saved, recalled and linked between the various plot option dialog boxes. The plot options may be saved and recalled in the STABLE32.INI configuration file between program runs with the **Save Config** and **Read Config** buttons. These configuration data contain additional parameters (such as axis labels and scale factors) for each plot type. The configuration file is read and saved automatically if the corresponding box is checked in the Configure dialog box. The Keep box may be checked to retain the x-axis settings for future plots. When using the default directory, new Stable32 GraphiC plot files (PHASE.TKF, FREQ.TKF, SPECTRUM.TKF and HISTO.TKF) will overwrite the previous ones. If it is desired to keep one of these files, it is important to rename it either from within the corresponding dialog box before plotting or manually from the command line after the plotting operation. Plot files may also be sent to another directory by changing the Plot Filename (or Plot File) field in the Plot Options dialog box. When the SmartScale configuration option is on, the default y-axis phase

data plot scales are, whenever possible, set to standard units like microseconds and nanoseconds, and the frequency scales to pp10<sup>XX</sup> notation. Checking the Cursor option activates crosshairs on the plot whose position is indicated at the right end of the GraphiC menu bar (please ignore the strange characters). This option also activates a line drawing feature. The right mouse button starts a line and the right button ends it. The line can be extended with another right button click, or a new line started with the left button. The lines can be deleted with the Del key. Press the Esc key to deactivate the cursor function. Checking the Wide option causes the data to be plotted with wide lines. This can be useful for producing plots with better visibility for presentations. See Appendix II for information about adding subscripts, superscripts, Greek letters, etc. to data plot titles, subtitles and labels.

### • Time Scales

The X-Axis Label control offers several choices for the data plot x-axis time scale, as shown in the following table.

<u>X-Axis Label</u>	<u>Units</u>	<u>Remarks</u>
Data Point	#	Data point #
Data Point x1000	#	Data point #/1000
Time, Seconds	Seconds	Tau
Time, Minutes	Minutes	Tau/60
Time, Hours	Hours	Tau/3600
Time, Days	Days	Tau/86400
Time, Weeks	Weeks	Tau/604800
DOY	Days	Same as days if no timetags. DOY with year if timetags.
MJD	Days	Same as days if no timetags. MJD - coarse MJD if timetags.
Timetag Point	#	Regularized data point #

Selecting the x-axis label automatically sets the x-axis offset and multiplier. If there are timetags, they are automatically used to annotate the year and MJD. The units for the x-axis offset are those of the data points. The x-axis start and end controls may be used to select a portion of the data for plotting. The Timetag Point option is available only for data with associated timetags. It offers a preview of how the data will look if regularized to show gaps indicated by the timetags. The Timetag Point x-axis scale automatically changes to the x1000 mode as necessary. The plot lines are not available for this option, but the fit lines may be used. Use the Regularize function to prepare the data for further analysis.

### • Curve Fits

The Options Type control offers several choices for curve fits, as shown in the following table. The Fit checkbox controls the display of the fit parameters at the bottom of the plot.

<u>Fit Type</u>	<u>Data Type</u>	<u>Description</u>
None	Both	No curve fit
Linear	Frequency	y=a+bt

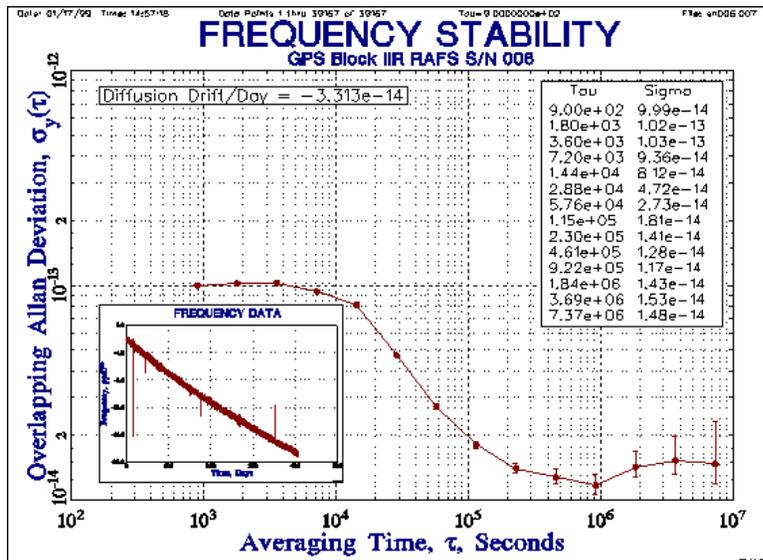
Quadratic	Phase	$x=a+bt+ct^2$
Log	Frequency	$y=a+\ln(bt+1)+c$
Diffusion	Frequency	$y=a+b(t+c)^{1/2}$
Average	Both	$x=\bar{x}, y=\bar{y}$

### Plot Messages

Several standard messages (such as the minimum, maximum and average values, the frequency drift, and any offset removed or line fits) are available by exposing the plot message drop down list.

### Plot Inserts

Mini-plots may be inserted at bottom left of phase data, frequency data and stability plots. These plots must have been generated previously, and named p\_insert.tkf, f\_insert.tkf and s\_insert.tkf respectively. For example, a small plot of the frequency data can be inserted into a stability plot, as shown in the sample below:



### Cursor

Checking the cursor option activates crosshairs on the plot whose position is indicated at the right end of the GraphC menu bar (please ignore the strange characters). This option also activates a line drawing feature. The right mouse button starts a line and the right button ends it. The line can be extended with another right button click or a new line started with the left button. The lines can be deleted with the Del key. Press the Esc key to deactivate the cursor function.

### Wide Lines

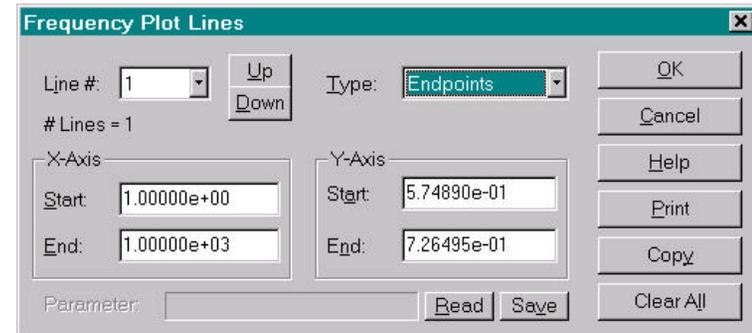
Checking the **Wide** option causes the data to be plotted with wide lines with better visibility for presentations.

## Plot Lines Function

### Purpose

Use the Plot Lines function to add lines to a data plot. The possible line types are as follows:

Line Type	Description	Parameter
None	No line	None
Endpoints	Line between certain endpoint values	None
Regression	Linear (frequency data) or quadratic (phase data) regression line	Frequency slope
Average	Horizontal line at average value	Average
Marker	Vertical line at certain x-axis value	None



### Controls

The Plot Line dialog box contains the following controls. Some parameters do not apply to all line types.

Control	Type	Description
Line #	Edit	Enter # of line to edit.
Up	Pushbutton	Increment line #.
Down	Pushbutton	Decrement line #.
Type	Combo	Select line type.
X-Axis	Group	X-Axis line parameters.
Start	Edit	Enter x-axis start of line.
End	Edit	Enter x-axis end of line.
Y-Axis	Group	Y-axis line parameters.
Start	Edit	Enter y-axis start of line.
End	Edit	Enter y-axis end of line.
Parameter	Text	Display line parameter.
# Lines	Text	# of lines entered.

<u>R</u> ead	Pushbutton	Read line information from INI file.
<u>S</u> ave	Pushbutton	Save line information in INI file.
<u>O</u> K	Pushbutton	Accept changes & close dialog box.
<u>C</u> ancel	Pushbutton	Abort changes & close dialog box.
<u>H</u> elp	Pushbutton	Invoke the Plot Lines help topic.
<u>P</u> rint	Pushbutton	Print line parameters.
<u>C</u> opy	Pushbutton	Copy line information to clipboard.
<u>C</u> lear <u>A</u> ll	Pushbutton	Clear all lines.

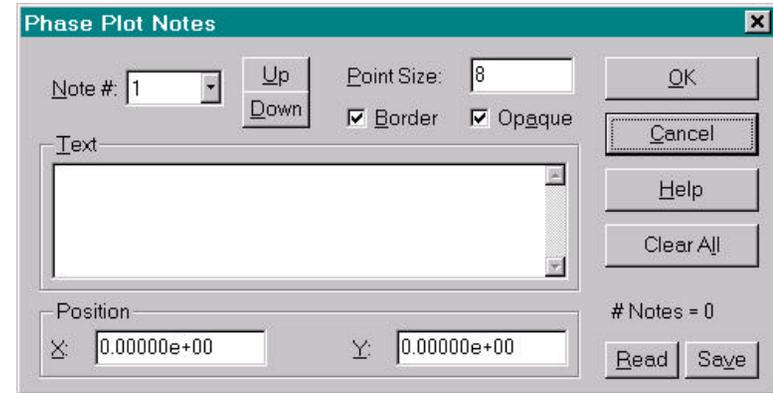
- **Operation**

Select the desired **Line #** and **Type**. The Enter the **X-Axis Start**, **X-Axis End**, **Y-Axis Start** and **Y-Axis End** values as applicable, and press **OK** to accept them, or **Cancel** to abort. Press **Clear All** to clear all lines, **Copy** to copy the line parameters to the clipboard, or **Print** to print the line parameters.

## Plot Notes Function

- **Purpose**

Use the Plot Annotations function to add annotations to a data plot.



- **Controls**

The Plot Annotations dialog box contains the following controls:

<u>C</u> ontrol	<u>T</u> ype	<u>D</u> escription
<u>N</u> ote #	Edit	Enter the # of the note to edit.
<u>U</u> p	Pushbutton	Increment the note #.
<u>D</u> own	Pushbutton	Decrement the note #.
<u>P</u> oint Size	Edit	Enter the point size for the note.
<u>T</u> ext	Edit	Enter the text of the note.
<u>P</u> osition	Group	Note position values.
<u>X</u>	Edit	Enter the x-position of the note.
<u>Y</u>	Edit	Enter the y-position of the note.
<u>#</u> Notes	Text	# notes entered.
<u>B</u> order	Checkbox	Draw a border around the note.
<u>O</u> paque	Checkbox	Hide grid in note area.
<u>C</u> lear <u>A</u> ll	Pushbutton	Clear all notes.
<u>R</u> ead	Pushbutton	Read line information from INI file.
<u>S</u> ave	Pushbutton	Save line information in INI file.
<u>O</u> K	Pushbutton	Accept the settings & close dialog box.
<u>C</u> ancel	Pushbutton	Abort the changes & close dialog box.
<u>H</u> elp	Pushbutton	Invoke the Plot Notes help topic.

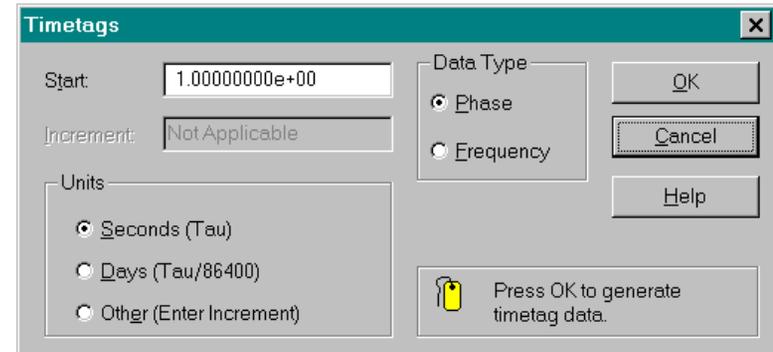
- **Operation**

Enter the desired plot annotations and press **OK** to accept them, or **Cancel** to abort. Select the **Border** and **Opaque** options as desired. Press **Clear All** to clear all annotations, or use the **Read** and **Save** buttons to retrieve and store note information. The **X** and **Y** positions are in user units that are the same as the plot scales.

## Timetags Function

- **Purpose**

Use the Timetags function to generate regular timetags for evenly spaced data. Such timetags can be useful for examining and editing data, especially if the starting time is known. Timetags are generated for the entire data array regardless of the analysis limits.



- **Controls**

The Timetags dialog box contains the following controls:

<u>C</u> ontrol	<u>T</u> ype	<u>D</u> escription
<u>S</u> tart	Edit	Enter starting timetag value.
<u>I</u> ncrement	Edit	Enter timetag increment (if applicable).
<u>U</u> nits	Group	Timetag units choices.
<u>S</u> econds	Radiobutton	Use $\tau$ timetag units in seconds.
<u>D</u> ays	Radiobutton	Use $\tau/86400$ timetag units in days.
<u>O</u> ther	Radiobutton	Use arbitrary timetag units & increment.
<u>D</u> ata Type	Group	Controls to select data type.
<u>P</u> hase	Radiobutton	Select phase data.
<u>F</u> requency	Radiobutton	Select frequency data.
<u>O</u> K	Pushbutton	Perform Timetag function & close dialog box.
<u>C</u> ancel	Pushbutton	Abort the Timetag function & close dialog.
<u>H</u> elp	Pushbutton	Invoke the Timetags help topic.

- **Operation**

Select the applicable data type (**Phase** or **Frequency**). Enter the desired **Start** timetag value and units (**Seconds**, **Days** or **Other**). If **Other** (arbitrary) units is chosen, enter the desired timetag **Increment**. Otherwise, if **Seconds** or **Days** units are chosen, the increment will be determined by the **t** value. Press **OK** to generate the timetags or **Cancel** to abort the Timetags function.

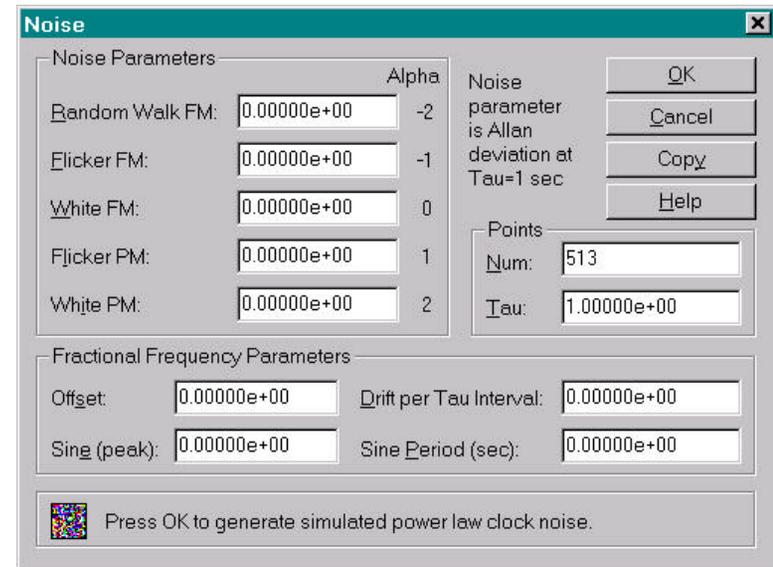
- **Caution**

Use caution when generating timetags for data that is not regular. Do not confuse these artificially generated timetags with actual ones. Such timetags can be useful to identify the position of data interruptions, but gaps should be manually inserted into the data to provide regularity, and the timetags should then be cleared. The Regularize function cannot accomplish this with artificially generated timetags.

## Noise Function

- **Purpose**

Use the Noise function to function to generate simulated power law clock noise.



- **Controls**

The Noise dialog box contains the following controls:

<b>Control</b>	<b>Type</b>	<b>Description</b>
Noise Params	Group	Power law noise parameters.
RW FM	Edit	Enter random walk FM noise coefficient $\sigma(1)$ .
Flicker FM	Edit	Enter flicker FM noise coefficient $\sigma(1)$ .
White FM	Edit	Enter white FM noise coefficient $\sigma(1)$ .
Flicker PM	Edit	Enter flicker PM noise coefficient $\sigma(1)$ .
White PM	Edit	Enter white PM noise coefficient $\sigma(1)$ .
Points	Group	# points and $\tau$ interval.
Num	Edit	Enter # points (default=513).
Tau	Edit	Enter $\tau$ interval, seconds (default=1).
Freq Params	Group	Fractional frequency parameters.
Offset	Edit	Enter fractional frequency offset.
Drift	Edit	Enter fractional frequency drift per $\tau$ interval.
Sine (peak)	Edit	Enter peak value of sine term.
Sine Period	Edit	Enter period of sine term
OK	Pushbutton	Perform the Noise function & close dialog box.
Cancel	Pushbutton	Abort the Noise function & close dialog box.
Copy	Pushbutton	Copy noise parameters to clipboard.
Help	Pushbutton	Invoke the Noise help topic.

- **Operation**

Enter the **Number** of data points wanted and their **Tau** interval. Then enter the  $\sigma(1)$  **Noise Parameters** for the desired power law clock noise. At least one noise parameter must be non-zero. Next enter any fractional frequency **Offset** and **Drift**, and/or the **peak** value and **Period** of any **Sine** term, that may be wanted. Then press **OK** to generate the simulated clock noise, or **Cancel** to abort.

- **F FM Noise Generation**

The long-term phase excursions of the generated flicker FM noise are too small on average (see Reference 2 below). This does not affect the Allan deviation results, and can be avoided by generating twice as many points as needed and using only the second half of the output.

- **FW FM and RR FM Noise Generation**

The more divergent flicker walk FM ( $a=3$ ) and random run FM ( $a=4$ ) power law noise types may be generated by using the  $1/f^2$  spectral property of a frequency to phase conversion. For example, to generate RR FM noise, first generate a set of RW FM noise and save the resulting phase data. Then read this RW FM phase data as frequency data and convert it to a new set of RR FM phase data.

- **Sine Generation**

The Sine (peak) and Sine Period controls allow entry of the fractional frequency amplitude and period (seconds) of a sinusoidal component to be added to the simulated noise. A finite amount of noise must still be included (to avoid zero sigma). These parameters are useful for simulating the effect of a periodic disturbance such as temperature or vibration. The phasing is that of a positive sinusoid on the fractional frequency data. No sinusoidal component is added unless both the peak value and period are non-zero.

- **Memory Usage**

Noise generation is a memory-intensive process that requires about 48 bytes of peak memory usage per data point. Large data sets can therefore involve relatively slow virtual (hard drive swap) memory.

- **Caution**

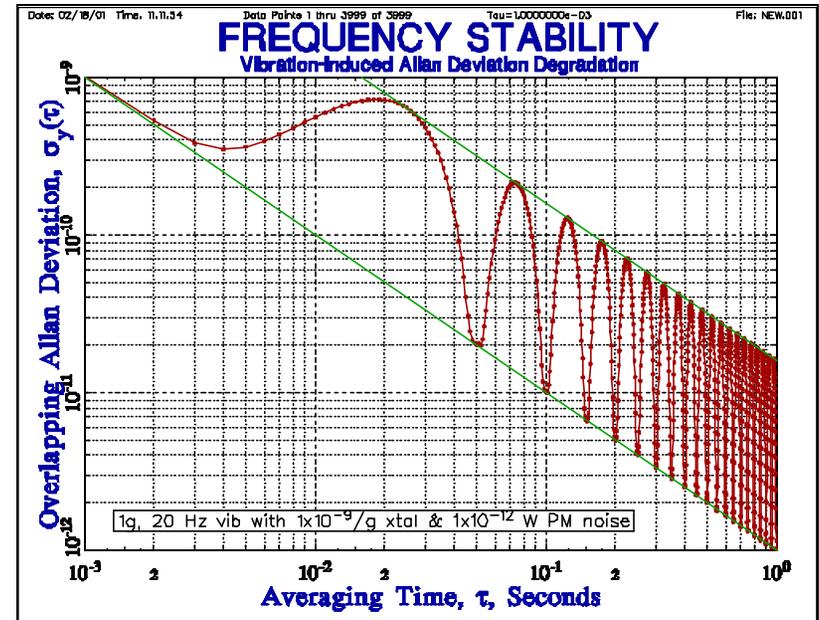
The results from the Noise function overwrite the current data in the phase and frequency arrays.

- **References**

1. N.J. Kasdin and T.Walter, "Discrete Simulation of Power Law Noise", *Proceedings of the 1992 IEEE Frequency Control Symposium*, pp. 274-283, May 1992.
2. C.A. Greenhall, "FFT-Based Methods for Simulating Flicker FM", *Proceedings of the 34th Annual Precise Time and Time Interval (PTTI) Systems and Applications Meeting*, December 2002, pp. 481-492.

- **Example**

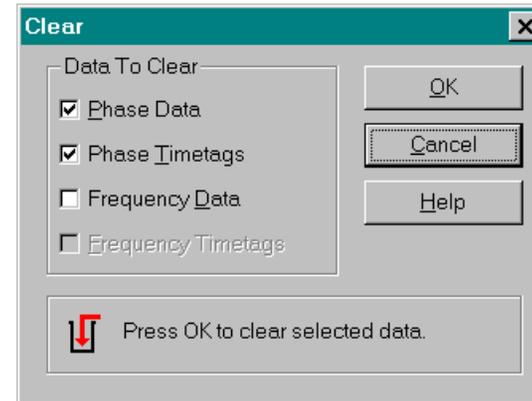
An example of the frequency stability of simulated data with a combination of white PM noise and a sinusoidal term is shown below:



## Clear Function

- **Purpose**

Use the Clear function to clear phase or frequency data or the corresponding timetags. Note: It is not necessary to clear the data array before reading another data file.



- **Controls**

The Clear dialog box contains the following controls:

<b>Control</b>	<b>Type</b>	<b>Description</b>
Data To Clear	Group	Choices for data to clear.
Phase Data	Checkbox	Clear phase data.
Phase Timetags	Checkbox	Clear phase timetags.
Frequency Data	Checkbox	Clear frequency data.
Frequency Timetags	Checkbox	Clear frequency timetags.
OK	Pushbutton	Perform the Clear function & close dialog box.
Cancel	Pushbutton	Abort the Clear function & close dialog box.
Help	Pushbutton	Invoke the Clear help topic.

- **Operation**

Select the **Phase Data**, **Phase Timetags**, **Frequency Data** or **Frequency Timetags** to be cleared. Then press **OK** to clear the selected data, or **Cancel** to abort.

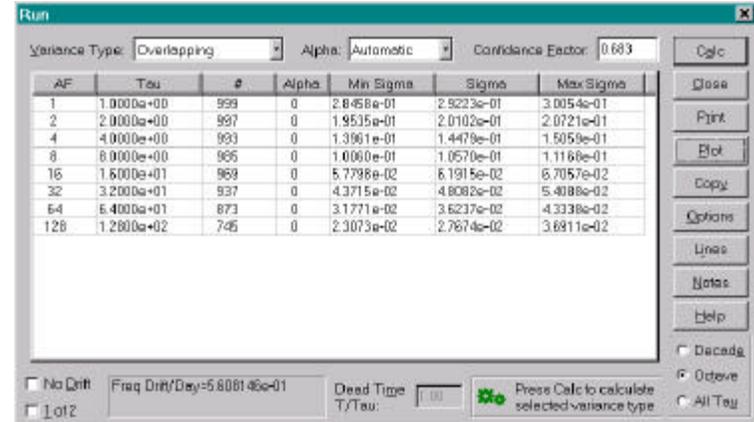
- **Caution**

There is no confirmation after OK is pressed - the selected data is cleared and cannot be recovered. It is possible to clear the phase or frequency data and retain the corresponding timetags, but this is an unusual thing to do.

## Run Function

- **Purpose**

Use the Run function to calculate (and optionally plot or print) Allan, total or Hadamard variance statistics for the selected portion of the current phase or frequency data over a series of averaging times. The averaging times may be selected in octave or sub-decade increments, or at every possible tau out to a reasonable fraction of the record length. (The Sigma function is available for performing this calculation at a single averaging time.) Frequency drift may be removed, and 1 of 2 unit scaling may be applied as options.



- **Controls**

The Run dialog box contains the following controls:

<u>C</u> ontrol	<u>T</u> ype	<u>D</u> escription
Variance Type	Combo	Choose variance type.
Alpha	Combo	Choose automatic or selected alpha value.
Conf Factor	Edit	Enter desired confidence factor.
Results	Group/Text	Display results of stability run.
Drift/Day	Text	Display linear frequency drift.
Plot	Pushbutton	Plot stability results.
Copy	Pushbutton	Copy analysis results to clipboard.
Options	Pushbutton	Open Run Options dialog box.
Lines	Pushbutton	Open Run Lines dialog box.
Notes	Pushbutton	Open Run Notes dialog box.
Print	Pushbutton	Print stability results.
No Drift	Checkbox	Remove linear frequency drift from data.
1 of 2	Checkbox	Scale results for 1 of 2 identical units.
T/Tau	Edit	Enter dead time ratio.
Decade	Radiobutton	Use decade tau increments.
Octave	Radiobutton	Use octave tau increments.
All Tau	Radiobutton	Use all tau increments.
Calc[1]	Pushbutton	Calculate table of stability statistics.

<u>S</u> tart AF[1]	Edit	Enter starting averaging factor for run.
<u>S</u> top[2]	Pushbutton	Stop run.
<u>R</u> esume[3]	Pushbutton	Resume run.
<u>C</u> lear	Pushbutton	Abort run.
<u>C</u> lose	Pushbutton	Close dialog box & restore original data.
<u>H</u> elp	Pushbutton	Invoke the Run help topic.

[1]=Before run, [2]=During run, [3]=When stopped.

### • Operation

Select the desired **Variance Type**, **Alpha**, and (if applicable) **Conf Factor**. Optionally check **Remove Drift** and **1 of 2**, choose **Decade** (1, 2, 4, 10), **Octave** (1, 2, 4, 8) or **All Tau** tau increments (no error bars are displayed for the **All Tau** option), or enter a **Start AF**. Enter the **Dead Time** ratio if it is not unity. Then Press **Calc** to perform a stability run, and **Plot, Copy** or **Print** the results as desired. To quit, press **Close**.

The default **Alpha** selection is Auto, which will automatically estimate the noise type. If the noise type is known a priori, then choose one of the specific alpha values. In either case, the alpha value for each averaging factor (AF) is used to determine each confidence interval. A progress bar is displayed during a TOTMDEV, TOTTDEV or All Tau analysis. Those runs may be **Stopped** and then **Resume**d. While stopped, the current results may be printed, plotted or copied. To abort a stopped run, press the **Clear** button.

### • Stability Results Table

The table of results from the stability run contains the following items:

<u>Item</u>	<u>Description</u>
AF	Averaging factor, $\tau/\tau_0$
Tau	Averaging time, $\tau$
#	# analysis points
Alpha	Estimated noise exponent, $\alpha$
Min Sigma	Lower confidence limit (as applicable)
Sigma	Nominal sigma
Max Sigma	Upper confidence limit (as applicable)

### • Stability Results Plot

Use the Plot button, along with the Options, Lines and Notes functions to plot the results of a stability run. Many aspects of the plot can be customized as desired. The individual stability points of a run are indicated by closed (normal) or open (modified) circles, which are automatically made smaller when the number of points exceeds 100 (e.g. an all tau run).

### • Variance Types

See the Sigma function for a list of the variance types that can be found with the Run function. In addition, both the overlapping and modified Allan deviations can be calculated and plotted during the same run.

### • Drift Removal

The Run function allows linear frequency drift to be removed from phase or frequency data before performing the stability analysis. Simply check the No **Drift** checkbox. The drift/day will be reported, but the original phase or frequency data is not changed.

### • 1 of 2 Scaling

The Run function allows scaling the results for one of two identical units. Simply check the **1 of 2** checkbox to scale the resulting sigma values by  $\sqrt{2}$ .

### • Dead Time

Bias correction for measurement dead time. **Dead\_Time** is available for the normal and overlapping Allan variances when the noise type is manually selected. Enter the T/Tau dead time ratio, where tau is the measurement time and T is the time between measurements. Dead time corrections are made using the B2 and B3 bias ratios, which handle cases of single and multiple (distributed) measurement dead time respectively.

### • Overlap & Thèo1 Hybrid

The Overlap & Thèo1 hybrid (called ThèoH) provides stability data over the entire range of averaging times from the measurement interval to 75% of the record length. Manually enter the type of power-law noise that applies to the data to invoke the appropriate bias correction, or use the Auto/BR (Bias Removed) feature to automatically determine and remove the bias correction.

### • All Tau

The All Tau option is useful as a means of spectral analysis to identify periodic components in time domain stability data. Without the Many Tau option, it will calculate the stability at every possible tau value, which can take a very long time. With the Many Tau option, which may be selected in the Configure Function, a selectable subset of the possible tau values will be used to provide a quasi-uniform distribution of points on the stability plot, which can provide much faster calculating, plotting and printing. The default setting for the approximate number of points is 500, which may be set with the ManySize parameter in the [Preferences] section of the Stable32.ini configuration file.

### • Dead Time

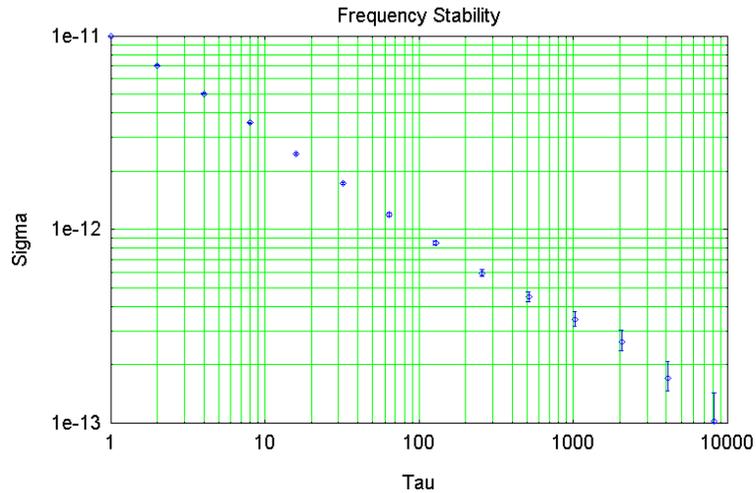
Bias correction for measurement dead time is available for the normal and overlapping Allan variances when the noise type is manually selected. Enter the T/Tau dead time ratio, where tau is the measurement time and T is the time between measurements. Dead time corrections are made using the B2 and B3 bias ratios, which handle cases of single and multiple (distributed) measurement dead time respectively.

### • Auto Calc

A run can be done automatically when the Run function is invoked by setting the AutoCalc item in the Configure function.

- **Stability File**

A stability file (with a default name of sigma.tau) is written to disk after each run. This file contains sigma, tau and errorbar data as described in the Read Function help topic, and may be used to document or replot the stability results. It may even be used to create a stability plot with another program (such as Gnuplot, as shown below).



The following wgnupl32.exe Gnuplot script was used for this stability plot:

```
set grid xtics ytics mxtics mytics 2
set nokey
set logscale
set title "Frequency Stability"
set xlabel "Tau"
set ylabel "Sigma"
set zero 1e-20
plot 'sigma.tau' using 1:3:4:5 with yerrorbars
```

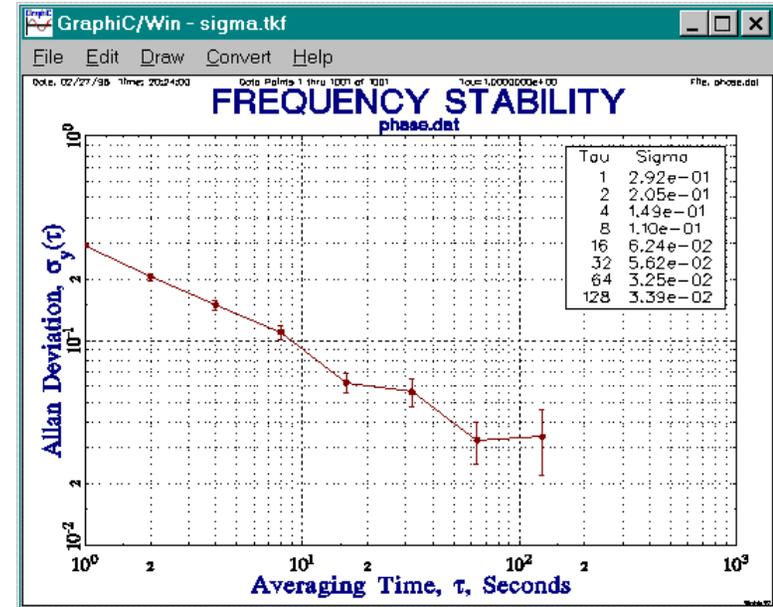
- **Notes**

Printing of the stability results is not available for the All Tau option. The TIE rms and MTIE statistics are available for phase data only, and do not include confidence intervals.

## Run Plot Function

- **Purpose**

Use the Run Plot function to generate a stability plot.



## Run Options Function

- **Purpose**

Use the Run Options function to control the attributes of the stability plot.

- **Controls**

The Run Options dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
<u>T</u> itle	Edit	Enter plot title.
<u>S</u> ub Title	Edit	Enter plot subtitle.
<u>M</u> essage	Edit	Enter plot message.
<u>M</u> sg Position	Combo	Choose message position.
<u>T</u> able <u>P</u> osition	Combo	Choose table position.
<u>X</u> -Axis	Group	X-Axis parameters.
<u>L</u> abel	Edit	Enter x-axis label.
<u>T</u> au <u>M</u> ax	Text	Display maximum tau value.
<u>S</u> cale <u>M</u> ax	Edit	Enter x-axis scale maximum.
<u>T</u> au <u>M</u> in	Text	Display minimum tau value.
<u>S</u> cale <u>M</u> in	Edit	Enter x-axis scale minimum.
<u>Y</u> -Axis	Group	Y-Axis parameters
<u>L</u> abel	Edit	Enter y-axis label.
<u>S</u> igma <u>M</u> ax	Text	Display maximum sigma value.
<u>S</u> cale <u>M</u> ax	Edit	Enter y-axis scale maximum.
<u>S</u> igma <u>M</u> in	Text	Display minimum sigma value.
<u>S</u> cale <u>M</u> in	Edit	Enter y-axis scale minimum.
<u>O</u> ptions	Group	Plot options.
<u>D</u> ate	Checkbox	Show date on plot.
<u>B</u> ox	Checkbox	Draw box around plot.

Error Bars <u>1</u>	Checkbox	Show error bars on plot.
Table <u>2</u>	Checkbox	Show stability table on plot.
Plot Filename	Edit	Enter plot file name.
Read Config	Pushbutton	Read plot configuration data from STABLE32.INI.
Save Config	Pushbutton	Save plot configuration data in STABLE32.INI.
Stability File	Edit	Enter stability file name.
Reset All	Pushbutton	Reset all plot attributes to defaults.
<u>O</u> K	Pushbutton	Set plot attributes & close dialog box.
<u>C</u> ancel	Pushbutton	Abort changes & close dialog box.
<u>H</u> elp	Pushbutton	Invoke the Plot Options help topic.

- **Operation**

Enter the desired plot attributes and press **OK** to accept them, or **Cancel** to abort. Press **Clear All** to reset all attributes to their default values.

- **Run Messages**

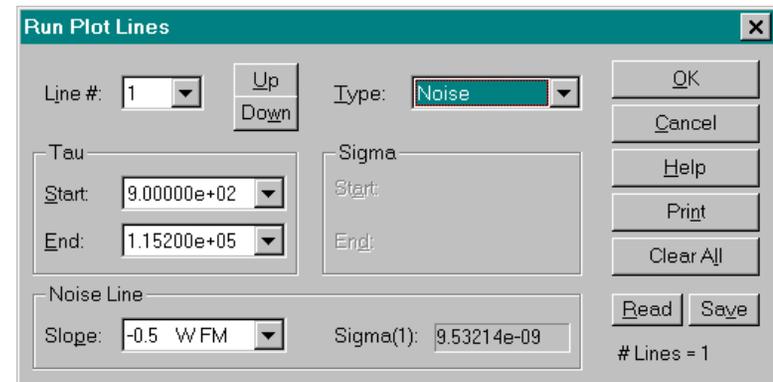
Messages describing all noise fits are available by exposing the run message drop down list.

## Run Lines Function

- **Purpose**

Use the Run Lines function to add lines to a stability plot. The possible line types are as follows:

<u>Line Type</u>	<u>Description</u>	<u>Parameter</u>
None	No line	None
Endpoints	Line between certain endpoint values	None
Horizontal	Horizontal line at certain sigma value	None
Vertical	Vertical line at certain tau value	None
Noise	Fit to certain power-law noise model	Sigma(1)



- **Controls**

The Run Line dialog box contains the following controls. Some parameters do not apply to all line types.

<u>Control</u>	<u>Type</u>	<u>Description</u>
Line #	Edit	Enter # of line to edit.
Up	Pushbutton	Increment line #.
Down	Pushbutton	Decrement line #.
Type	Combo	Select line type.
Tau	Group	X-Axis line parameters.
Start	Edit or Combo	Enter x-axis start of line.
End	Edit or Combo	Enter x-axis end of line.
Sigma	Group	Y-axis line parameters.
Start	Edit	Enter y-axis start of line.
End	Edit	Enter y-axis end of line.
Slope:	Combo	Noise type for fit
Sigma(1)	Text	Noise line fit parameter.
# Lines	Text	# of lines entered.

<u>C</u> lear <u>A</u> ll	Pushbutton	Clear all lines.
<u>R</u> ead	Pushbutton	Read line information from INI file.
<u>S</u> ave	Pushbutton	Save line information in INI file.
<u>P</u> rint	Pushbutton	Print line parameters.
<u>O</u> K	Pushbutton	Perform Run Line function & close dialog box.
<u>C</u> ancel	Pushbutton	Abort Run Line function & close dialog box.
<u>H</u> elp	Pushbutton	Invoke the Run Lines help topic.

• **Operation**

Enter the desired plot lines and press **OK** to accept them, **Print** to print them, or **Cancel** to abort. Press **Clear All** to clear all lines, or use the **Read** and **Save** buttons to retrieve and store line information. A particular line can be chosen with either a **Line #** entry or the **Up/Down** buttons. Up to 12 lines may be entered. The line type is then selected from the **Type** list, which varies according to the sigma type that is being run. Most of the line types require a combination of **Tau** and **Sigma Start** and **End** values, which may be either entered into an edit box or, in the case of noise lines, chosen from a list. Noise lines also require a **Slope** choice that is appropriate for the particular sigma type and power law noise process. The **Sigma(1)** value is shown for the noise fit. The total **#** of active **Lines** is also displayed. All plot lines are cleared if the Run conditions are changed.

• **Noise Lines**

The following power law noise lines are supported:

Noise	Alpha	Slope	Remarks
W PM	2	-1.0	Slope -1.5 for mod sigma
F PM	1	-1.0	
W FM	0	-0.5	
F FM	-1	0.0	
RW FM	-2	+0.5	
Aging		+1.0	
FW FM	-3	+1.0	Hadamard sigma only
RR FM	-4	+1.5	Hadamard sigma only

• **Masks**

Masks can be added to a stability plot to show specification limits and the like. They are defined in user-written \*.msk files per the format shown below. To add a mask, first select **Mask** from the list of line **Types**, and then choose the desired mask file from the **File** list that appears in the **Masks** group box. The mask and its label will then be included on the stability plot. The label can be edited on the Run Notes function.

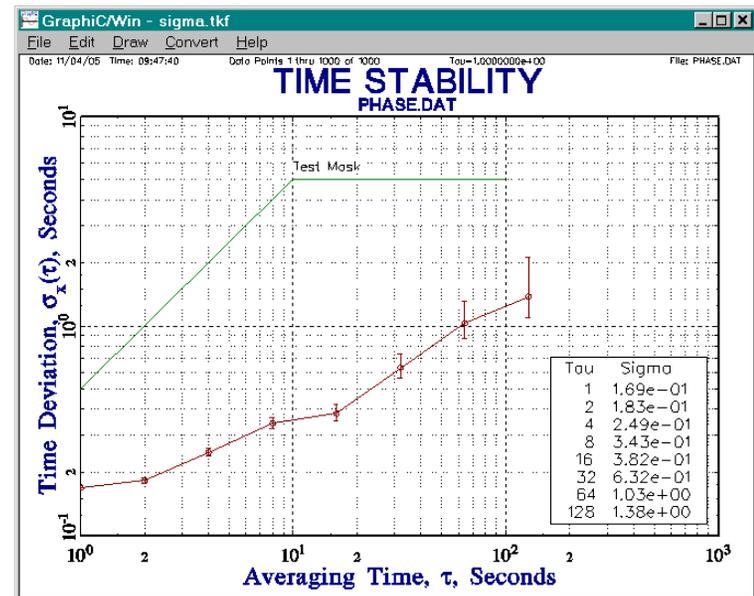
Mask files may be created with a text editor according to the following format:

1. No line shall exceed 80 characters.
2. An optional first line beginning with # as the first character can contain a comment (such as information about the name, origin & purpose of the mask) for documentation purposes only.

3. A mandatory header line with four fields: n x y l, where:  
int n = # mask data points  
double x = label x position in stability plot units  
double y = label y position in stability plot units  
char \*l = mask label string, which must be enclosed in double quotes.  
This label becomes a plot note and can be edited as a note before it appears on the plot. The intent is for the label to automatically identify the mask on the plot.
4. An arbitrary number of following lines containing x-y data in stability plot units defining the mask. These tau-sigma points must be in exactly n rows of two space-delimited columns. Plot lines will connect these points on the plot, so they need only define the corners of the mask on log-log axes. Otherwise, the points can be dense enough to draw a general smooth curve. The last line should be terminated with a CR/LF.
5. The mask filename shall have a \*.msk extension, and its name should relate in an obvious way to the mask label because it is the means for the user to select it.
6. The mask file must reside in the same directory as the Stable32.exe executable file that launched the program. Any number of such files can exist and are available for selection in the Run Plot Lines dialog.

Example of mask file "test.msk":

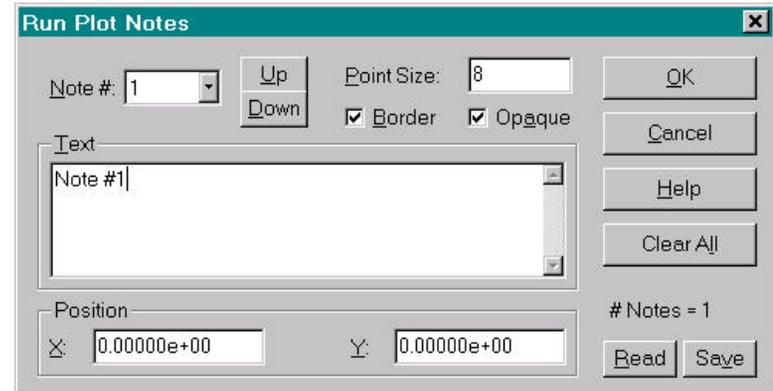
```
# Test mask to use with PHASE.DAT test data TDEV plot
3 10 5.5 "Test Mask"
1.5
10 5
100 5
```



## Run Notes Function

- **Purpose**

Use the Run Notes function to add annotations to a stability plot.



- **Controls**

The Run Notes dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
<u>Note #</u>	Edit	Enter the # of the note to edit.
<u>Up</u>	Pushbutton	Increment the note #.
<u>Down</u>	Pushbutton	Decrement the note #.
<u>Point Size</u>	Edit	Enter the point size for the note.
<u>Text</u>	Edit	Enter the text of the note.
<u>Position</u>	Group	Note position values.
<u>X</u>	Edit	Enter the x-position of the note.
<u>Y</u>	Edit	Enter the y-position of the note.
<u># Notes</u>	Text	# notes entered.
<u>Border</u>	Checkbox	Draw a border around the note.
<u>Opaque</u>	Checkbox	Hide grid in note area.
<u>Clear All</u>	Pushbutton	Clear all notes.
<u>Read</u>	Pushbutton	Read line information from INI file.
<u>Save</u>	Pushbutton	Save line information in INI file.
<u>OK</u>	Pushbutton	Accept the settings & close dialog box.
<u>Cancel</u>	Pushbutton	Abort the changes & close dialog box.
<u>Help</u>	Pushbutton	Invoke the Run Notes help topic.

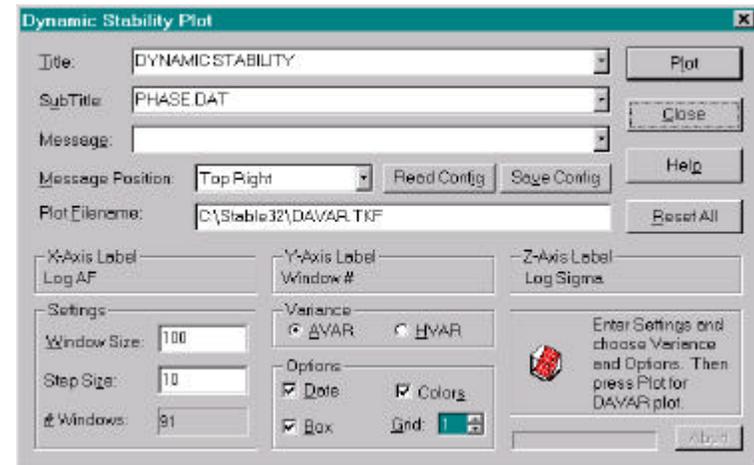
- **Operation**

Enter the desired plot annotations and press **OK** to accept them, or **Cancel** to abort. Press **Clear All** to clear all annotations.

## Dynamic Stability Function

- **Purpose**

Use the Dynamic Stability (DVAR) function to perform a dynamic Allan or Hadamard variance (DAVAR or DHVAR) stability analysis. A dynamic stability analysis is useful to show stability changes, nonstationarity, noise bursts and other aspects of clock behavior.



- **Controls**

The Dynamic Stability dialog box contains the following controls:

<u>Dialog Box</u>	<u>ALT</u>	<u>Control</u>	<u>Control</u>
<u>Control</u>	<u>Key</u>	<u>Type</u>	<u>Description</u>
Plot	L	Pushbutton	Plot dynamic stability.
Title	T	Combo	Enter or select plot title.
SubTitle	U	Combo	Enter or select plot subtitle.
Message	E	Combo	Enter or select plot message.
Msg Position	M	List	Select message location on plot.
Read Config	I	Pushbutton	Read config data from INI file.
Save Config	V	Pushbutton	Save configuration data to INI file.
Plot Filename	F	Edit	Enter plot filename.
X-Axis Label		Groupbox	X-axis label.
(Label)		Static Text	X-axis label (Log AF).
Y-Axis Label		Groupbox	Y-axis label.
(Label)		Static Text	Y-axis label (Window #).
Z-Axis label		Groupbox	Z-Axis label.
(Label)		Static Text	Z-Axis label (Log Sigma)
Settings		Groupbox	Window settings
Window Size	W	Edit	Enter analysis window size
Step Size	Z	Edit	Enter analysis step size
# Windows	#	Text	Display # analysis windows

Variance		Groupbox	Variance choices
AVAR	A	Radio Button	Allan variance
HVAR	H	Radio Button	Hadamard variance
Options		Groupbox	Plot options.
Date	D	Checkbox	Show date on plot.
Box	B	Checkbox	Draw box around plot.
Colors	S	Checkbox	Show multicolor surface plot
Grid	G	Listbox	Select color surface plot grid
(None)		Progress Bar	Show calculation progress
Abort	O	Pushbutton	Abort calculation
Close	C	Pushbutton	Close the Histogram dialog box.
Reset All	R	Pushbutton	Reset all parameters to defaults.
Help	P	Pushbutton	Invoke this help topic.

- **Operation**

Enter desired plot **Title**, **SubTitle**, **Message**, message **Position**, **Settings**, and **Options**. Then press **Plot** to produce a dynamic stability plot, or **Close** to close the dialog box.

The **Window Size** and **Step Size** set the width of the sliding analysis window and the # of data points that it advances for each step. Together, these parameters determine the # of analysis **Windows**, and are constrained to be within acceptable limits. A large # of analysis windows will require a long calculation time, which is shown in the **Progress** bar, and can be stopped with the **Abort** button, in which case the results obtained so far will be plotted.

A dynamic stability analysis can be performed with either the overlapping Allan (**AVAR**) or overlapping Hadamard (**HVAR**) variance. The latter is more appropriate when there is frequency drift in the data.

The resulting dynamic stability plot is saved as a \*.tkf file with a filename entered in the **Plot Filename** edit box (default DAVAR.TKF).

The plot options provide choices to put the **Date** and other information at the top of the page, draw a **Box** around the page, and use multiple **Colors** for the 3D surface plot. For color surface plots, the **Grid** control selects the grid type: 0=No Grid, 1=Full Grid, 2-9=# of x and y mesh lines skipped.

The preferred **Message Position** is Top Right, where there is the most space. The character sequence '\n' (backslash and lower case n) will insert a carriage return/line feed into the message to allow a longer, more vertical format.

The dynamic stability analysis options and settings can be saved in the Stable32.ini configuration file with the **Save Config** button, recalled with the **Read Config** button, and restored to their default settings with the **Reset** button.

- **Dynamic Stability Plot**

The dynamic stability plot is a 3D surface plot of ADEV or HDEV (log sigma) versus averaging time (log tau or log averaging factor, AF) as a function of time (window #). Sigma is calculated using fully overlapping samples at all possible averaging factors up to one-quarter of the window size. In the axes orientation used, the tau scale is opposite to that usually used (AF and tau increase toward the left), which is the convention for a dynamic stability plot, and has the advantage of best showing the sigma-tau surface that usually has lower sigma at longer tau. The y-axis and-z axis are labeled as log sigma and log AF respectively, and a log-log grid is shown in the back x plane, while the linear x-axis is labeled as window #. The dynamic stability plot includes annotations that show the sigma range, tau range, window parameters, and time span. A plot title can be included at the top, a sub-title at the bottom, and a message can be shown in any of 9 positions. Date, time, data analysis limits, and data filename can be included at the top, and an optional plot label (e.g. organization name) can be included at the bottom.

The dynamic stability 3D surface plot can be shown as a full red mesh or as a sigma-dependent multicolor display with or without an adjustable-density mesh. The mesh density also depends on the window and step sizes. For a large # of analysis windows, it is desirable to use the grid control to reduce the number of mesh points shown (this does not affect the underlying surface plot).

- **XYZ Data File**

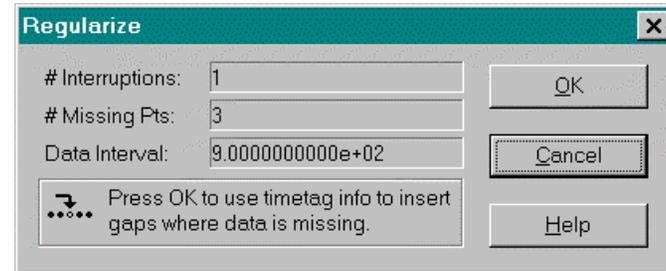
A data file comprising 3 columns of x (Averaging Factor), y (Window #) and z (Sigma) data named XYZ.dat is written to the Stable32 working directory each time a dynamic stability plot is generated. This file must be manually renamed before the next plot to save it. The XYZ.dat data is the actual stability analysis results before log scaling and interpolation. An example of the first 5 lines of an XYZ.dat file is shown below:

```
1.000000e+000 1.000000e+000 2.952219e-001
2.000000e+000 1.000000e+000 1.895182e-001
3.000000e+000 1.000000e+000 1.582051e-001
4.000000e+000 1.000000e+000 1.422213e-001
5.000000e+000 1.000000e+000 1.354947e-001
```

## Regularize Function

- **Purpose**

Use the Regularize function to find interruptions and missing points in time tagged phase or frequency data, and optionally insert gaps as required to provide a regular sequence of data points. The regularize function is not available for data that does not have timetags.



- **Controls**

The Regularize dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
# Interruptions	Text	Display # of interruptions in the data.
# Missing Pts	Text	Display total # missing points in the data.
Data Interval	Text	Display time interval between data points.
<u>O</u> K	Pushbutton	Perform the Regularize function & close dialog box.
<u>C</u> ancel	Pushbutton	Abort the Regularize function & close dialog box.
<u>H</u> elp	Pushbutton	Invoke the Regularize help topic.

- **Operation**

Observe the **# Interruptions**, **# Missing Pts** and **Data Interval** displays. If there are missing points that you wish to remove, verify that the **Data Interval** is correct. Then press **OK** to regularize the data. To abort, press **Cancel** (or **OK** if # interruptions = 0).

- **Caution**

The **Data Interval** must indicate the proper time spacing. It is determined by the difference between the first two timetags. If this value is not correct, edit the data file with a text editor before executing the Regularize function.

The Regularize function operates on the full data array.

## Domain Function

- **Purpose**

Use the Domain function to perform basic time and frequency domain conversions for power law noise processes.

Frequency Domain		Time Domain	
PSD Type:	L(f), dBc/Hz	Sigma Type:	Normal
SB Freq (Hz):	1.00000e+00	Tau (Sec):	1.00000e+00
Carrier Freq (MHz):	1.00000e+01	Avg Factor:	1
BW (Hz):	1.59155e+00	BW Factor:	10.000

Power Law Noise			Power Law Noise		
Type	dB/dec	PSD	Type	Mu	Sigma
RWF	-40:	None	RWF	+1:	0.00000e+00
EFM	-30:	None	EFM	0:	0.00000e+00
WFM	-20:	-80.0	WFM	-1:	1.00000e-11
FFM	-10:	None	FFM	-2:	0.00000e+00
WPM	0:	None	WPM	-2:	0.00000e+00
All:		-80.0	All:		1.00000e-11

Enter parameters to perform domain conversion for power law noise. An L(f) value of 0 denotes none. BW applies only to W & F PM noise.

- **Controls**

The Domain dialog box contains the following controls:

<u>C</u> ontrol	<u>T</u> ype	<u>D</u> escription
<u>S</u> igma Type	Combo	Choose sigma type (normal, modified or time).
<u>T</u> au (Sec)	Edit	Enter basic averaging time $\tau_0$ , in seconds.
<u>A</u> vg Factor	Edit	Enter averaging factor $AF = \tau_0/t$ , where $\tau$ is the analysis averaging time.
<u>B</u> W Factor	Edit	Enter bandwidth factor $2\pi f_h \tau_0$ , where $f_h$ is the upper cutoff frequency of the measuring system in Hz, and applies only to white and flicker PM noise.
<u>P</u> SD Type	Combo	Choose power spectral density (PSD) type.
<u>S</u> B Freq (Hz)	Edit	Enter the sideband frequency in Hz.
<u>C</u> arrier Freq	Edit	Enter the carrier frequency in MHz.
<u>B</u> W (Hz)	Edit	Enter the measuring system bandwidth in Hz.
<u>P</u> SD <u>R</u> WFM	Label	Use to select PSD entries from keyboard.
<u>P</u> SD <u>R</u> WFM	Edit	Enter or read the random walk FM noise PSD parameter.
<u>P</u> SD <u>F</u> FM	Edit	Enter or read the flicker FM noise PSD param.
<u>P</u> SD <u>W</u> FM	Edit	Enter or read the white FM noise PSD param.

PSD <u>F</u> PM	Edit	Enter or read the flicker PM noise PSD parameter.
PSD <u>W</u> PM	Edit	Enter or read the white PM noise PSD parameter.
All	Text	Total PSD value.
Sigma	Label	Use to select sigma entries from keyboard.
Sigma <u>R</u> WFM	Edit	Enter or read the random walk FM noise sigma parameter.
Sigma <u>F</u> FM	Edit	Enter or read the flicker FM noise sigma parameter.
Sigma <u>W</u> FM	Edit	Enter or read the white FM noise sigma parameter.
Sigma <u>F</u> PM	Edit	Enter or read the flicker PM noise sigma parameter.
Sigma <u>W</u> PM	Edit	Enter or read the white PM noise sigma parameter.
All	Text	Total sigma value.
Clear <u>A</u> ll	Pushbutton	Restore all parameters to their default values.
<u>C</u> lose	Pushbutton	End the Domain function & close dialog box.
Copy <u>I</u>	Pushbutton	Copy conversion results to clipboard.
<u>H</u> elp	Pushbutton	Invoke the Domain help topic.

### • Operation

The Domain function is a flexible spreadsheet-like tool that performs a variety of conversions as values are entered. Begin by entering values for the domain being converted from. Choose the desired **PSD** or **Sigma Type** and the associated basic parameters in the upper panels. These are the **SB** and **Carrier Freq** for the frequency domain or **Tau** and **Avg Factor** for the time domain. The **BW** and **BW Factor** applies only to white and flicker PM noise and may be entered later. Then enter the **PSD** or **Sigma** power law noise parameters for any combination of **RW FM**, **F FM**, **W FM**, **F PM** and **W PM** noise. The corresponding parameters for the other domain are updated automatically, as is the total noise for each domain. Besides these domain conversions, other common operations are changing the **PSD Type** to perform units conversions, changing the **SB Freq** to obtain the phase noise versus sideband frequency, and changing the **Avg Factor** to obtain sigma versus averaging time. In all cases, the overall display is automatically updated in a consistent manner. The Domain dialog box may be cleared to its default values with the **Clear All** button and closed with the **Close** button.

### • Caution

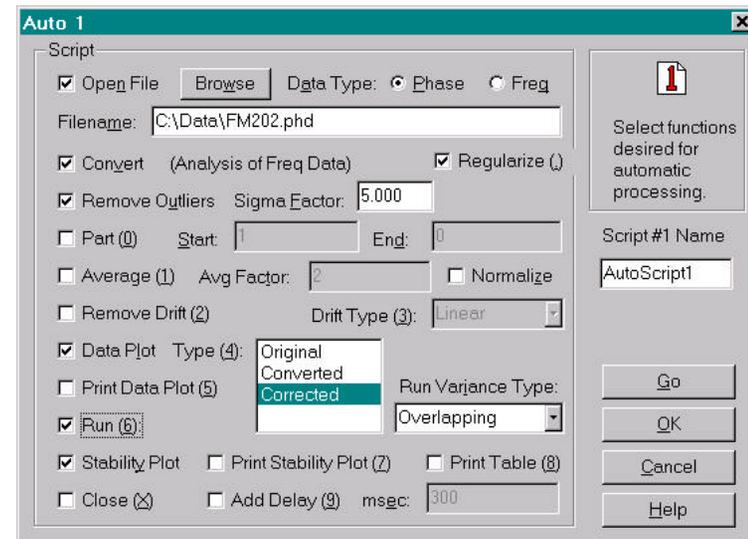
The domain conversions use simple closed-form expressions for the 5 most common power-law noise processes, and their summation depends on the noise PSD slope decreasing toward higher sideband frequencies. Better results may be obtained by numerical integration for real noise data, and other expressions are required to handle particular cases (such as a filter) where the noise slope increases above a certain SB frequency.

## Auto# Functions

### • Purpose

Use one of the Auto# functions (Auto1, Auto2 or Auto3) to automate a stability analysis. The Auto# functions execute automation scripts that describe the steps of a frequency stability analysis. These steps are arranged in a standard order, and can be individually selected or skipped. Plots and tables can be drawn and printed to show the results. The available options are as follows:

Step	Description	Options	Plot/Print
Open File	Open data file	Phase or freq data Browse for file	Original data
Convert	Convert data	To other analysis type	Converted data
Remove Outliers	Remove bad data	Frequency only Set sigma factor	Corrected data
Part	Use part of data	Set start/end points End=0=last point	Selected data
Average	Combine data for longer $\tau$	Set averaging factor	Averaged data
Normalize	Remove median from data		
Remove Drift	Remove fit	Choose fit type	Residual data
Run	Stability analysis	Choose variance type	Stability plot Stability table
Close	Close last screen	Leave open for manual analysis	Add delay to observe steps



- **Controls**

The Auto# dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
Open <u>F</u> ile	Checkbox	Open data file for analysis
Browse	Pushbutton	Browse for data file.
Data Type	Text	Label
<u>P</u> hase	Radiobutton	Choose phase data.
Freq	Radiobutton	Choose frequency data.
File <u>n</u> ame	Edit	Enter name of data file.
Con <u>v</u> ert	Checkbox	Convert to other data type
Regularize( <u>.)</u>	Checkbox	Use timetags to insert gaps.
Remove <u>O</u> utliers	Checkbox	Remove bad freq data.
Sigma <u>F</u> actor	Edit	Enter outlier criterion.
Part( <u>0</u> )	Checkbox	Use part of data.
<u>S</u> tart	Edit	Enter starting point (1=1 <sup>st</sup> ).
<u>E</u> nd	Edit	Enter ending point (0=last).
Average( <u>1</u> )	Checkbox	Combine data for longer $\tau$ .
Avg Factor	Edit	Enter averaging factor.
Normal <u>i</u> ze	Checkbox	Normalize data.
Remove Drift( <u>2</u> )	Checkbox	Remove drift.
Drift Type( <u>3</u> )	Listbox	Select drift fit type.
Data <u>P</u> lot	Checkbox	Plot data on screen.
Type( <u>4</u> )	Listbox	Select data plot type(s).
Print Data Plot( <u>5</u> )	Checkbox	Print data plot.
Run( <u>6</u> )	Checkbox	Run stability analysis.
Variance	Combobox	Select variance type.
Stability <u>P</u> lot	Checkbox	Plot stability on screen.
Print Stab Plot( <u>7</u> )	Checkbox	Print stability plot.
Print Table( <u>8</u> )	Checkbox	Print stability table.
Close( <u>X</u> )	Checkbox	Close at end of script.
Add Delay( <u>9</u> )	Checkbox	Add delay between steps.
msec	Edit	Enter delay in msec.
Script Name	Edit	Enter name of auto script.
<u>G</u> o	Pushbutton	Execute automation script.
<u>O</u> K	Pushbutton	Save automation script.
<u>C</u> ancel	Pushbutton	Abort Auto# function.
<u>H</u> elp	Pushbutton	Invoke Auto# help topic.

- **Manual Use**

The Auto# functions may be used manually by setting the desired analysis steps and either saving them by pressing **OK**, or executing them by pressing **Go**. Pressing **Cancel** aborts any changes made. The analysis functions are the same as provided for a non-automated Stable32 analysis. The automation script options are interlocked so as to simplify their selection. All reasonable combinations are allowed, but their order may not be changed. Default timetag and data columns are used for multicolumn data. The **Close** checkbox determines whether the last result remains on the screen. Extra delay may be added with the **Add Delay** and **msec** controls to allow viewing of the intermediate results.

- **Command Line Use**

The Auto# functions may be invoked from the command line with the `-o auto1`, `-o auto2` or `-o auto3` options. When invoked in this way, the automation script is executed immediately without requiring that it be started with the **Go** button.

- **Automation Scripts**

Three separate automation scripts are stored in the Stable32.ini configuration file under the [Auto#] sections. Each of these sections has the same items that define the analysis steps for each automated analysis. While it is possible to edit these configuration file items directly, it is recommended that the corresponding Auto# dialog box be used to avoid errors.

- **Analysis Methods**

The Stable32 automation script feature is intended as a convenient way to quickly produce basic stability analysis information. It is not intended to replace either manual analysis or complete batch files. The former has the advantage of close user integration and insight, while the latter can harness the full capabilities of the program (including plot customization and annotation).

- **Caution**

While considerable effort has been atmen to make this automation process robust and useful, many factors (including data size, hardware speed and concurrent programs) can affect the script execution, and successful operation cannot be guaranteed.

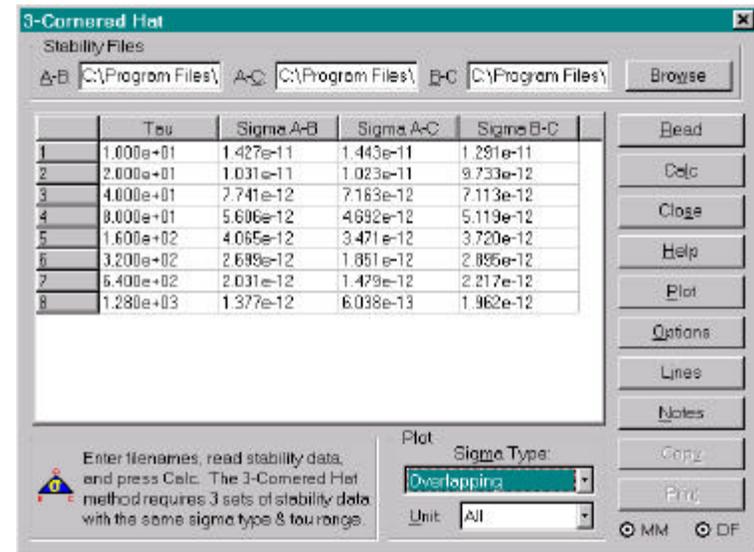
## 3-Cornered Hat Function

- **Purpose**

Use the 3-Cornered Hat function to (with discretion) to separate variances from 3 sets of stability data.

- **Controls**

The 3-Cornered Hat dialog box contains the following controls:



<b>Control</b>	<b>Type</b>	<b>Description</b>
Stability Files	Group	Choose stability filenames.
A-B	Edit	Enter A-B stability data filename.
A-C	Edit	Enter A-C stability data filename.
B-C	Edit	Enter B-C stability data filename.
Browse	Pushbutton	Invoke file open dialog box to select data file.
Tau, etc.	Table	Table of tau and stability data.
Read	Pushbutton	Read stability data.
Calc	Pushbutton	Perform calculation to separate variances.
Close	Pushbutton	Close dialog box.
Help	Pushbutton	Invoke this help topic.
Plot	Pushbutton	Plot stability data.
Options	Pushbutton	Open Options dialog box.
Lines	Pushbutton	Open Lines dialog box.
Notes	Pushbutton	Open Notes dialog box.
Copy	Pushbutton	Copy stability results to Windows Clipboard.
Print	Pushbutton	Print stability results.
MM	Indicator	Indicates if min & max sigma data read
DF	Indicator	Indicates if degrees of freedom data read

- **Operation**

Enter stability data filenames using A-B, A-C, and B-C edit boxes or Browse. These stability files must contain 3 sets of data with the same sigma type and tau range, in the standard Stable32 stability data file format, as described for the Read function. Only a placeholder value (e.g. 1) is needed for the # of analysis points, and no entries are needed for the minimum and maximum sigma columns. Then press Read to read the 3 sets of stability data, which will be shown in the table, and can be examined with Plot, Options, Lines and Notes buttons. Then press Calc to separate the variances, which can also be examined with Plot, Options, Lines and Notes features, copied to the clipboard with the Copy button, or printed with the Print button. When done, end the 3-Cornered Hat function with the Close button.

Error bars can be shown on the stability plots for a single unit. They are determined by the maximum and minimum sigmas of the measured stability, and the reduced number of degrees of freedom for the processed data. The former requires maximum and minimum sigma data, while the latter requires that the number of measured degrees of freedom be available. The availability of these data are indicated by their respective MM and DF indicators.

- **Reference Noise Removal**

The 3-Cornered Hat function can also be used to remove the contribution of a reference whose noise is known apriori, either from its specification or by a separate determination. Denote the device under test as unit A, and the (known) reference as units B and C. Enter the measured data as both A-B and A-C, multiply the stability data for the reference by  $\bar{O}_2$ , and enter it as B-C. After the 3-cornered hat calculation, the corrected stability data for the unit under test will be in column A.

- **Results Data Files**

Three data files containing the results of a 3-cornered hat calculation are written to disk in the same folder as the original data. These files, named UnitA.sig, UnitB.sig and UnitC.sig, have the same stability file format as the input file (see Read Function).

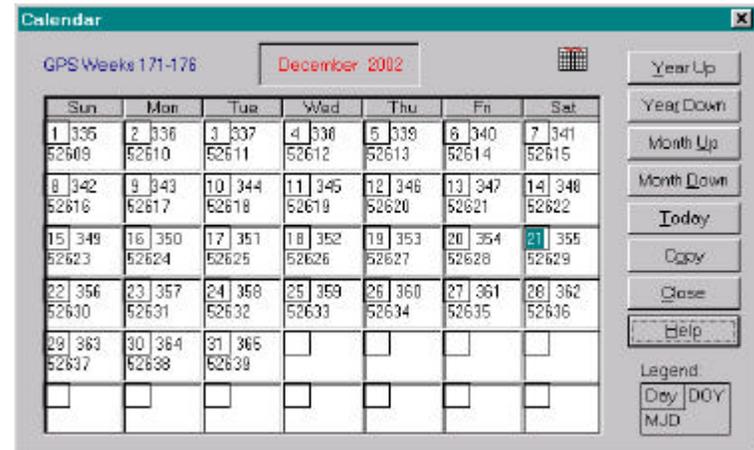
- **Caution**

Do not use results that contain negative variances. The 3-cornered hat method is not a substitute for a low noise reference. It is best used for units having similar stability (e.g. to determine which unit is best). Negative variances are a sign that the method is failing (because it was based on insufficient measurement data, or because the units under test have disparate stability). Plots containing negative variances will have off-scale points.

## Calendar Function

- **Purpose**

Use the Calendar function to display a monthly calendar showing the Day of Year (DOY) and Modified Julian Date (MJD). The MJD can be used as a timetag for phase or frequency data. It is common to append the decimal fraction of a day to the MJD integer to indicate the exact date (date and time). When this is done, the time is usually UTC.



- **Controls**

The Calendar dialog box contains the following controls:

<u>Control</u>	<u>Type</u>	<u>Description</u>
<u>Y</u> ear Up	Pushbutton	Advance the calendar by 1 year.
<u>Y</u> ear Down	Pushbutton	Retard the calendar by 1 year.
<u>M</u> onth <u>U</u> p	Pushbutton	Advance the calendar by 1 month.
<u>M</u> onth <u>D</u> own	Pushbutton	Retard the calendar by 1 month.
<u>T</u> oday	Pushbutton	Go to current month & highlight current day.
<u>C</u> lose	Pushbutton	Close the Calendar dialog box.
<u>H</u> elp	Pushbutton	Invoke the Calendar help topic.

- **Operation**

The calendar opens to the current month, with the current day highlighted. Use the **Year Up**, **Year Down**, **Month Up** and **Month Down** auto repeating buttons to move to the month desired. Use the **Today** button to return to the current month. When finished, **Close** the calendar dialog box.

- **MJD**

The Modified Julian Date (MJD) is based on the astronomical Julian Date, the # of days since noon on January 1, 4713 BC. The MJD is the Julian Date - 2,400,000.5; it starts at midnight on November 17, 1858. MJD 50,000 was on October 10, 1995.

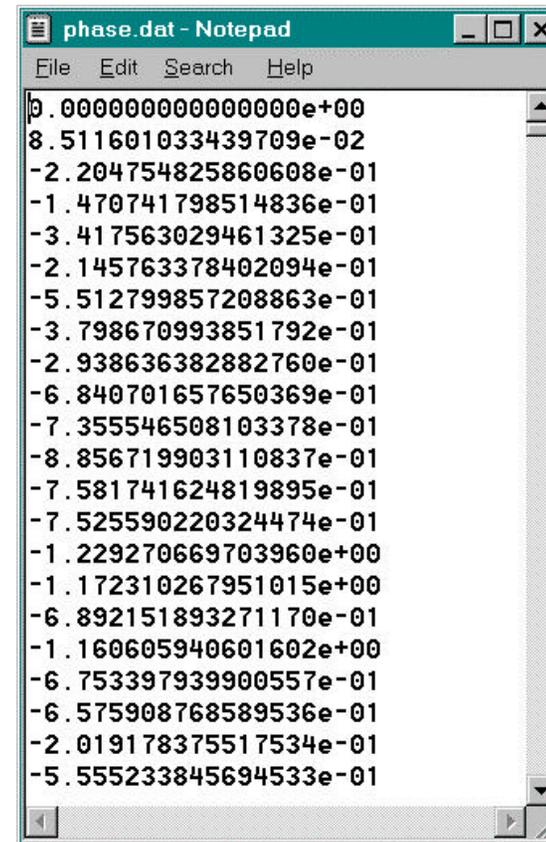
- **GPS Week**

The GPS week number is displayed at the top left of the calendar (when applicable). This number is used by the Global Positioning System to measure its system time. It has an origin (time zero) of 00:00:00 UTC 6 January 1980, Julian Day 2,444,244.500, MJD 44244. A GPS week cycle is 1024 weeks.

## Pad

- **Purpose**

Use the Pad function to invoke Windows Notepad or an alternative text editor program.



- **Operation**

The current data file is automatically opened when the Pad function is invoked. Consult the Notepad help topics for information regarding that program. Use the Notepad= item in the [Preferences] section of the STABLE32.INI configuration file to set the name of the text editor program to be invoked.

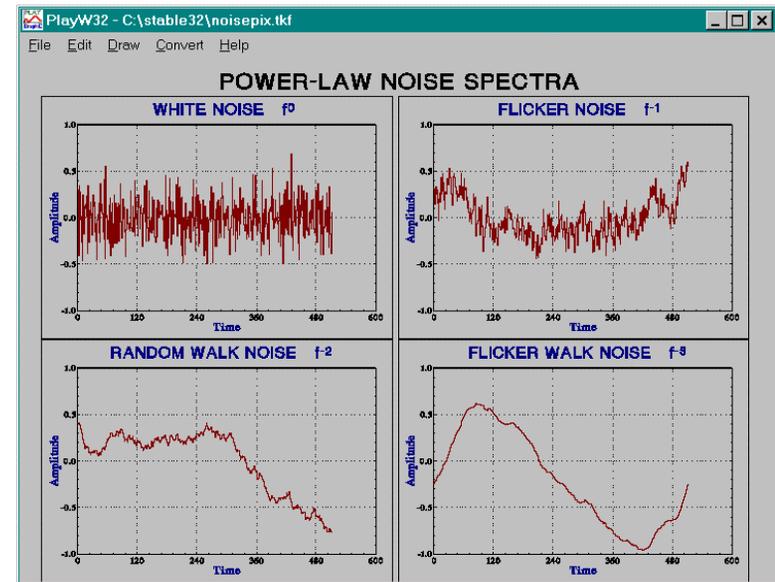
- **Caution**

Opening of the current data file may be prohibited as a file sharing violation.

## Play

- **Purpose**

Use the Play function to invoke the Play/Win program to redisplay a previous GraphiC \*.TKF plot.



- **Operation**

Consult the Play/Win help topics for information regarding that program.

- **Note**

The Play program is not included with all versions of the Stable32 software package.

## Glossary

- **Glossary**

The following terms are used in the field of frequency stability analysis and the Stable32 program

<b>Aging</b>	The change in frequency with time due to internal effects within the device.
<b>Allan Variance</b>	The 2-sample variance $\sigma_y^2(\tau)$ commonly used to measure frequency stability.
<b>Averaging</b>	The process of combining phase or frequency data into samples at a longer averaging time.
<b>Averaging Time</b>	See Tau.
<b>BW</b>	Bandwidth, Hz.
<b>Confidence Limit</b>	The uncertainty associated with a measurement. Often a 68% confidence level or error bar.
<b>Drift</b>	The change in frequency with time due to all effects (including aging and environmental sensitivity).
<b>Dynamic Stability</b>	A form of the Allan variance showing the stability versus time.
<b>Frequency Data</b>	A set of fractional frequency values, $y[i]$ , where $i$ denotes equally-spaced time samples.
<b>Hadamard Variance</b>	A 3-sample variance, HVAR, that is similar to the 2-sample Allan variance. It uses the 2nd differences of the fractional frequencies, and is unaffected by linear frequency drift.
<b><math>\mathcal{L}(f)</math></b>	The ratio of the SSB phase noise power in a 1 Hz BW to the total carrier power, dBc/Hz.
<b>MJD</b>	The Modified Julian Date is based on the astronomical Julian Date, the # of days since noon on January 1, 4713 BC. The MJD is the Julian Date - 2,400,000.5.
<b>Modified Sigma</b>	A modified version of the Allan or total variance that uses phase averaging to distinguish between white and flicker PM noise processes.
<b>MTIE</b>	The maximum time interval error of a clock.

<b>Normalize</b>	To remove the average value from phase or frequency data .
<b>Phase Data</b>	A set of time deviates, $x[i]$ with units of seconds, where $i$ denotes equally-spaced time samples. Called "phase" data to distinguish them from the independent time variable.
<b>Phase Noise</b>	The spectral density of the phase deviations.
<b>Sampling Time</b>	See Tau.
<b>Sigma</b>	The square root or deviation of a variance, often the 2-sample or Allan deviation, $\sigma_y(\tau)$ .
<b>Slope</b>	The change in frequency per tau interval.
<b>SSB</b>	Single sideband.
<b><math>S_i(f)</math></b>	The one-sided spectral density of the phase deviations, $\text{rad}^2/\text{Hz}$ .
<b><math>S_x(f)</math></b>	The one-sided spectral density of the time deviations, $\text{sec}^2/\text{Hz}$ .
<b><math>S_y(f)</math></b>	The one-sided spectral density of the fractional frequency deviations, $1/\text{Hz}$ .
<b>Tau</b>	The interval between phase measurements or the averaging time used for a frequency measurement.
<b>Total</b>	An analysis technique using an extended data set that provides better confidence at long averaging times.
<b>Thêo1</b>	A statistic for analyzing stability at large averaging factors extending out to 75% of the record length.
<b>ThêoBR</b>	A bias-removed version of Thêo1.
<b>ThêoH</b>	A hybrid version of Thêo1, combining ThêoBR with the Allan variance.
<b>TIE</b>	The time interval error of a clock. Can be expressed as the rms time interval error TIE rms or the maximum time interval error MTIE.

<b>Total Variance</b>	A 2-sample variance similar to the Allan variance with improved confidence at large averaging factors.
<b><math>x(t)</math></b>	The instantaneous time deviation from a nominal time, $x(t) = \phi(t)/2\pi\nu_0$ , seconds, where $\nu_0$ is the nominal frequency, Hz. This dependent time variable is often called "phase" to distinguish it from the independent time variable $t$ .
<b><math>y(t)</math></b>	The instantaneous fractional frequency deviation from a nominal frequency, $y(t) = [\nu(t)-\nu_0]/\nu_0 = x'(t)$ , where $\nu_0$ is the nominal frequency.

## References

### • References

The following references are recommended to support the use of the Stable32 program for the analysis of frequency stability:

1. "Characterization of Clocks and Oscillators", *NIST Technical Note 1337*, D. B. Sullivan, D.W. Allan, D.A. Howe and F. L. Walls, Eds., U.S. Dept. of Commerce, National Institute of Standards and Technology, March 1990.
2. S.R. Stein, "Frequency and Time-Their Measurement and Characterization", Chap. 12, *Precision Frequency Control*, E.A. Gerber and A. Ballato, Eds., Academic Press, Inc., Orlando, 1985, ISBN 0-12-280602-6 (Vol. 2).
3. D.W. Allan, "Statistics of Atomic Frequency Standards", *Proc. IEEE*, Vol. 54, pp. 221-230, February 1966.
4. J.A. Barnes, et al, "Characterization of Frequency Stability", *IEEE Trans. Instrum. Meas.*, Vol. IM-20, No. 2, pp. 105-120, May 1971.
5. D.A. Howe, D.W. Allan and J.A. Barnes, "Properties of Signal Sources and Measurement Methods", *Proc. 35th Annu. Freq. Control Symp.*, pp. A1-A47, June 1981.
6. Report 580, "Characterization of Frequency and Phase Noise", International Radio Consultative Committee (C.C.I.R.), pp. 142-150, 1986.
7. J.A. Barnes, "The Analysis of Frequency and Time Data", Austron, Inc., December 1991.
8. D.W. Allan, et al., "Standard Terminology for Fundamental Frequency and Time Metrology", *Proc. 42nd Annu. Freq. Control Symp.*, pp. 419-425, June 1988.
9. W.J. Riley, "A Test Suite for the Calculation of Time Domain Frequency Stability", *Proc. 1995 IEEE Freq. Contrl. Symp.*, pp. 360-366, June 1995.
10. W.J. Riley, "Addendum to a Test Suite for the Calculation of Time Domain Frequency Stability", *Proc. 1996 IEEE Freq. Contrl. Symp.*, pp. 880-881, June 1995.
11. "IEEE Standard Definitions of Physical Quantities for Fundamental Frequency and Time Metrology," *IEEE Std. 1139-1988*.
12. J.A. Barnes, "The Measurement of Linear Frequency Drift in Oscillators", *Proc. 15th Annu. PTTI Meeting*, pp. 551-582, December 1983.
13. M.A. Weiss and C. Hackman, "Confidence on the Three-Point Estimator of Frequency Drift", *Proc. 24th Annu. PTTI Meeting*, pp. 451-460, December 1992.
14. MIL-O-55310B, Oscillators, Crystal, General Specification For.
15. D.W. Allan and J.A. Barnes, "A Modified Allan Variance with Increased Oscillator Characterization Ability", *Proc. 35th Annu. Symp. on Freq. Contrl.*, pp. 470-474, May 1981.

16. D.W. Allan, D.D. Davis, J. Levine, M.A. Weiss, N. Hironaka, and D. Okayama, "New Inexpensive Frequency Calibration Service From NIST", *Proc. 44th Annu. Symp. on Freq. Contrl.*, pp. 107-116, June 1990.
17. D.W. Allan, M.A. Weiss, and J.L. Jespersen, "A Frequency-Domain View of Time-Domain Characterization of Clocks and Time and Frequency Distribution Systems", *Proc. 45th Annu. Symp. on Freq. Contrl.*, pp. 667-678, May 1991.
18. D.W. Allan, "Time and Frequency Metrology: Current Status and Future Considerations", *Proc. 5th European Freq. and Time Forum*, pp. 1-9, March 1991.
19. C.A. Greenhall, "A Shortcut for Computing the Modified Allan Variance", *Proc. 1992 IEEE Freq. Contrl. Symp.*, pp. 262-264, May 1992.
20. B.E. Blair (Editor), *Time and Frequency: Theory and Fundamentals*, NBS Monograph 140, Annex 8.E, p. 181, May 1974.
21. S.K. Park and K.W. Miller, "Random Number Generators: Good Ones are Hard to Find", *Comm. ACM*, Vol. 31, No. 10, pp. 1192-1201, October 1988.
22. W.H. Press, B.P. Flannery, S.A. Teukolsky and W.T. Vetterling, *Numerical Recipes in C*, Cambridge Univ. Press, Cambridge, U.K., 1988, pp. 216-217.
23. D.B. Percival, "Use of Robust Statistical Techniques in Time Scale Formation", *Preliminary Report*, U.S. Naval Observatory Contract No. N70092-82-M-0579, 1982.
24. Gernot M.R. Winkler, "Introduction to Robust Statistics and Data Filtering", *Tutorial at 1993 IEEE Freq. Contrl. Symp.*, Sessions 3D and 4D, June 1, 1993.
25. C.A. Greenhall, "Recipes for Degrees of Freedom of Frequency Stability Estimators", *IEEE Trans. Instrum. Meas.*, Vol. 40, No. 6, pp. 994-999, December 1991.
26. C.A. Greenhall, "Estimating the Modified Allan Variance", *Proc. IEEE 1995 Freq. Contrl. Symp.*, pp. 346-353, May 1995.
27. Kasdin and T. Walter, "Discrete Simulation of Power Law Noise", *Proc. 1992 IEEE Freq. Contrl. Symp.*, pp. 274-283, May 1992.
28. Howe, "Methods of Improving the Estimation of Long-Term Frequency Variance", *Proc. 11th European Freq. and Time Forum*, pp. 91-99, March 1997.
29. S. Bregni, "Clock Stability Characterization and Measurement in Telecommunications", *IEEE Trans. Instrum. Meas.*, Vol. 46, No. 6, pp. 1284-1294, December 1997.
30. D. Howe, R. Beard, C. Greenhall, F. Vernotte and W. Riley, "A Total Estimator of the Hadamard Function Used for GPS Operations", *Proceedings of the 32nd Annual Precise Time and Time Interval (PTTI) Systems and Applications Meeting*, pp. 255-268, November 2000.
31. C. Greenhall and W. Riley, "Uncertainty of Stability Variances Based on Finite Differences", *Proc. 2003 PTTI Meeting*, December 2003.
32. D.A. Howe and T.K. Pepler, "Very Long-Term Frequency Stability: Estimation Using a Special-Purpose Statistic", *Proceedings of the 2003 IEEE International Frequency Control Symposium*, pp. 233-238, May 2003.
33. W.J. Riley and C.A. Greenhall, "Power Law Noise Identification Using the Lag 1 Autocorrelation", *Proc. 18th European Frequency and Time Forum (EFTF)*, April 2004, Session 6B.
34. D.A. Howe, R.L. Beard, C.A. Greenhall, F. Vernotte, W.J. Riley and T.K. Pepler, "Enhancements to GPS Operations and Clock Evaluations Using a "Total" Hadamard Deviation", *IEEE Trans. UFFC*, Vol. 52, No. 8, August 2005.

## Example

- **Example**

1. Double click on the Stable32 program group.
2. Start the Stable32 program by double clicking on its icon.
3. Go to the file open function by clicking on the Open toolbutton.
4. Select the frequency data type by clicking on the Frequency Radiobutton.
5. Open the SAMPLE.DAT file by selecting it and pressing OK.
6. Observe the plot and statistics for this data file in the Data File Opened dialog box.
7. Close the Data File Opened dialog box by pressing OK.
8. Notice the frequency data file information shown in the status bar at the bottom right of the screen.
9. Display the frequency data by clicking on the Edit toolbutton.
10. Use the scrollbars to examine the frequency data. The Edit function can display timetags, gaps and outliers and can perform several search and replace functions, which you can explore later. For now, exit the Edit function by pressing Cancel.
11. Select the Plot function by clicking on its toolbutton.
12. Press the Plot button to produce a GraphiC plot of the frequency data. The plot shows white frequency noise plus drift. Close the plot by double clicking on its system menu icon. If you wish, you can try pressing the plot Options button to change the plot title or add a linear regression line to the plot. Then leave the overall plot function.
13. Select the Drift function from the Analysis menu or toolbar. Notice that you have the option to Show Details for this function.
14. Choose Linear drift and check the Remove Drift box. Then press Calc to calculate and remove the linear drift of about 1.0 per tau interval from the frequency data. Notice that the unsaved symbol has appeared in the status bar indicating that the data has changed. Also, if enabled in the Configure box, the filename extension will have been changed to .001.

15. Close the Drift dialog box by pressing Close.
16. Select the Statistics function. Notice that the plot now shows only noise. Close the Statistics box.
17. Choose the Scale function. Change the data so it more closely resembles actual stability data by entering a Multiplier of 1.0e-11 and pressing OK. If you wish, examine the data again with the Edit function to see how it has changed.
18. Go to the Sigma function. Accept the default Normal sigma type and Averaging Factor=1. Then press Calc to calculate the sigma value (about 2.99e-10). Try the Show Details option and some other sigma types and averaging factors for this sample of white FM noise.
19. Go to the Run function. Select the desired Variance Type and press Calc to produce a table of stability data. Then press Plot to see it plotted, or Print to send it to your printer. Try the Options, Lines and Notes functions if you wish to add a noise line or otherwise customize the stability plot.
20. Go to the Convert function and convert the frequency data into the corresponding phase data. Notice that the status bar now shows a 513-point SAMPLE.004 phase data file.
21. Repeat some of the analysis and plotting functions with the phase data.
22. Try other Stable32 functions as desired. Generate and analyze some simulated clock noise. Try other plot options. Try the domain conversions, or generate and use timetags. Use the context sensitive Help button as needed.

## Test Suite

- **Test Suite**

The following tables summarize the values for several common frequency stability measures for both the classic NBS data set and a 1000-point portable test suite.

- **NBS Data Set**

A "classic" suite of frequency stability test data is the set of nine 3-digit numbers from Annex 8.E of NBS Monograph 140 shown in Table I. Those numbers were used as an early example of an Allan variance calculation. This frequency data is also normalized to zero mean by subtracting the average value, and then integrated to obtain phase values. A listing of the properties of this data set is shown in Table II. While nine data points are not sufficient to calculate large frequency averages, they are, nevertheless, a very useful starting point to verify frequency stability calculations since a small data set can easily be entered and analyzed manually. A small data set is also an advantage for detecting "off-by-one" errors.

**Table I. NBS Monograph 140, Annex 8.E Test Data**

#	<u>Freq-</u> <u>uency</u>	<u>Normalized</u> <u>Frequency</u>	<u>Phase</u> <u>(t=1)</u>
1	892	103.11111	0.00000
2	809	20.11111	103.11111
3	823	34.11111	123.22222
4	798	9.11111	157.33333
5	671	-117.88889	166.44444
6	644	-144.88889	48.55555
7	883	94.11111	-96.33333
8	903	114.11111	-2.22222
9	677	-111.88889	111.88889
10			0.0000

**Table II. NBS Monograph 140, Annex 8.E Test Data Statistics**

<u>Averaging Factor</u>	<u>1</u>	<u>2</u>
# Data Points	9	4
Maximum	903	893.0
Minimum	644	657.5
Average	788.8889	802.875
Median	809	830.5
Linear Slope	-10.20000	-2.55
Intercept	839.8889	809.25
Standard Deviation [1]	100.9770	102.6039
Normal Allan Deviation	91.22945	115.8082
Overlapping Allan Dev	91.22945	85.95287
Modified Allan Dev	91.22945	74.78849
Time Deviation	52.67135	86.35831
Hadamard Deviation	70.80607	116.7980
Overlap Hadamard Dev	70.80607	85.61487
Hadamard Total Dev	70.80607	91.16396
Total Deviation	91.22945	93.90379
Total modified Dev	75.50203	75.83606
Time Total Deviation	43.59112	87.56794

Table II Notes:

[1] Sample (not population) standard deviation.

• **1000-Point Test Suite**

The larger frequency data test suite used here consists of 1000 pseudo-random frequency data points. It is produced by the following prime modulus linear congruential random number generator:

$$n_{i+1} = 16807 n_i \text{ Mod } 2147483647$$

This expression produces a series of pseudo-random integers ranging in value from 0 to 2147483646 (the prime modulus,  $2^{31}-1$ , avoids a collapse to zero). When started with the seed  $n_0 = 1234567890$ , it produces the sequence  $n_1 = 395529916$ ,  $n_2 = 1209410747$ ,  $n_3 = 633705974$ , etc. These numbers may be divided by 2147483647 to obtain a set of normalized floating-point test data ranging from 0 to 1. Thus the normalized value of  $n_1$  is 0.5748904732. A spreadsheet program is a convenient and reasonably universal way to generate this data. The frequency data set may be converted to phase data by assuming an averaging time of 1, yielding a set of 1001 phase data points. Similarly, frequency offset and/or drift terms may be added to the data. These conversions can also be done by a spreadsheet program.

The values of this data set will be uniformly distributed between 0 and 1. While a data set with a normal (Gaussian) distribution would be more realistic, and could be produced by summing a number of independent uniformly distributed data sets, or by the Box-Muller method [24], this simpler data set is adequate for software validation.

**Table III. 1000-Point Frequency Data Set**

<u>Averaging Factor</u>	<u>1</u>	<u>10</u>	<u>100</u>
# Data Points	1000	100	10
Maximum	9.957453e-01	7.003371e-01	5.489368e-01
Minimum	1.371760e-03	2.545924e-01	4.533354e-01
Average [1]	4.897745e-01	4.897745e-01	4.897745e-01
Median	4.798849e-01	5.047888e-01	4.807261e-01
Linear Slope [2,3]	6.490910e-06	5.979804e-05	1.056376e-03
Intercept [3]	4.865258e-01	4.867547e-01	4.839644e-01
Bisection Slope [2]	-6.104214e-06	-6.104214e-05	-6.104214e-04
1st Diff Slope [2]	1.517561e-04	9.648320e-04	1.011791e-03
Log Fit [2,4], a=	5.577220e-03	5.248477e-03	7.138988e-03
y(t)=a·ln(bt+1)+c, b=	9.737500e-01	4.594973e+00	1.420429e+02
y'(t)=ab/(bt+1), c=	4.570469e-01	4.631172e-01	4.442759e-01
Slope at end	5.571498e-06	5.237080e-05	7.133666e-04
Standard Dev [5]	2.884664e-01	9.296352e-02	3.206656e-02
Normal Allan Dev [6]	2.922319e-01	9.965736e-02	3.897804e-02
Overlap Allan Dev [8]	2.922319e-01	9.159953e-02	3.241343e-02
Mod Allan Dev [7,8]	2.922319e-01	6.172376e-02	2.170921e-02
Time Deviation [8]	1.687202e-01	3.563623e-01	1.253382e-00
Hadamard Deviation	2.943883e-01	1.052754e-01	3.910861e-02
Overlap Had Dev	2.943883e-01	9.581083e-02	3.237638e-02
Hadamard Total Dev	2.943883e-01	9.614787e-02	3.058103e-02
Total Deviation	2.922319e-01	9.134743e-02	3.406530e-02
Total modified Dev	2.418528e-01	6.499161e-02	2.287774e-02
Time Total Deviation	1.396338e-01	3.752293e-01	1.320847e-00

Table III Notes:

[1] Expected value = 0.5.

[2] All slopes are per interval.

[3] Least squares linear fit.

[4] Exact results will depend on iterative algorithm used. Data not suited to log fit.

[5] Sample (not population) standard deviation. Expected value =  $1/\sqrt{12} = 0.2886751$ .

[6] Expected value equal to standard deviation for white FM noise.

[7] Equal to normal Allan deviation for averaging factor = 1.

[8] Calculated with listed averaging factors from the basic  $\tau = 1$  data set.

**Table IV. Error Bars for n=1000 Point =1 Data Set with Avg Factor=10**

<u>Allan Dev</u> <u>Type/#Pts</u>	<u>Sigma Value</u> <u>Noise Type &amp; Ratio</u>	<u>Confidence Interval</u> <u>Remarks</u>	<u>c<sup>2</sup> for</u> <u>95%</u>
Normal CF	9.965736e-02	CI= 8.713870e-03 [2,3]	
99	W FM [1], B1=0.870	±1σ CI. [6]	
Overlapping 981	9.159953e-02 W FM # $\chi^2$ df=146.177	Max $\sigma_y(\tau)$ =1.014923e-01 [7] Max $\sigma_y(\tau)$ = 1.035201e-01[8] Min $\sigma_y(\tau)$ = 8.223942e-02 [8]	119.07 114.45 181.34
Modified [4] 972	6.172376e-02 W FM[5], R(n)=0.384 # $\chi^2$ df=94.620	Max $\sigma_y(\tau)$ =7.044412e-02 [7] Max $\sigma_y(\tau)$ = 7.224944e-02 [8] Min $\sigma_y(\tau)$ = 5.419961e-02 [8]	72.64 69.06 122.71

Table IV Notes:

[1] Theoretical B1=1.000 for W FM noise and 0.667 for F and W PM noise.

[2] Simple, noise-independent CI estimate  $=\sigma_y(\tau)/\sqrt{N}=1.001594e-02$ .

[3] This CI includes  $\kappa(\alpha)$  factor that depends on noise type:

<u>Noise</u>	<u>a</u>	<u>k(a)</u>
W PM	2	0.99
F PM	1	0.99
W FM	0	0.87
F FM	-1	0.77
RW FM	-2	0.75

[4] BW factor  $2\pi f_i \tau_0 = 10$ . Applies only to F PM noise.

[5] Theoretical R(n) for W FM noise = 0.500 and 0.262 for F PM noise.

[6] Double-sided 68.% confidence interval: p = 0.158 and 0.842.

[7] Single-sided 95% confidence interval: p = 0.950.

[8] Double-sided 95% confidence interval: p = 0.025 and 0.975.

## Analysis Procedure

### • Analysis Procedure

The Stable32 functions can be used freely as the circumstances dictate. Nevertheless, the following list shows a typical analysis procedure.

1. Open a phase data file from disk.
2. Examine the phase data in the Data File Opened box and enter the averaging time (tau) value.
3. Convert the phase data to frequency data.
4. Check the frequency data for outliers.
5. Plot the frequency data.
6. Remove the frequency drift.
7. Plot the frequency residuals.
8. Run a stability analysis.
9. Plot the stability data.

The results of an analysis can be documented by printing a table, plotting a graph, copying values to the Windows clipboard, reading a stability file, replaying a plot, or reading or pasting a converted plot into another application. In addition, a log file is produced that documents all the analysis steps performed.

### • Data Precision

There are relatively few numerical precision issues relating to the analysis of frequency stability data. One exception, however, is phase data for a highly stable frequency source having a relatively large frequency offset. The raw phase data will be essentially a straight line (representing the frequency offset), and the instability information is contained in the small deviations from the line. A large number of digits must be used unless the frequency offset is removed by subtracting a linear term from the raw phase data. Similar considerations apply to the quadratic phase term (linear frequency drift). Many frequency stability measures involve averages of first or second differences. Thus, while their numerical precision obviously depends upon the variable digits of the data set, there is little error propagation in forming the summary statistics.

### • Data Preprocessing

Preprocessing of the measurement data is often necessary before performing the actual analysis, which may require data averaging, or removal of outliers, frequency offset and drift.

Phase data may be converted to frequency data and vice versa. Phase and frequency data can be combined for a longer averaging time. Frequency offset may be removed from phase data by subtracting a line determined by the average of the first differences, or by a least squares linear fit. An offset may be removed from frequency data by normalizing it to have an average value of zero. Frequency drift may be removed from phase data by a least-squares or 3-point quadratic fit, or by subtracting the average of the second

differences. Frequency drift may be removed from frequency data by subtracting a least-squares linear fit, by subtracting a line determined by the first-differences or by calculating the drift from the difference between the two halves of the data. The latter, called the bisection drift, is equivalent to the 3-point fit for phase data. Other more specialized log and diffusion models may also be used. The latter are particularly useful to describe the stabilization of a frequency source. In general, the objective is to remove as much of the deterministic behavior as possible, obtaining random residuals for subsequent noise analysis.

- **Gaps, Jumps and Outliers**

It is not uncommon to have gaps and outliers in a set of raw frequency stability data. Missing or erroneous data may occur due to power outages, equipment malfunctions and interference. For long-term tests, it may not be possible or practical to repeat the run, or otherwise avoid such bad data points. Usually the reason for the gap or outlier is known, and it is particularly important to explain all phase discontinuities. Plotting the data will often show the bad points, which may have to be removed before doing an analysis to obtain meaningful results.

Frequency outliers are found by comparing each data point with the median value of the data set plus or minus some multiple of the absolute median deviation. These median statistics are more robust because they are insensitive to the size of the outliers. The outliers can be replaced by gaps or filled with interpolated values.

Frequency jumps can also be a problem for stability analysis. They are an indication that the statistics are not stationary, and it may be necessary to divide the data into portions and analyze them separately.

- **Data Plotting**

Data plotting is often the most important step in the analysis of frequency stability. Visual inspection can provide vital insight into the results, and is an important preprocessor before numerical analysis. A plot also shows much about the validity of a curve fit.

Phase data is plotted as line segments connecting the data points. This presentation properly conveys the integral nature of the phase data. Frequency data is plotted as a horizontal line between the frequency data points. This shows the averaging time associated with the frequency measurement, and mimics the analog record from a frequency counter. As the density of the data points increases, there is essentially no difference between the two plotting methods. Missing data points are shown as gaps without lines connecting the adjacent points.

- **Variance Selection**

It is the user's responsibility to select an appropriate variance for the stability analysis. The overlapping Allan variance is recommended in most cases, especially where the frequency drift is small or has been removed. The Allan variance is the most widely used time-domain stability measure and the overlapping form provides better confidence than the original non-overlapping version. The total variances and  $\text{Th}\sigma_1$  can be used for even better confidence at large averaging factors (at the expense of longer computation time). The modified Allan variance is recommended to distinguish between white and flicker PM noise, and, again, a total form of it is available for better confidence at long tau. The time variance provides a good measure of the time dispersion of a clock due to noise, while MTIE measures the peak time deviations. TIE rms can also be used to assess clock performance, but TVAR is generally preferred. Finally, the overlapping Hadamard variance is recommended over its non-overlapping form for analyzing stability in the presence of divergent noise or frequency drift. In all cases, the results are reported in terms of the deviations.

The choice of tau interval depends mainly on whether interference mechanisms are suspected which cause the stability to vary periodically. Normally, octave spacing will be used, with decade spacing available to analyze tau multiples of ten. The all tau option can be useful as a form of spectral analysis to detect cyclic disturbances (such as temperature cycling).

## Validation

- **Validation**

Considerable effort has been taken to insure that the results obtained from Stable32 are correct. In particular, each of the six validation methods listed in Reference 9 has been employed. Stable32 uses the same basic analysis functions of Stable/DOS and the FrequenC Library, both of which have been used extensively by a number of individuals and organizations for over ten years. In addition, feedback and peer review has been (and still is) encouraged. Please report any suspected discrepancies. A convenient way to do this is via the Feedback item under the Help menu.

One important validation method is comparison of the program results against the test suite described above. Copies of the NBS and 1000-point test data are included with the Stable32 distribution.

## Acknowledgments

### • Credits

Stable32 was written by W.J. Riley in C using the Win32 API and the Microsoft VisualC++ 5.0 compiler. Two major 3rd party software components were used, GraphiC /Win 7.1 from Scientific Endeavors Corporation for the plotting functions and DataTable 3.0 from ProtoView Development Corporation for the data tables. Most of the stability analysis functions use the FrequenC Library from Hamilton Technical Services. NuMega BoundsChecker 5.03 Visual C++ Edition was used for debugging. The help file was prepared with Help Writer's Assistant from Olson Software Limited, and the installation program is InstallShield Express2 from InstallShield Software Corporation. Extensive use was made of the mathematical algorithms in *Numerical Recipes in C*. Many of these products, as well as Windows from Microsoft Corporation, are covered by copyrights and trademarks. The Database function uses PostgreSQL, an open-source client-server relational database program. Its software, tools and documentation are of exceptional quality, and are highly recommended. Information about PostgreSQL can be found at [www.postgresql.org](http://www.postgresql.org).

### • History

Stable32 began as a set of programs for the HP-85 computer in 1980. When the IBM PC was introduced in 1981, a DOS version of Stable was written for it in 1982. The name, besides its obvious implications for frequency stability, came from the name of a bar in Nashua, NH. Its initial application was to support the development of rubidium clocks for the GPS navigation satellites. Stable was ported to Microsoft Windows as Stable/Win in 1992, and then to its current Stable32 form in 1997 when Windows evolved to 32 bits.

### • Acknowledgments

The author wishes to acknowledge the contributions of many colleagues in the Time and Frequency community who have contributed the analytical tools that are so vital to this field. Some of those persons are listed in the References help topic. David W. Allan has had a major impact on this field, and has been a source of much inspiration and encouragement for the Stable programs. The work of the late J.A. Barnes influenced the drift analysis and noise recognition techniques, while that of C.A. Greenhall, C. Ekstrom and P. Koppang has helped with the numerical methods and confidence limits. T. Walter has contributed particularly toward the noise simulation, as has G. Winkler toward outlier detection. Credit also goes to M.A. Weiss for work regarding modified and time sigma, and D.A. Howe for the new total variances and for contributing many beneficial suggestions toward this program. More recently, important work was done by D.A. Howe, C.A. Greenhall, F. Vernotte and T.K. Pepler on implementing the modified and Hadamard total variances, while D.B. Percival & A.T. Walden's book and J.M. Lees & J. Park's code supported multitaper spectral analysis. D.A. Howe and T.N. Tasset have contributed significantly with the Thèo1 statistic. Perkin Elmer, Inc. supported the development of the Database function, and Matthew Phillips of Timing Solutions Corporation assisted in its writing.

### Contributors

Many people have contributed to the development of Stable32, reporting problems and offering ideas for new features. Some of those persons are as follows:

- Jim Barnes, Dave Allan and Dave Howe for the statistical foundations.
- Sam Stein for suggesting the GraphiC library.
- Ed Powers for early evaluation and suggestions.
- Ken Lyon for user interface suggestions.
- Gernot Winkler for practical robust statistics.
- Todd Walter for power-law noise generation.
- Steve Kelley for GraphiC support.
- Dave Howe for total statistics.
- Trudi Pepler for programming collaboration.
- Chuck Greenhall for many contributions related to edf.
- Jack Kusters for the TDEV loci on MDEV plots.
- Paul Koppang for suggesting the 1 of 2 option.
- Paul Koppang and Chris Ekstrom for the 3-cornered hat edf.
- Robert Lutwak for many suggestions, including the automation feature.
- Ken Senior for careful checking of the Sigma and Run results.
- Patrizia Tavella for unequally spaced data suggestions, and dynamic stability analysis.
- Tom Becker for user interface bug reports.
- Doug Hogarth and John Hartnett for bug reports.
- Richard Hambly and Tom Clark for suggesting Excel date compatibility.
- Tom Van Baak for the many tau idea.
- Dave Howe and Tiffany Tasset for Thêo1, ThêBR, and ThêoH.
- Nino De Falcis for suggesting the stability plot masks.

Thanks all those who have contributed ideas toward, and reported bugs for, the Stable program. I hope this latest version will also prove useful.

## Stable32 Functions

### Stable32 Functions

The Stable32 program includes the following functions:

#### File Functions

- Open Function
- Read Archive
- Multicolumn Data
- Large Data File
- Data File Opened
- Add Function
- Save Function
- Database Function
- Read Function
- Print Function

#### Editing Functions

- Edit Function
- Find Function
- Replace Function
- Fill Function
- Scale Function
- Part Function
- Regularize Function
- Timetags Function
- Normalize Function
- Average Function
- Clear Function

#### Conversion Functions

- Convert Function
- Domain Function

#### Plot Functions

- Plot Function
- Plot Options
- Plot Lines
- Plot Notes

#### Analysis Functions

- Sigma Function
- Run Function
- Run Options
- Run Lines
- Run Notes
- Dynamic Stability Function
- Check Function
- Statistics Function
- Drift Function
- Plot Function
- Autocorrelation Function

### Analysis Functions (Con't)

- Power Function
- Histogram Function

### Miscellaneous Functions

- Calendar
- Noise Function
- Configure
- Filenames
- Auto#
- 3-Cornered Hat Function

## Special Characters

- **Special Characters**

The Stable32 GraphiC graphics functions make use of several special text characters. The symbols [, ], ^, `, # and @ have special meanings.

Superscripts and subscripts may be entered into text strings as [super] and [sub] respectively. Text color may be changed by entering [N], where N is the standard IBM color numbers shown below. For example, use |2| for green text to annotate a noise line. The ^, ` and # symbols are not used by Stable32.

To use the symbols [, ], ^, `, #, and @ normally, these characters should be preceded by @. For example, use @# and @@ to place # and @ into the text.

<u>#</u>	<u>Color</u>
0	Black
1	Blue
2	Green
3	Cyan
4	Red
5	Magenta
6	Brown
7	White
8	Dark Gray
9	Light Blue
10	Light Green
11	Light Cyan
12	Light Red
13	Light Magenta
14	Yellow
15	Bright White

Stable32 uses four special characters to control the font and symbol set used for plot labels. These control characters are inserted into the label text for plot labels (with NUM LOCK on) by pressing the ALT key and using the numeric keypad to enter one of the 4-digit numbers shown in the following table:

<u>Font</u>	<u>Code</u>	<u>Character</u>
News	0200	È
Swiss	0201	É
Newsgrm	0202	Ê
Simplex	0203	Ë

The GraphiC News and Newsgrm fonts are used for all plot titles, sub-titles and axis labels. For example, to insert the lower case Greek letter  $\tau$  into a

stability plot x-axis label, use the character sequence ÊtÈ. This will shift to the Greek-math symbol set of the Newsgrm font, changing the character t to a τ, and then return to the normal News font. Lower case Greek letters are entered with the characters a through z in a fashion similar to the Windows Symbol set. The News font is defined for codes between 33 and 199, while the Newsgrm font for 33 to 146.

The News, Swiss, Newsgrm and Simplex fonts are shown in the tables on the following pages.

### news.fnt

33	!	50	2	67	C	84	T	101	e	118	v	135	ç	152	ÿ	169	“	186	í
34	"	51	3	68	D	85	U	102	f	119	w	136	ê	153	Ö	170	”	187	î
35	#	52	4	69	E	86	V	103	g	120	x	137	ë	154	Û	171	Á	188	ï
36	\$	53	5	70	F	87	W	104	h	121	y	138	è	155	€	172	Â	189	ò
37	%	54	6	71	G	88	X	105	i	122	z	139	ï	156	£	173	Ï	190	ó
38	&	55	7	72	H	89	Y	106	j	123	{	140	í	157	¥	174	È	191	ô
39	'	56	8	73	I	90	Z	107	k	124		141	ì	158	¤	175	É	192	ù
40	(	57	9	74	J	91	[	108	l	125	}	142	Ā	159	f	176	ā	193	ú
41	)	58	:	75	K	92	\	109	m	126	~	143	Ă	160	á	177	ä	194	û
42	*	59	;	76	L	93	]	110	n	127		144	É	161	í	178	Ë	195	ÿ
43	+	60	<	77	M	94	^	111	o	128	Ç	145	æ	162	ó	179	Ì	196	ß
44	,	61	=	78	N	95	_	112	p	129	ü	146	Æ	163	ú	180	œ	197	«
45	-	62	>	79	O	96	'	113	q	130	é	147	ô	164	ñ	181	Œ	198	»
46	.	63	?	80	P	97	a	114	r	131	â	148	ö	165	Ñ	182	À	199	•
47	/	64	@	81	Q	98	b	115	s	132	ë	149	ò	166	ª	183	Ã		
48	0	65	A	82	R	99	c	116	t	133	à	150	û	167	º	184	Ö		
49	1	66	B	83	S	100	d	117	u	134	ã	151	ù	168	¿	185	Ş		

News Font (Numeric Code = 0200)

### swiss.fnt

33	!	50	2	67	C	84	T	101	e	118	v	135	ç	152	ÿ	169	“	186	í
34	"	51	3	68	D	85	U	102	f	119	w	136	ê	153	Ö	170	”	187	î
35	#	52	4	69	E	86	V	103	g	120	x	137	ë	154	Û	171	Á	188	ï
36	\$	53	5	70	F	87	W	104	h	121	y	138	è	155	€	172	Â	189	ò
37	%	54	6	71	G	88	X	105	i	122	z	139	ï	156	£	173	Ï	190	ó
38	&	55	7	72	H	89	Y	106	j	123	{	140	í	157	¥	174	È	191	ô
39	'	56	8	73	I	90	Z	107	k	124		141	ì	158	¤	175	É	192	ù
40	(	57	9	74	J	91	[	108	l	125	}	142	Ā	159	f	176	ā	193	ú
41	)	58	:	75	K	92	\	109	m	126	~	143	Ă	160	á	177	ä	194	û
42	*	59	;	76	L	93	]	110	n	127		144	É	161	í	178	Ë	195	ÿ
43	+	60	<	77	M	94	^	111	o	128	Ç	145	æ	162	ó	179	Ì	196	ß
44	,	61	=	78	N	95	_	112	p	129	ü	146	Æ	163	ú	180	œ	197	«
45	-	62	>	79	O	96	'	113	q	130	é	147	ô	164	ñ	181	Œ	198	»
46	.	63	?	80	P	97	a	114	r	131	â	148	ö	165	Ñ	182	À	199	•
47	/	64	@	81	Q	98	b	115	s	132	ë	149	ò	166	ª	183	Ã		
48	0	65	A	82	R	99	c	116	t	133	à	150	û	167	º	184	Ö		
49	1	66	B	83	S	100	d	117	u	134	ã	151	ù	168	¿	185	Ş		

Swiss Font (Numeric Code = 0201)

# newsgrm.fnt

33	e	50	ƒ	67	S	84	≤	101	ε	118	)	135	ϕ	152		169	186
34	(	51	⊗	68	Δ	85	ŕ	102	φ	119	ω	136	...	153		170	187
35	≡	52	⊙	69	ℵ	86	↔	103	γ	120	ξ	137	ϕ	154		171	188
36	π	53	™	70	⊕	87	⊖	104	χ	121	ψ	138	⊗	155		172	189
37	↑	54	≡	71	Γ	88	⊖	105	ι	122	ζ	139	⊕	156		173	190
38	√	55	≡	72	⊖	89	ψ	106	ι	123	}	140	⊕	157		174	191
39	'	56	8	73	Σ	90	≥	107	κ	124	}	141	⊖	158		175	192
40	∩	57	∇	74	⊥	91	⌈	108	λ	125	}	142	⊖	159		176	193
41	∪	58	⇒	75	<	92	⊕	109	μ	126	8	143	♀	160		177	194
42	×	59	⇌	76	Δ	93	⌋	110	v	127		144	⌋	161		178	195
43	±	60	≤	77	>	94	∩	111	o	128	*	145	∇	162		179	196
44	∫	61	≠	78	∠	95	↓	112	π	129	≡	146	∧	163		180	197
45	±	62	≥	79	°	96	"	113	θ	130	←	147		164		181	198
46	•	63	≈	80	Π	97	α	114	ρ	131	↑	148		165		182	199
47	÷	64	∩	81	⊕	98	β	115	σ	132	→	149		166		183	
48	∇	65	ε	82	P	99	η	116	τ	133	↓	150		167		184	
49	ℵ	66	∴	83	Σ	100	δ	117	v	134	↔	151		168		185	

Newsgrm Font (Numeric Code = 0202)

# simplex.fnt

33	!	50	2	67	C	84	T	101	e	118	v	135	ç	152	ÿ	169	“	186	í
34	"	51	3	68	D	85	U	102	f	119	w	136	ê	153	Ö	170	"	187	î
35	#	52	4	69	E	86	V	103	g	120	x	137	ë	154	Ü	171	Á	188	ï
36	\$	53	5	70	F	87	W	104	h	121	y	138	è	155	ϕ	172	Â	189	ò
37	%	54	6	71	G	88	X	105	i	122	z	139	ï	156	£	173	Ï	190	ó
38	&	55	7	72	H	89	Y	106	j	123	{	140	í	157	¥	174	È	191	ô
39	'	56	8	73	I	90	Z	107	k	124		141	ì	158	¤	175	Ê	192	ù
40	(	57	9	74	J	91	[	108	l	125	}	142	Ä	159	f	176	ǎ	193	ú
41	)	58	:	75	K	92	\	109	m	126	~	143	Å	160	đ	177	õ	194	û
42	*	59	;	76	L	93	]	110	n	127		144	É	161	í	178	Ë	195	ÿ
43	+	60	<	77	M	94	^	111	o	128	Ç	145	æ	162	ó	179	Ì	196	ß
44	,	61	=	78	N	95	_	112	p	129	ü	146	Æ	163	ú	180	œ	197	«
45	-	62	>	79	O	96	'	113	q	130	é	147	ô	164	ñ	181	Œ	198	»
46	.	63	?	80	P	97	a	114	r	131	â	148	ó	165	Ñ	182	À	199	°
47	/	64	@	81	Q	98	b	115	s	132	á	149	ò	166	°	183	Ã		
48	0	65	A	82	R	99	c	116	t	133	à	150	û	167	°	184	Ö		
49	1	66	B	83	S	100	d	117	u	134	â	151	ù	168	¿	185	ÿ		

Simplex Font (Numeric Code = 0203)

### Editing Keys

- **Editing Keys**

The Stable32 program uses the standard Windows editing keys except that the **Enter** key can generally be used to terminate edit control entries. This causes the entry to be reformatted and checked for validity.

The following special editing keys that apply to the Edit dialog box:

**F2** The existing data may be edited by first pressing the F2 key. It is necessary to delete a portion of the existing data to make room for the new entries.

**Esc** The original value may be restored with the Esc key.

**Enter** The changes are accepted with the Enter key.

## Appendix IV

### Configuration File

- **[Files]**

The [Files] section of the STABLE32.INI file can contain the following parameters:

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
LastFile	Full filename of last file opened	None
PhaseFile1	Full filename of last phase file opened	None
FreqFile1	Full filename of last freq file opened	None
PhaseFile2	Full filename of 2nd from last phase file opened	None
FreqFile2	Full filename of 2nd from last freq file opened	None
PhaseFile3	Full filename of 3rd from last phase file opened	None
FreqFile3	Full filename of 3rd from last freq file opened	None
Ext	Additional file extension patterns	None

The Ext parameter takes the form Ext=Label (\*.ext)\*.ext|, where the Label part defines the label for the pattern and ext defines the filename extension. Both parts must end with a | character. These filename patterns will be added to the standard ones in the Open and Save functions.

- **[Preferences]**

The [Preferences] section of the STABLE32.INI file can contain the following parameters:

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
ShowFileOpened	Show the Opened dialog box whenever a data file is opened	0
UseFreqPlotIni	Use INI file data for frequency plot parameters	0
UsePhasePlotIni	Use INI file data for phase plot parameters	0
UseSigmaPlotIni	Use INI file data for sigma plot parameters	0
UseHistoPlotIni	Use INI file data for histogram plot parameters	0
UseSpectPlotIni	Use INI file data for spectrum plot parameters	0
UseAutoPlotIni	Use INI file data for ACF plot parameters	0
UseDAVARPlotIni	Use INI file data for DAVAR plot parameters	0
DataType	Select Phase or Freq data ("Phase" selects phase data, any other text selects freq)	Phase
LabelPlot	Add a label to plots	0
PlotLabel	Label text for plots	Null
IncExtension	Increment the filename extension when data changes	0
WriteSigmaFile	Write a file containing sigma data after a Run	0
AutoCalc	Automatically calculate sigmas when starting Run function	0
AutoPlot	Automatically draw plot when starting Plot function	0
Notepad	Use the specified text editor	NOTEPAD.EXE
MaxSize	Maximum size of data arrays	99999

Splash	Show startup splash screen	1
SigmaType	Variance type for Sigma function	0
RunType	Variance type for Run fiction	0
PhaseDriftType	Drift type for frequency data	0
FreqDriftType	Drift type for frequency data	0
NonNums	# non-numeric characters to skip in lines of input data	0
Comma	Use comma instead of decimal point in input data	0
WarnZeroFreq	Warn about zero frequency points in input data	1
SmartScale	Use smart y-axis data plot scales	1
Left	Left main window screen position	0
Top	Top main window screen position	0
Right	Right main window screen position	0
Bottom	Bottom main window screen position	0
LastOpened	Type of data last opened	0
OpenScale	Flag to scale input data	0
Header	Flag to write header in data file	0
NoGreek	Use Greek characters in stability plot labels	1
ShowWelcome	Flag to show welcome screen	1
CheckDetails	Flag for detailed check box	0
DriftDetails	Flag for detailed drift box	0
SigmaDetails	Flag for detailed sigma box	0
WideEdit	Flag for wide edit box	0
NormFreq	Flag to normalize frequency during conversion	0
XferTags	Flag to transfer timetags during conversion	0
ManyTau	Flag for many tau run	1
ManySize	Many tau parameter	500
UseTags	Flag to use timetags for tau	0

A zero right coordinate gives a main window size that fills the desktop area.

• **[FreqPlot]**

The [FreqPlot] section of the STABLE32.INI file can contain the following parameters:

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
Title	Plot title text	Null
Subtitle	Plot subtitle text	Null
Message	Message text	Null
X-Label	X-axis (time) label text	Null
Y-Label	Y-axis (freq) label text	Null
PlotFile	*.TKF filename	Null
Position	Message position	0
X-Div	# x-axis divisions	5
Y-Div	# y-axis divisions	5
Y-Tic	# y-axis tics	5
Line	Line type	0
Mouse	Mouse flag	0

Date	Data annotation flag	0
Box	Draw box around plot flag	0
Wide	Use wide plot line flag	0
X-Offset	X-axis scale offset	0.0
FPlotX	Frequency data plot x-axis label #	0
X-Mult	X-axis scale multiplier	1.0
Y-Offset	Y-axis scale offset	0.0
Y-Mult	Y-axis scale multiplier	1.0
Y-Min	Y-axis scale minimum value	0.0
Y-Max	Y-axis scale maximum value	1.0
Fit	Draw fit line flag	0
FXDefault		

• **[PhasePlot]**

The [PhasePlot] section of the STABLE32.INI file can contain the following parameters:

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
Title	Plot title text	Null
Subtitle	Plot subtitle text	Null
Message	Message text	Null
X-Label	X-axis (time) label text	Null
Y-Label	Y-axis (phase) label text	Null
PlotFile	*.TKF filename	Null
Position	Message position	0
X-Div	# x-axis divisions	5
Y-Div	# y-axis divisions	5
Y-Tic	# y-axis tics	5
Line	Line type	0
Mouse	Mouse flag	0
Date	Date annotation flag	0
Box	Draw box around plot flag	0
Wide	Use wide plot line flag	0
Fit	Draw fit line flag	0
X-Offset	X-axis scale offset	0.0
X-Mult	X-axis scale multiplier	1.0
Y-Offset	Y-axis scale offset	0.0
Y-Mult	Y-axis scale multiplier	1.0
Y-Min	Y-axis scale minimum value	0.0
Y-Max	Y-axis scale maximum value	1.0
PPlotX	Phase data plot x-axis label #	0
PXDefault		

• **[SigmaPlot]**

The [SigmaPlot] section of the STABLE32.INI file can contain the following parameters:

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
Title	Plot title text	Null
Subtitle	Plot subtitle text	Null
Message	Message text	Null

PlotFile	Plot *.TKF filename	Null
Position	Message position	0
X-Min	X-axis scale minimum value	Null
X-Max	X-axis scale maximum value	Null
Y-Min	Y-axis scale minimum value	Null
Y-Max	Y-axis scale maximum value	Null
TablePos	Table position	1
Wide	Use wide plot line flag	0
X-Label	X-axis (tau) label text	Null
Y-Label	Y-axis (sigma) label text	Null
StabFile	Stability data filename	Null
Date	Date annotation flag	0
Box	Draw box around plot flag	0
Table	Draw sigma-tau table on plot flag	1
ErrorBar	Show error bars on stability plot flag	1
Fit	Draw fit line flag	0

- **[SpectPlot]**

The [SpectPlot] section of the STABLE32.INI file can contain the following parameters:

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
Title	Plot title text	Null
Subtitle	Plot subtitle text	Null
Message	Message text	Null
Position	Message position	0
Wide	Use wide plot line flag	0
Y-Tic	# y-axis tics	5
Date	Date annotation flag	0
Box	Draw box around plot flag	0
Fit	Draw fit line flag	0
Points	Show PSD points on plot flag 1	
PSD	PSD type	0
WinType	FFT windowing type	1
Win#	# FFT windows	1
Mouse	Mouse flag	0
Info	Show PSD info on plot flag	0
Y-Min	Y-axis scale minimum value	Null
Y-Max	Y-axis scale maximum value	Null
Freq	Carrier frequency (Hz)	0

- **[HistoPlot]**

The [HistoPlot] section of the STABLE32.INI file can contain the following parameters:

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
Title	Plot title text	Null
Subtitle	Plot subtitle text	Null
Message	Message text	Null
Position	Message position	0

- **[AutoPlot]**

The [AutoPlot] section of the STABLE32.INI file can contain the following parameters:

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
Title	Plot title text	Null
Subtitle	Plot subtitle text	Null
Message	Message text	Null
Position	Message position	0
Wide	Use wide plot line flag	0
Date	Date annotation flag	1
Box	Draw box around plot flag	1
Zero	Draw horizontal line at 0 flag	1
Lag1	Show lag k scatter plot flag	1
Lag#	Scatter plot lag #	1
Lags	# lags on plot	1
Mouse	Mouse flag	0
AutoFile	ACF data filename	Null
PlotFile	ACF *.TKF plot filename	Null
AF	Averaging factor	1

- **[DAVARPlot]**

The [DAVARPlot] section of the STABLE32.INI file can contain the following parameters:

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
Position	Message position	0
Date	Date annotation flag	0
Box	Draw box around plot flag	0
SigmaType	Allan or Hadamard sigma type	0
WindowSize	Analysis window size	100
StepSize	Analysis step size	10
Colors	Use colors flag	0
Title	Plot title text	Null
Subtitle	Plot subtitle text	Null
Message	Plot message text	Null
Grid	Show plot grid flag	1

- **[FreqLines]**

The [FreqLines] section of the STABLE32.INI file can contain the following parameters (the items are repeated for each line, as indicated by the index [i], i=1 to 12):

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
Type[i]	Line type (0-5, 0=None)	0
X-Start[i]	X-axis start point	0.0
X-End[i]	X-axis end point	0.0
Y-Start[i]	Y-axis start point	0.0
Y-End[i]	Y-axis end point	0.0
Slope[i]	Line slope (0-5)	0

- **[PhaseLines]**

The [PhaseLines] section of the STABLE32.INI file can contain the following parameters (the items are repeated for each line, as indicated by the index [i], i=1 to 12):

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
Type[i]	Line type (0-5, 0=None)	0
X-Start[i]	X-axis start point	0.0
X-End[i]	X-axis end point	0.0
Y-Start[i]	Y-axis start point	0.0
Y-End[i]	Y-axis end point	0.0
Slope[i]	Line slope (0-5)	0

- **[SigmaLines]**

The [SigmaLines] section of the STABLE32.INI file can contain the following parameters (the items are repeated for each line, as indicated by the index [i], i=1 to 12):

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
Type[i]	Line type (0-5, 0=None)	0
X-Start[i]	X-axis start point	0.0
X-End[i]	X-axis end point	0.0
Y-Start[i]	Y-axis start point	0.0
Y-End[i]	Y-axis end point	0.0
Slope[i]	Line slope (0-5)	0
MaskFile[i]	Mask filename	Null

- **[FreqNotes]**

The [FreqNotes] section of the STABLE32.INI file can contain the following parameters (the items are repeated for each note, as indicated by the index [i], i=1 to 8):

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
Size[i]	Font point size	8
Border[i]	Border flag	0
X-Position[i]	X-axis position	0.0
Y-Position[i]	Y-axis position	0.0
Text[i]	Note text	Null

- **[PhaseNotes]**

The [PhaseNotes] section of the STABLE32.INI file can contain the following parameters (the items are repeated for each note, as indicated by the index [i], i=1 to 8):

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
Size[i]	Font point size	8
Border[i]	Border flag	0
X-Position[i]	X-axis position	0.0
Y-Position[i]	Y-axis position	0.0
Text[i]	Note text	Null

- **[SigmaNotes]**

The [SigmaNotes] section of the STABLE32.INI file can contain the following parameters (the items are repeated for each note, as indicated by the index [i], i=1 to 8):

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
Size[i]	Font point size	8
Border[i]	Border flag	0
X-Position[i]	X-axis position	0.0
Y-Position[i]	Y-axis position	0.0
Text[i]	Note text	Null

- **[Toolbuttons]**

The [Toolbuttons] section of the STABLE32.INI file can contain the following flag parameters (the order has no effect):

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
Open	Show Open function toolbar	0
Add	Show Add function toolbar	0
Read	Show Read function toolbar	0
Save	Show Save function toolbar	0
Conv	Show Convert function toolbar	0
Stats	Show Statistics function toolbar	0
Check	Show Check function toolbar	0
Run	Show Run function toolbar	0
Norm	Show Normalize function toolbar	0
Plot	Show Plot function toolbar	0
Drift	Show Drift function toolbar	0
Histo	Show Histogram function toolbar	0
Cal	Show Calendar function toolbar	0
Avg	Show Average function toolbar	0
Print	Show Print function toolbar	0
Noise	Show Noise function toolbar	0
Clear	Show Clear function toolbar	0
Power	Show Power function toolbar	0
Edit	Show Edit function toolbar	0
Sigma	Show Sigma function toolbar	0
Fill	Show Fill function toolbar	0
Scale	Show Scale function toolbar	0
Play	Show Play program toolbar	0
Tag	Show Timetag function toolbar	0
Reg	Show Regularize function toolbar	0
Dom	Show Domain function toolbar	0
Part	Show Part function toolbar	0
Help	Show Help function toolbar	0
Pad	Show Notepad function toolbar	0
DVAR	Show DAVAR function toolbar	0
Auto1	Show Auto1 function toolbar	0
Auto2	Show Auto2 function toolbar	0

Auto3	Show Auto3 function toolbar	0
ACF	Show ACF function toolbar	0
Conf	Show Configure function toolbar	0
File	Show Filenames function toolbar	0
Comm	Show 5110Comm program toolbar	0
Dbase	Show Database function toolbar	0

- **[Inputs]**

The [Inputs] section of the STABLE32.INI file can contain several read-only items that serve as user-selectable default values for the following parameters:

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
PhaseTau	Tau value for phase data	1.0
FreqTau	Tau value for frequency data	1.0
PhaseAddend	Default value to be added to phase data in Scale function	0.0
FreqAddend	Default value to be added to frequency data in Scale function	0.0
PhaseMultiplier	Default factor to multiply phase data by in Scale function	1.0
FreqMultiplier	Default factor to multiply frequency data by in Scale function.	1.0
PhaseTagScale	Timetag scale multiplier for phase data	1.0
FreqTagScale	Timetag scale multiplier for freq data	1.0
PhaseTagOffset	Timetag scale multiplier for phase data	1.0
FreqTagOffset	Timetag scale multiplier for freq data	1.0

If there is no corresponding item in the [Inputs] section of the INI file, the default value is used. Otherwise, the one in the INI file overrides that value. The tau number can be changed in the Opened dialog box or by editing the value in the status bar. The addend and multiplier values do not take effect unless the Scale function is executed. These items are very useful for setting the tau to a standard measurement interval time, for adding a frequency offset, and for scaling the data by a particular factor (e.g. microseconds). If there is timetag data, and no tau [Input], then the spacing of the first two timetags, scaled by the TagScale value, is used as the default tau when a data file is opened. The TagOffset is applied as the data file is read.

- **[Archive]**

The [Archive] section of the STABLE32.INI file can contain the following parameters:

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
Num	# Channels	4
Meas	Measurement Channel #	2
Ref	Reference Channel #	1

The [Archive] section of the INI file contains the default values set by the Read Archive Configure function.

- **[Database]**

The [Database] section of the STABLE32.INI file can contain the following parameters:

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
Host	Host ID address	1
Database	Database name	tsc-postgres
User	User name	tsc
Password	Password	tsc (encrypted)

- **[Auto#]**

The [Auto#] sections (where #=1-3) of the STABLE32.INI file can contain the following parameters that define the three automation scripts:

<u>Keyword</u>	<u>Description</u>	<u>Default</u>
FileOpen	Flag to open data file	0
Convert	Flag to convert data type	0
Outliers	Flag to remove outliers	0
Average	Flag to average data	0
Drift	Flag to remove drift	0
PhaseDataPlot	Flag to plot phase data	0
FreqDataPlot	Flag to plot frequency data	0
Run	Flag to run stability analysis	0
StabPlot	Flag to print stability plot	0
StabTable	Flag to print stability table	0
Close	Flag to close Auto# dialog box	0
DataPlot	Flag to plot data	0
DataPrint	Flag to print data	0
PrintOriginal	Flag to print original data	0
PrintCorrected	Flag to print corrected data	0
PrintSelected	Flag to print selected data	0
PrintAveraged	Flag to print averaged data	0
PrintResidual	Flag to print residual data	0
RunPlot	Flag to plot stability	0
Part	Flag to use part of data	0
Delay	Flag to add extra delay	0
Script	Script name	Script_#
PhaseVariance	Variance type for phase data	Normal
FreqVariance	Variance type for freq data	Normal
Filename	Data filename to open	Null
PhaseDriftType	Drift fit type for phase data	Quadratic
FreqDriftType	Drift fit type for freq data	Linear
AvgFactor	Averaging factor	1
DataType	Data type (0=Phase, 1-Freq)	0
AnalysisType	Stability analysis type	0
SigmaFactor	Outlier sigma factor	5.000
DelayTime	Extra delay time in msec	0
Start	Start data point #	1
End	End data point # (0=last point)	0

### Hints and Notes

- **Launching Stable32**

1. Stable32 can be launched like any Windows application from the Start menu, from the Run dialog or via an icon on the desktop. By associating Stable32 with a filename extension (such as \*.dat), Stable32 may also be launched by clicking on a data file or by dragging a data file to the Stable32 icon on the desktop.

- **Main Window**

1. The Stable32 main window has the best appearance and functionality with XGA (1024x768) or higher screen resolution, and with the main window adjusted for nearly full width. This allows the use of the mini data plots in the Stable32 status bar, and all the toolbuttons. If only a lower screen resolution is available, use the Configure function to choose the desired toolbuttons, and use the Stats function as an alternate way to observe the current data. The position of the main window is saved between runs.

- **GraphiC Plots**

1. When performing a metafile file format conversion for a GraphiC plot within Stable32, make sure that the target path and filename comply with DOS filename conventions (e.g. no more than 8 characters). Otherwise, the file creation process will produce an error, and the conversion will fail. For example, to use the operating system default path C:\My Documents, the directory name must be shortened to mydocu~1.
2. An alternative file format conversion method for bitmaps and metafiles is to use the Edit/Copy GraphiC menu command. This is the easiest way to transfer a plot into another open document.
3. The best format for including GraphiC (.tkf) plots in a Microsoft Word document is as a Windows metafile (.wmf).
4. GraphiC (.tkf) plots can be included in web documents as .gif files by first converting them into bitmaps and then using the free conversion program BMPToGIF (available on the Hamilton Technical Services web site). This is the best format for including GraphiC plots in an html document.
5. Better-looking data plot scales can be obtained by scaling the yaxis to pp10<sup>xx</sup> or standard units. This will also eliminate duplicate scale markings and the colon that can appear in them. This format is the default when the Smart Plot Scales configuration item is selected. In the Y-Axis Label edit control of the Plot Options dialog, it is important to understand that [] brackets will create a superscript (see Appendix II).
6. Problems have been reported while printing plots after reading data from read-only network drives because the GraphiC functions are unable to write the temporary metafile needed for printing. One solution to this problem is to copy the data file to a local drive (for which you have write privileges) before opening it for analysis.

7. Do not access the GraphiC menu items or resize the plot while it is drawing. Problems have been reported on some systems if this is done.
8. See the Plot Options section for information about the cursor feature.

- **Stability Analysis**

1. Repetitive Stable32 stability analysis procedures can be automated with WinBatch scripts. This can save time, assure consistency and allow the analysis to be done by less-experienced personnel (see batch Files below).
2. Time can be saved, and errors avoided, during a complex analysis by first setting up the data and plot filenames with the Files function.
3. Enter the tau value in the Data File Opened dialog box, as a PhaseTau or FreqTau [Inputs] item in the configuration file, or automatically using timetag data (without the [Inputs] values). In the latter case, the tau value can be scaled to seconds with the TagScale [Inputs] item. The resulting tau value may have some small rounding error.
4. Observe the blue symbol at the top center of the status bar that indicates if the two data types are synchronized.

- **Large Data Sets**

1. The most important factor for analyzing large data sets with Stable32 (like most Windows applications) is RAM memory size, not processor speed.
2. If possible, average large data sets before beginning an analysis run.
3. Use the Stop button on All Tau runs if the full results are not needed at longer averaging factors.
4. The maximum size data file that can be read is set by the Max Data File Size item in the Configure function. The purpose of this setting is to automatically activate the data averaging dialog box when a large data file is read. Set this parameter to a value appropriate to your computer's speed and RAM.
5. The Stats, Edit and Check functions can be very slow when processing frequency data having many repeated values. This is because of the qsort() function used in finding the median value. One way to work around this problem is to add a very small slope to the data with the Scale function in order to eliminate exactly identical data points.

- **Convenience Features**

1. Use the [Inputs] section of the STABLE32.INI configuration file to automatically set the tau and scale factor you use most often.
2. Use the stored text in the plot options drop-down combo controls to avoid having to reenter labels between plots.
3. Use the Save and Read buttons to store and recall complete plot configurations.
4. Associate the .dat filename extension with the Stable32 program to automatically launch it by clicking on a data file, with the Open command on the Explorer File menu, or by dragging the data to the Stable32 icon on the desktop.
5. Use the Copy command of the Drift, Sigma, Run and Domain functions to write their results to the clipboard.

6. The File Opened dialog box can be eliminated with the Show File Opened configuration item to speed data file entry.

- **Batch Files**

1. The WinBatch batch language program for Windows from Wilson WindowWare, Inc. is highly recommended to automate repetitive analysis procedures with Stable32. Information is available at <http://www.windowware.com>.

- **Screen Setup**

1. Use 1024 x 768 XGA screen resolution or higher, if possible, in order to support the mini data plots on the Stable32 status bar. This resolution also supports the use of all toolbar buttons.
2. With lower than XGA resolution, use the Configure function to select the toolbar buttons that you want.
3. The phase and frequency mini data plots can be eliminated by deselecting the Status Plots box in the Configure function. This can save screen space and reduce update time on older systems.

- **Data Output**

1. The results of an analysis can be documented or captured in several ways for use in another analysis or application. The results of a Check or Run can be printed. The results of a Sigma or Drift calculation, a stability Run, or a Domain conversion can be copied to the clipboard with the Copy button. Software such as the Microsoft ClipBook Viewer can be used to capture multiple outputs. The results of a Run are written to a stability file that can be read by the Read function or another application. All of the GraphiC plots can be printed, copied to the clipboard, or converted and saved in a number of file formats. And finally, a synopsis of a Stable32 analysis session is recorded in a log file.

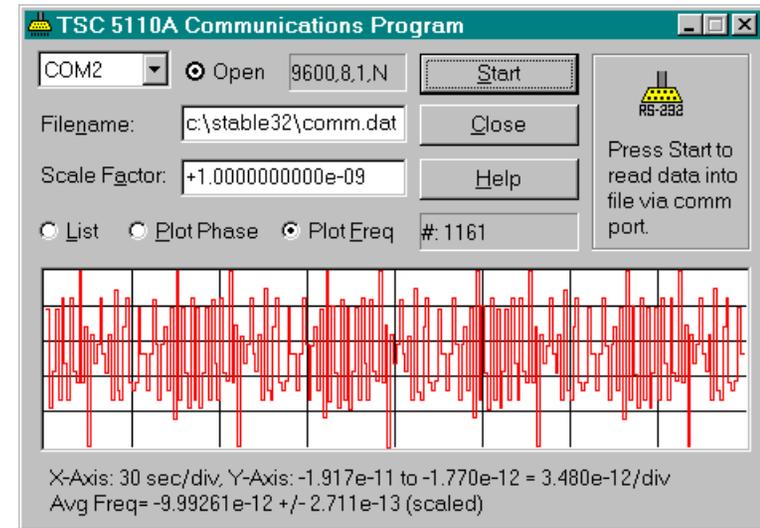
- **Run Function**

1. Don't press Stop during an octave or decade run until at least one result is displayed.
2. The analysis point being processed is repeated if the run is stopped, so the most efficient time to interrupt a run is immediately after a point is completed.
3. It may take a little while for the program to respond to the Stop button during a run with a large data set.
4. It may take a while for the progress bar to indicate anything when starting a run with a large data set.
5. The progress bar shows each point during a TOTMDEV, TOTTDEV or MTIE octave or decade run, and the overall progress during an all tau run. During the latter, the current averaging factor is displayed on the status bar.

### 5110Comm Communications Program

- **5110Comm**

5110Comm is a program to capture data from the Timing Solutions Corporation (TSC) Model 5110A Time Interval Analyzer for analysis by Stable32. It may be launched from within Stable32 by clicking on its toolbutton or the TSC 5110A item in the Utilities menu.



- **Condensed Operating Instructions**

1. Connect the TSC 5110A to a PC COM port using a null-modem cable.
2. Launch the 5110Comm program directly or from within Stable32.
3. Verify that the COM port is open by observing the  Open indicator or status message in the List box.
4. Enter the desired filename for the data file. Note: If there is an existing file with this name, it will be overwritten each time a new run is started.
5. Enter the scale factor to convert cycles of the A channel frequency into seconds. This is equal to the period of the A channel frequency (e.g. 100 nsec or 1e-7 for 10 MHz).
6. Press the Start button. Data capture will begin immediately at 1-second intervals, or as soon as the TSC 5110A is started.
7. The captured data can be displayed as a listing, a phase plot, or a frequency plot by selecting the desired Radiobutton above the display area.
8. The data listing shows the data point number, the raw TSC 5110A data (in cycles of the A channel frequency), and the scaled data file values (in seconds of phase difference), as labeled below the listing.

9. For plots, the annotation in the first line below the display shows the plot scales. The X-axis scale is shown in sec/div. The Y-axis scale is shown as the maximum and minimum data values and the scale factor per division. Phase plots have units of seconds, while frequency plots show dimensionless fractional frequency. For plots, the average fractional frequency and its standard error are shown on the lower line.
10. The current data point number is shown in the text box above the plot. This number is useful for tagging purposes if a measurement condition (such as supply voltage) is changed.
11. During data capture, all the controls in the 5110Comm dialog box are disabled except for the display selection, window minimization (–), and the Stop button.
12. If the 5110Comm dialog box is minimized, its icon in the system tray displays the data point number to show the progress of the run.
13. It is OK to read data from the 5110Comm data file into Stable32 “on the fly” during a run. In that way, a preliminary analysis of the results can be performed while continuing to capture data.
14. Access to the help file during a run can be obtained by launching another instance of 5119Comm.
15. After a run, the on-screen portion of the listing, phase plot and frequency plot can be examined. And, of course, the data file can be read into Stable32 for a complete analysis.
16. The 5110Comm program uses a standard Windows initialization file, 5110Comm.ini, to hold configuration information about the program. This file retains the previous COM port, data filename, scale factor and display mode. It can also be manually edited with a text editor such as Notepad to change those and other parameters. In particular, the number of listing and plot points can be set with the ListSize and PlotSize items in the [Preferences] section. Because the plots have 8 X-axis divisions, it is best if the PlotSize value have a “round” number after division by 8 (e.g. PlotSize=240 will provide a scale of 30 sec/div).

- **Note**  
The 5110Comm program is not included with all versions of the Stable32 software package.

## Index

### A

Acknowledgments.....	247
Add Function .....	105
Allan Variance.....	31, 33, 155, 193
Analysis Limits.....	9
Analysis Procedure .....	241
Archive File .....	15, 91
AutoCalc .....	28, 150, 154, 160, 193, 259
Autocorrelation .....	73, 167
Automated Operation .....	23
Auto# Function .....	215
AutoPlot.....	28, 173, 259
Average Function.....	139
Averaging, Data.....	98, 139
Averaging, Large Files.....	11, 28, 97

### B

Bias Functions.....	67
Bisection.....	153
Bitmap (bmp) Files.....	269
Bitmaps .....	10

### C

Calendar Function .....	221
Check Function.....	149
Clear Data .....	191
Clear Function.....	191
Clear Timetags.....	191
Command Line Options .....	21
Commas (Instead of Decimal Points).....	12
Confidence Intervals.....	61
Configuration File .....	27, 29, 259
Configure Function.....	27
Convert Function.....	135
Copy to Clipboard.....	156, 187, 193, 214, 241, 271

### D

Data Arrays.....	5
Data Averaging when Opening .....	89, 97
Database .....	107
Data File Opened.....	99
Data Format .....	11
Data Plot Function.....	175

Day of Year (DOY).....	221
Dead Time.....	71
Dialog Boxes .....	5
Domain Conversions.....	83, 213
Domain Function.....	213
Drift Function.....	151
Dynamic Stability .....	59
Dynamic Stability Function.....	207

## E

Edit Function.....	127
Editing Keys .....	257
E-Mail .....	2
Error Bars .....	61
Example .....	235

## F

Fax.....	2
Feedback (Help Menu Item) .....	7
File Functions .....	85
File Opened (Data File Opened).....	99
Filenames .....	17
Filenames Function.....	87
Fill Function .....	141
Fill Gaps .....	12, 141
Find Function .....	131
Find Interruptions (Gaps) .....	211
Fonts .....	3, 251
Format Conversions.....	269
Frequency Domain Stability.....	75

## G

Gaps.....	12, 211
Generate Simulated Clock Noise. ....	187
Generate Timetags .....	185
Glossary .....	227
GPS Week.....	222
GraphiC.....	1, 3, 225, 247
Graphics Interchange Format (gif) Files .....	269

## H

Hadamard Variance.....	32, 45, 155, 193
Hamilton Technical Services .....	2
Help .....	1
Help Files .....	3

Hints and Notes .....	269
Histogram Function .....	171

## I

Index.....	275
Inputs (Function) .....	101
Inputs (from Configuration File).....	266, 270
Insert Gaps .....	211
Insert Plot.....	180
Installation .....	3
Introduction.....	1

## J

Julian Date.....	222
------------------	-----

## K

Keys, Editing.....	257
--------------------	-----

## L

Lag 1 Autocorrelation.....	73, 168
Large Data Files .....	97
License.....	2
Log File .....	25

## M

Main Window .....	5
Many Tau.....	195
Masks .....	202
Mean Value.....	137, 147
Median.....	13, 147, 150
Menu Tree.....	7
MIL-O-55310B.....	154
Modified Allan Variance.....	37
Modified Julian Date (MJD).....	221, 222
MTIE .....	55
Multicolumn Data.....	95

## N

Noise Function.....	187
Noise Processes.....	77
Noise Spectra .....	81
Non-Numeric Characters.....	11
Normalize Function.....	137

Notepad (Pad) .....	223
---------------------	-----

## O

One-of-Two Scaling .....	195
Opaque Plot Notes .....	183, 205
Open Function.....	89
Outliers.....	12, 149
Overlapping Allan Variance .....	35
Overlapping Hadamard Variance .....	47

## P

Pad Function.....	223
Part Function .....	145
Phase Noise Diagram.....	82
Picinbono Variance .....	45
Play Function .....	225
Play/Win.....	225
Plot Annotations .....	183, 205
Plot Function.....	173
Plot Inserts .....	180
Plot Lines .....	181
Plot Notes.....	183
Plot Options .....	177
Power Function.....	161
Power Law Noise .....	77
Power Spectral Density (PSD).....	77, 161
Precision.....	11, 131, 133
Program Model.....	5
Print Function.....	125

## Q

Quadratic .....	152
-----------------	-----

## R

Read Archive Function .....	91
Read Archive Configure .....	93
Read Function.....	115
Read Lines Function.....	121
Read Notes Function .....	123
Read Options Function.....	119
Read Plot Function .....	117
References .....	231
Regression.....	153
Regularize Function .....	211
Remove Frequency Drift .....	151

Remove Gaps .....	141
Remove Outliers .....	149
Replace Function.....	133
Run Function.....	193
Run Lines .....	201
Run Notes .....	205
Run Options .....	199
Run Plot Function.....	197

## S

Save Function .....	103
Scale Function.....	143
Second Differences .....	152
Sigma Function.....	155
Sigma-Tau Diagrams .....	79
Special Characters .....	251
Stable32.....	1
Stable32 Functions.....	249
Statistics Function.....	147
Status Bar.....	9
Support.....	2

## T

Tau, Setting.....	10, 19, 21, 101
Test Suite .....	237
Thêo1 .....	51
ThêoBR.....	52
ThêoH .....	52
Three-Point Fit.....	153
TIE rms .....	57
Time Domain Stability.....	31
Time Variance .....	39
Timetags.....	19, 185
Timetags Function .....	185
Timing Solutions Corporation (TSC).....	15, 91, 107
TKF Plot Files.....	17, 225, 269
Toolbar.....	5, 271
Total Variance (Total Allan Variance).....	31, 41
Total Hadamard Variance.....	49
Total Modified Variance .....	43

## U

User Interface .....	5
----------------------	---

## V

Validation .....	245
------------------	-----

Variations .....78

## Notes

### W

Web Site ..... 2

WinBatch ..... 23, 270

Windows Metafiles (wmf) ..... 269

### X

x (Phase Data) ..... 5

### Y

y (Frequency Data) ..... 5

### Z

Zero Frequency ..... 12, 136

Zero Mean ..... 137

### 0-9

3-Cornered Hat Function ..... 219

5110Comm ..... 273