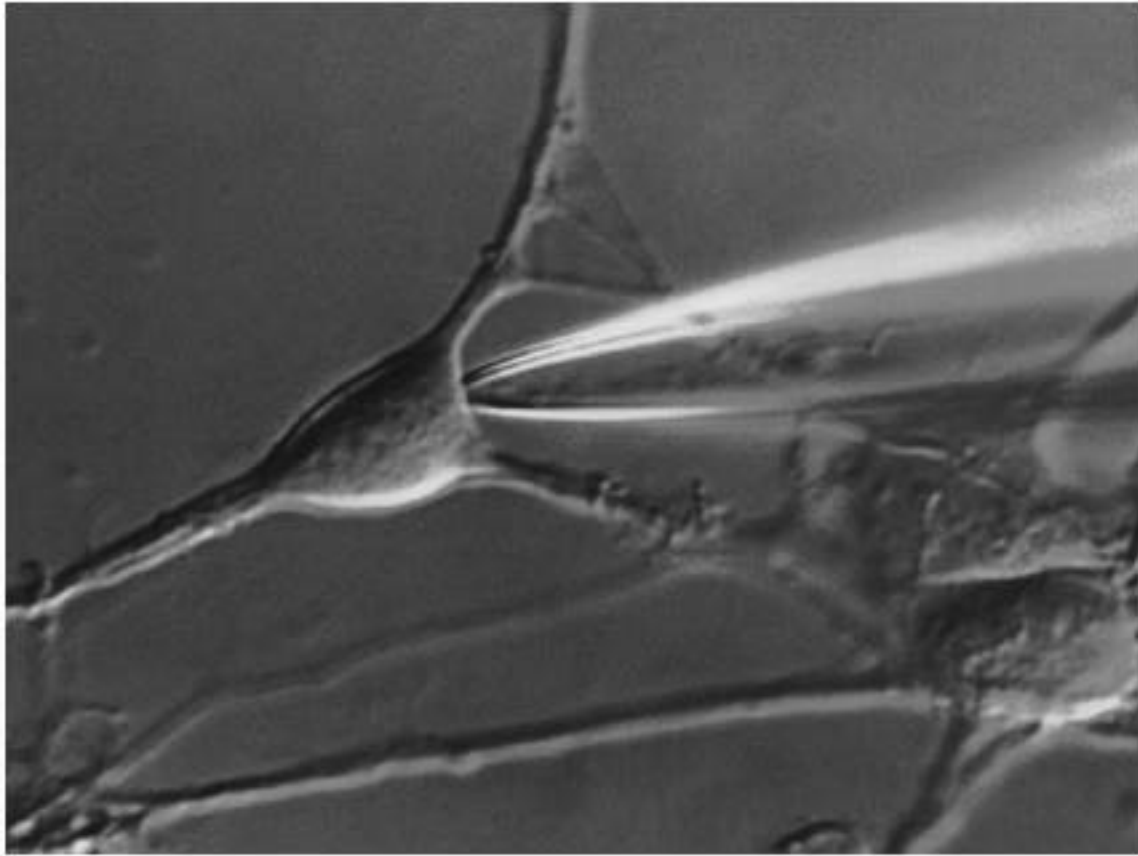


8.50 / 1.20 PulseTools Manual



HEKA Elektronik • Dr. Schulze GmbH
Wiesenstraße 71 • D-67466 Lambrecht • Germany
Tel: +49 (0) 6325 9553 0 • Fax: +49 (0) 6325 9553 50
Web Site: <http://www.heka.com>
E-mail: sales@heka.com • support@heka.com



Table of Contents

1. Introduction	4
Scope	4
Version Number.....	4
Starting PulseTools	5
Replay Windows.....	6
2. Menu	7
PulseTools Menu	8
Tools Menu	9
3. Data Edit Tools	10
Copy Functions	10
Copy Settings Dialog.....	10
Example.....	14
Example.....	15
Edit Raw Data	17
Segment Continuous Data	18
Segment Continuous Settings Dialog.....	18
Example.....	19
4. Conditional Averaging	20
Example.....	20
Conditional Averaging Dialog.....	21
Leak Correction	23
5. Trace Fit	24
Example.....	24
Trace Fit Dialog	24

6. Noise Analysis	26
--------------------------	-----------

Non-Stationary Noise Analysis	26
--------------------------------------	-----------

Example..... 26

Noise Analysis Dialog..... 27

Variance Fit	30
---------------------	-----------

Variance Fit 30

Example..... 32

Variance Fit Dialog..... 33

Power Spectrum	35
-----------------------	-----------

Example..... 35

Power Spectrum Dialog..... 36

7. Amplitude Histogram	37
-------------------------------	-----------

Example..... 37

Amplitude Histogram Dialog..... 38

8. Event Detection	40
---------------------------	-----------

Event Detection 40

Appendix I: Examples	42
-----------------------------	-----------

Nonstationary Noise Analysis & Variance Fit	42
--	-----------

Introduction..... 42

Pulse Protocol for Data Acquisition 42

Example..... 45

1. Introduction

Scope

The *PulseTools* program provides procedures for the handling and analysis of data generated by *Pulse*. It was developed by Rüdiger Steffan and Stefan H. Heinemann who would like to thank Hubert Affolter, Michael Pusch, and Franco Conti for help with earlier versions and stimulating input.

PulseTools was developed in the *PowerMod* environment and makes use of the modular concept of *Pulse*. Therefore, most of the features of *Pulse* are also available in *PulseTools*. The basic novel feature of *PulseTools*, however, is that two data files can be opened at the same time.

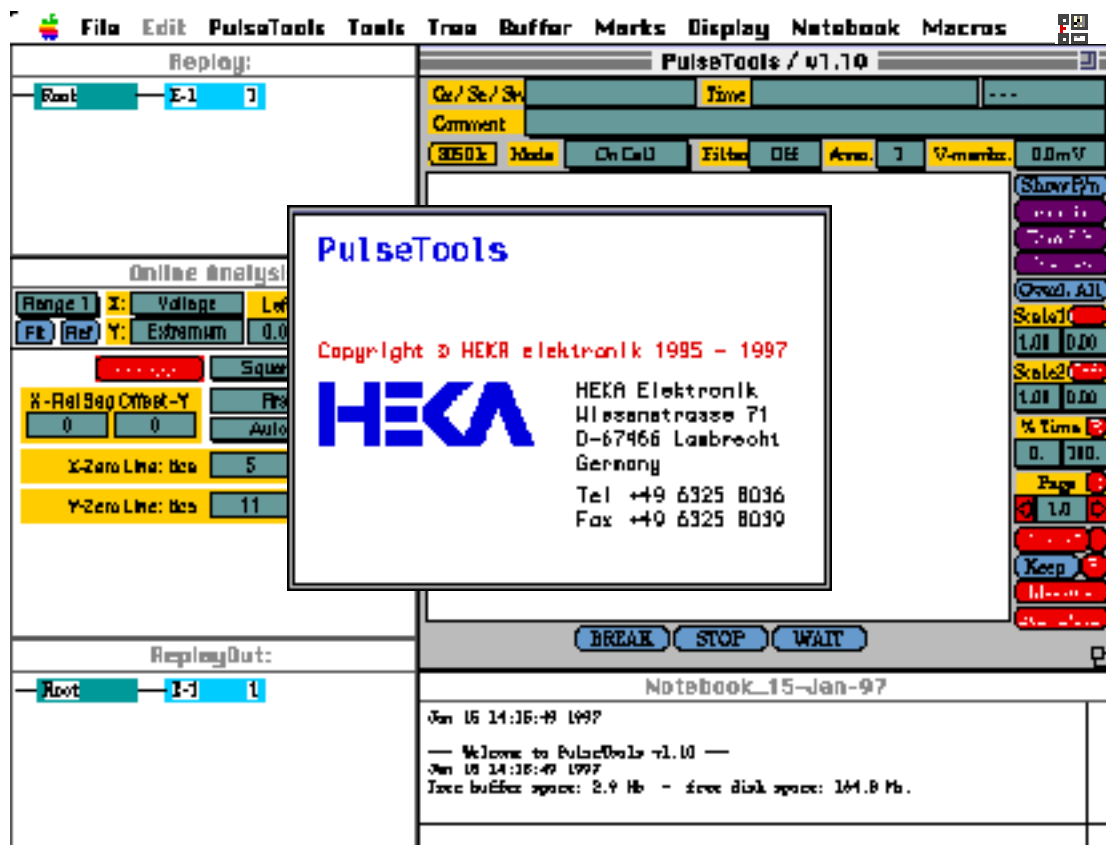


Version Number

This manual is referring to *PulseTools*-related modules version 1.10 (see *PulseTools Manual*, "Tools Menu" and *Read Me - PulseTools*). The version number of the *PulseTools* program is referring to the *Pulse* environment where the digit preceding the period denotes the currently supported *Pulse* data format, i.e. this version of *PulseTools* can handle data files created with *Pulse+PulseFit* versions 8.xx.

Starting PulseTools

The following picture shows the computer screen while launching *PulseTools*. The basic dialogs are *Oscilloscope*, *Online Analysis*, *Notebook*, *Replay*, and *ReplayOut*. As a very helpful feature, the free computer memory space is displayed in the item, which is known as the button *Store* in *Pulse*. It is recommended to allow *PulseTools* the allocation of at least 4 MB memory space (see *Read Me - PulseTools- "Hardware Requirements"*). The memory space for allocation has to be increased according to the size of the data file to be processed.

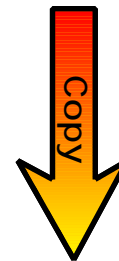
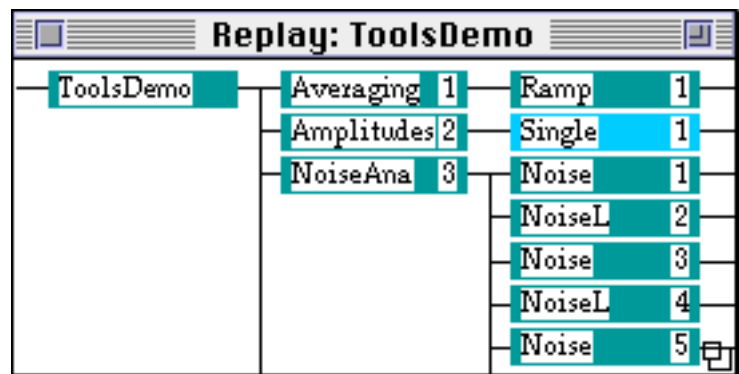


Replay Windows

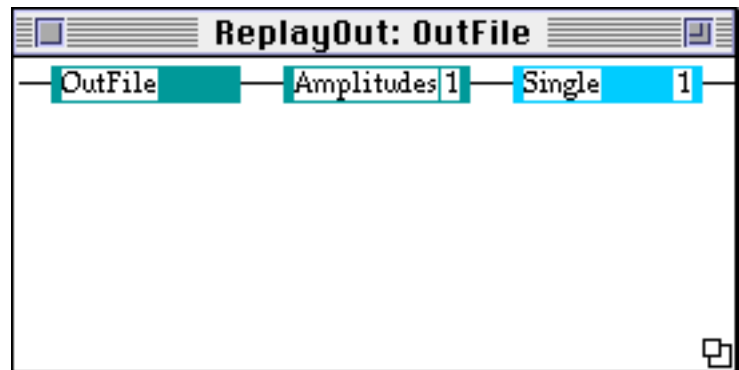
Data files are opened with the corresponding items in the *File* menu, as described in the *Pulse Manual* as well as in the following chapter of this manual.

An input data file can only be opened *Read-Only*, whereas output data can be modified and stored in an output file. For this purpose, a variety of operations are available which create output from the input. One of the standard features, for example, is the copying of tree branches from the input to the output. In addition, a series of application tools are offered that serve specific purposes, like noise analysis, conditional averaging, or compilation of amplitude histograms.

Input data tree



Input data tree



2. Menus

The menu of *PulseTools* contains the new entries *PulseTools* and *Tools*. All other menus provide the same features as the corresponding entries in *Pulse* (see *Pulse Manual* “Menus”). Tree and file operations are always applied to the *Replay* window which is currently active.

For example, in order to open an input file, the *Replay* window for the input tree data must be active when a *File* menu item is selected. If tree operations are only allowed for a specific tree, the corresponding menu items will be disabled when the wrong *Replay* window is active. The example shows the menu list *File* when the input *Replay* window is active; functions which require write permission are disabled.

In the following, all menu items of the new menus are described briefly. The next chapters will then treat all of the corresponding dialog windows in more detail.

File	
New...	⌘N
Open Read Only...	⌘O
Open Modify...	⌘M
Update File	⌘U
Close	
File Status	⌘I
Convert To Native	
Page Setup...	
Page Margins...	
Print Notebook...	⌘P
Quit	⌘Q

PulseTools Menu

New Group: Generates a new group when a file is opened without write protection. If the *Replay* window for output data is active, a new group is appended to the output tree. In all other cases the *New Group* item is disabled. After the addition of a new group, an automatic update of the corresponding data file is performed.

New Experiment: Generates a new group and increments the experiment number.

Replay / ReplayOut: Switches to the *Replay* window for input data or to the *Replay* window for output data, respectively. Both commands can be executed by the same menu key. With **F13** a switch to the *Replay* window for input data is performed. However, if the *Replay* window for input data is already active, then the *Replay* window for output data becomes active; i.e., **F13** toggles between the input and output *Replay* windows.

Buffer Allocation...: Specifies the memory that can be allocated by *PulseTools*. This setting is stored in the "Preferences" folder ("PulseTools Settings" file).

All remaining items correspond to the entries in the *Pulse* menu and are therefore not described here (see *Pulse Manual, Chapter 3 - Menus*).

PulseTools	
New Group	(^N)
New Experiment	(^E)
<hr/>	
Oscilloscope	F15
Replay	F13
ReplayOut	F13
Pulse Generator	F12
Configuration	F11
Parameters	F9
Online Analysis	F8
Notebook	F5
Solution Base	
<hr/>	
Front Dialog	▶
PulseTools Help	⌘?
Show Keys	
<hr/>	
Buffer Allocation ...	

Tools Menu

The *Tools* menu contains tools for the handling and specific analyses of *Pulse* data. The analysis tools are controlled by dialog windows as described below. The user interface is designed such that only one operation can be performed at the same time. Therefore, once an analysis dialog is opened, all other analysis menu items are disabled. The dialog window of the selected application then becomes the active analysis window which is internally treated like all other dialogs (see *Pulse Manual*).

About Tools...: Shows the version number of the *PulseTools*-related analysis modules.

Copy: Shows a submenu for selection of a specific copy function (see below):

- Plain Copy
- Conditional Copy
- Settings...

Edit Raw Data: Edit raw data sweeps.

Continuous: This submenu provides options for segmenting continuous data sweeps (see below):

- Segment Continuous Sweep
- Settings...

Conditional Averaging: Switch to *Conditional Averaging* window.

Trace Fit: Switch to *Trace Fit* window.

Noise Analysis: Shows a submenu of the following analyses:

- **Nonstationary Noise:** Switch to *Nonstationary Noise Analysis* window.
- **Variance Fit:** Switch to *Variance Fit* window.
- **Power Spectrum:** Switch to *Power Spectrum* window.

Amplitude Histogram: Switch to *Amplitude Histogram* window.



3. Data Edit Tools

Copy Functions

Plain Copy: Copies selected tree branches from the input tree to the output tree. "Selected" means either marked (see *Marks* menu) or highlighted in the input tree, where priority is given to marked branches. I.e., highlighted branches are only considered when no branch is marked.

If a single branch is copied, a new parent branch will be created in the output tree. However, if the input parent branch was not changed since the last copy operation, the copied branch will be appended to the same parent in the output.

Conditional Copy: Copies selected tree branches from the input tree to the output tree, according to parameters specified in the *Copy Settings* dialog.

Settings...: In this dialog the settings for the *Conditional Copy* feature are specified.

Copy Settings Dialog

Source: Data that are used for copying from input to output.

- **Raw Data:** the stored raw data.
- **Display Data:** data as displayed, i.e. after digital filtering or before leak and zero subtraction.

Note: All combinations of the "Copy Settings" are allowed. As an example, the "Copy Settings" dialog is shown for a multiple choice of settings.

1st Trace: With this selection it is specified which trace is going to be the new 1st trace in the output file. Of course, the setting "1st trace" is commonly used, but the other options provide the possibilities to create new output sweeps in which P/n, 2nd trace, or the sweep buffer are the main traces.

- **1st Trace:** 1st trace of the input becomes 1st trace of the output (default).
- **P/n Leak:** The leak traces associated to the 1st trace (P/n trace).
- **2nd Trace:** The second data trace is treated as 1st trace in the output.
- **Buffer:** The sweep buffer is considered as a new 1st trace in the output file. Note that the sweep buffer does not contain any tree information; these are taken from the selected branch.

COPY SETTINGS

<i>Source</i>	<i>1st Trace</i>	<i>Output</i>
<input checked="" type="radio"/> Raw Data	<input checked="" type="radio"/> 1st Trace	1st Trace
<input type="radio"/> Display Data	<input type="radio"/> P/n Leak	<input checked="" type="checkbox"/> P/n Leak
	<input type="radio"/> 2nd Trace	<input checked="" type="checkbox"/> 2nd Trace
	<input type="radio"/> Buffer	

Scaling

<input type="radio"/> No Scaling	<input checked="" type="radio"/> Ask Scaling Before Sweep
<input checked="" type="radio"/> Fixed Scaling	<input type="radio"/> Ask Scaling Before Series
<input type="radio"/> Use Online Results	
<input type="radio"/> Leak Correction	

Miscellaneous

<input type="checkbox"/> Subtract Buffer	Delete Segments
<input type="checkbox"/> Subtract Reference	<input type="text" value="1,3"/>

OK **Cancel**

Output: With this selection it is specified which traces are output additionally.

- **1st Trace:** There is always a 1st trace - the one which was specified under "1st Trace" (see above).
- **P/n Leak:** Copy the P/n leak trace. Note that this only makes sense if one selected the 1st traces as the actual new first trace.
- **2nd Trace:** Copy the second data trace.

Note: The displayed dialog shows the default situation, i.e. the 1st traces are copied as 1st traces; p/n and 2nd traces are also copied whenever they are present.

Miscellaneous: Data subtractions and elimination of data segments.

Subtract Buffer: Subtracts the sweep buffer before storing the input data to the output data file. For this purpose, any selected sweep trace can be loaded in the

sweep buffer by using the corresponding *Buffer* commands (see *Pulse Manual*, Chapter 3 - *Menus*). This operation can be very helpful for carrying out leak correction or for investigating the difference of records.

Note: A trace which is stored in the "Sweep Buffer" does not contain any sweep or series information. Scaled buffer points are subtracted from scaled data according to the number of data points of the sweep to be copied, regardless of whether there are more or fewer data points stored in the "Sweep Buffer".

Subtract Reference: (in preparation)

Delete Segments: Deletes one or more data segments from the sweep to copy. This feature can be used to remove long pre-pulse segments, for example. In the pulse generator file (*.pgf) the segment durations of the corresponding stimulation entries are set to zero; the corresponding raw data are removed irreversibly from the output sweep before storing. Different segment numbers must be separated by commas as shown in the example. If no segment shall be deleted, the string entry of the *Delete Segments* item must be either empty or *None*.

Note: If the "Relevant Segment" is deleted, the first segment is set to the new "Relevant Segment". In the case of incremented time intervals, the maximum number of data points of the corresponding output series is updated after segment deletion.

Scaling: Scaling or leak-correction of data before copying:

- **No Scaling:** Copy data as they are.
- **Fixed Scaling:** Copy data using a manually specified scaling factor and offset. These scaling factors can be entered either before each sweep or before each series.
- **Use Online Results:** Copy data using scaling factors based on the online analysis results.
- **Leak Correction:** Copy data after subtraction of the automatically scaled sweep buffer.

Thus, the copied data can be scaled by a factor and shifted by an offset value ("Fixed Scaling"). Alternatively, the data can be scaled according to the *Y-result* of the online analysis ("Use Online Results"). The values for scaling can be entered either before each sweep, or before each series which is to be copied. This option can be set with the respective buttons "Ask Scaling Before Sweep" and "Ask Scaling Before Series".

Scale Sweep (1_1_2):

Factor	<input type="text" value="1.00E+00"/>
Offset	<input type="text" value="0.00E+00"/>

Fixed Scaling: While copying, a scaling factor and an offset value can be entered either before each sweep, or before each series to be copied. In the example, the scaling dialog is shown when "Ask Scaling Before Sweep" is selected. The *Retrieve* button can be used to reset the entered numbers to the previous values. With *Do All* the options "Scaling Before Sweep" or "Scaling Before Series" are ignored and scaling is performed according to the last entered values for the entire selected tree branch.

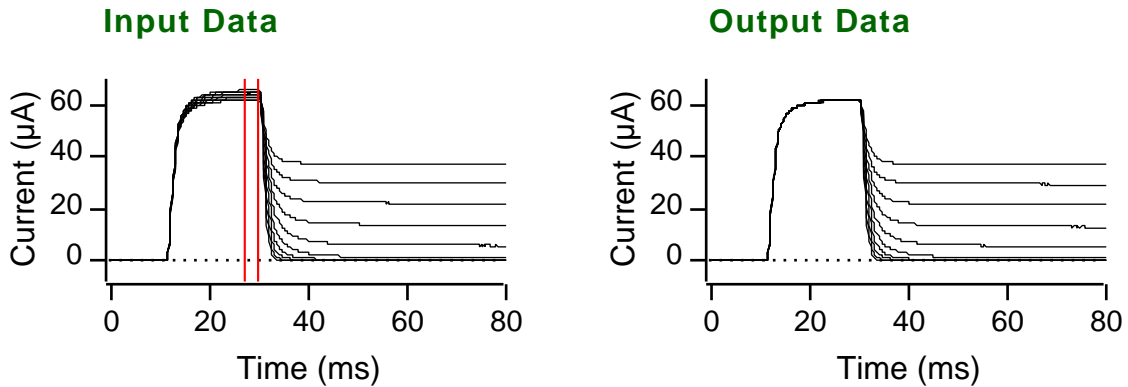
Use Online Results: The scaling factor is calculated automatically according to the displayed equation. In the example, the scaling dialog is shown when "Ask Scaling Before Series" is selected.

Scale Series (1_1) for Copy:

Factor*OnlineYResult =

Example

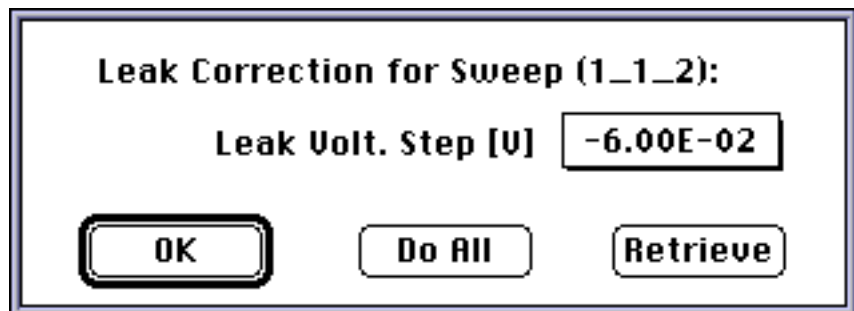
Raw data was copied with the setting "Use Online Results", where the Y-result of the online analysis is the mean current inside the indicated cursor range. All sweeps were scaled to the value 61.91 μA , which is the Y-result of the first sweep.



Leak Correction: This feature can be used in order to perform a simple automatic leak correction. Traces to be considered as control leak sweeps are accumulated in the sweep buffer. Before

copying a selected target, the program inquires formation necessary for the correct scaling of the leak sweep before subtraction. For this purpose the "Leak Volt. Step" (see above) has to be specified. This is the equivalent potentials step for the sweep buffer from the segment preceding the *Relevant Y Segment* to the *Relevant Y Segment*.

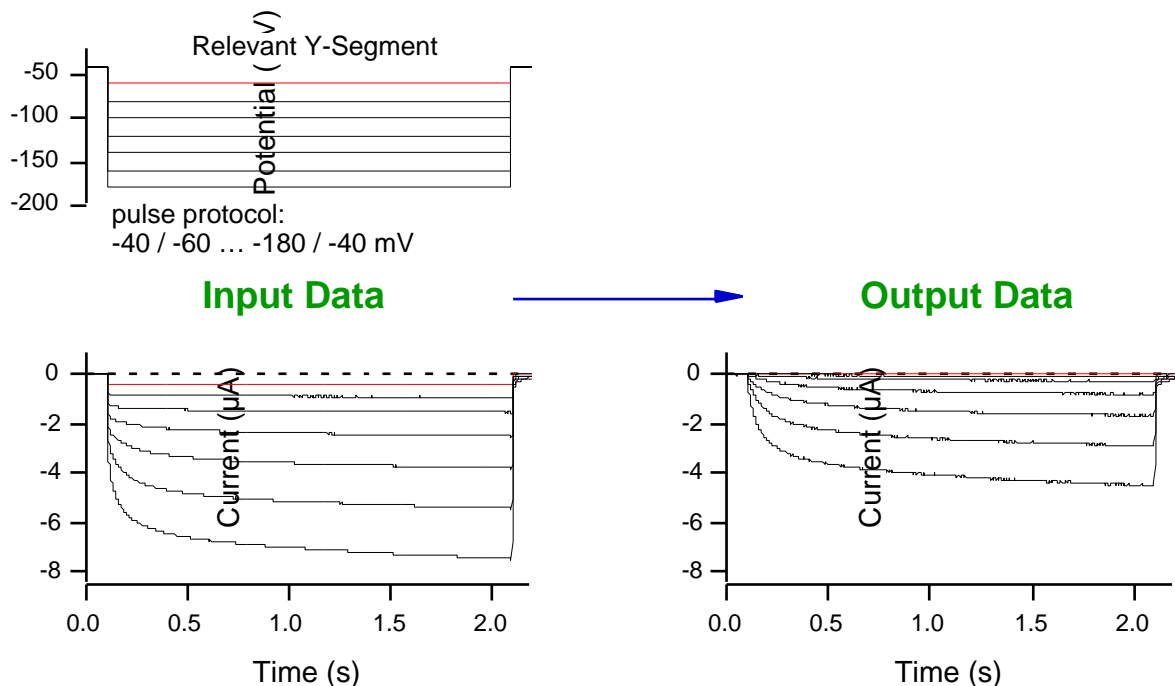
Note: This operation only gives satisfactory results when several preconditions are considered: 1) Do the leak subtraction only when ZeroSubtraction is turned on in the Oscilloscope dialog. 2) The leak subtraction only works if the "Leak Volt. Step" represents a real scaling factor for the entire pulse template, i.e. this step size should be the only one used in the protocol (symmetrical pulses). 3) Make sure that the selected leak sweeps which are loaded into the buffer can be used as such, i.e. they should not show a current signal to be measured.



Example

The following example illustrates the leak correction of hyperpolarization-activated ionic currents elicited with voltage steps from a holding potential of -40 mV to -60 through -180 mV. We assume that during the pulse to -60 mV only linear leak current is measured. The shown output data were generated by executing the following commands:

- 1) Turn on "Leak Correction" in *Copy Settings*.
- 2) Clear *Sweep Buffer* (<C>)
- 3) Select sweep 1 in the input tree and load it into the *Sweep Buffer* (<+>). Thus, we take the sweep corresponding to a step from -40 mV to -60 mV as leak sweep.
- 4) Select series to be copied after leak correction.
- 5) Copy Conditional: A dialog will come up asking for a "Leak Volt. Step". Enter -20E-3 as the selected leak trace in the *Sweep Buffer* corresponds to a potential step from -40 mV to -60 mV.







Note: If there are more than one sweep to be considered as leak sweeps (what must be done in order to reduce the noise), they have to be accumulated in the sweep buffer.

If we assume that we would have taken Sweep2 (step to -80 mV) in addition to Sweep1 (step to -60 mV), we would have added Sweep2 to the sweep buffer (see procedure 3). The "Leak Volt. Step" now would have been -60E-3, i.e. -20 mV from Sweep1 and -40 mV from Sweep2.

Edit Raw Data

If the *Replay* window for output data is active, the raw data of a selected sweep can be modified in a table containing the individual sample points. If the replay window for input data is active, the raw data points can only be displayed since this file is *Read Only*.

A sweep can contain up to three traces: the first trace, the (P/n) trace for the first data sweep (leak trace), and the second trace, where the latter two traces are optional. The first shown data point corresponds to the data point of the left cursor setting in the *Oscilloscope* dialog. Data point values are displayed in a table, where the first data point can be adjusted by using the following keys:

-  : Move forward or backward by one data point.
-  : Increment or decrement the lower data point in steps of 15 data points.

The cursors of the *Oscilloscope* window are updated for the selected range of data points. If the replay window for output data is active, raw data can be edited point-by-point in order to correct artifacts manually, for example. In addition, an entire column can be multiplied by a scaling factor, provided at the top of a column; the first trace and the P/n trace have identical scaling factors. As an example, a sweep is displayed which contains all of the three traces. The second trace is the voltage trace as recorded during a two-electrode voltage clamp experiment. After 100 data points, the command voltage was stepped from -120 mV to +60 mV; at the potential electrode, a rising phase of about 3 sample points was measured.

Edit Data			
#	Data	Leak	2nd
	1.00E+00		1.00E+00
97	-3.05E-07	-4.69E-09	-1.22E-01
98	-3.03E-07	-1.25E-08	-1.22E-01
99	-2.94E-07	-1.56E-08	-1.22E-01
100	-3.05E-07	2.03E-08	-3.34E-02
101	-3.69E-07	2.85E-06	4.30E-02
102	-1.83E-07	4.19E-07	5.82E-02
103	9.37E-09	-1.25E-06	6.03E-02
104	4.69E-07	-6.50E-07	6.03E-02
105	1.27E-06	5.31E-08	6.02E-02
106	2.37E-06	4.22E-07	6.01E-02
107	3.71E-06	5.61E-07	6.01E-02
108	5.17E-06	6.00E-07	6.00E-02
109	6.67E-06	6.11E-07	6.00E-02
110	8.18E-06	6.03E-07	6.00E-02
111	9.63E-06	5.98E-07	6.00E-02
112	1.10E-05	5.98E-07	6.04E-02

Segment Continuous Data

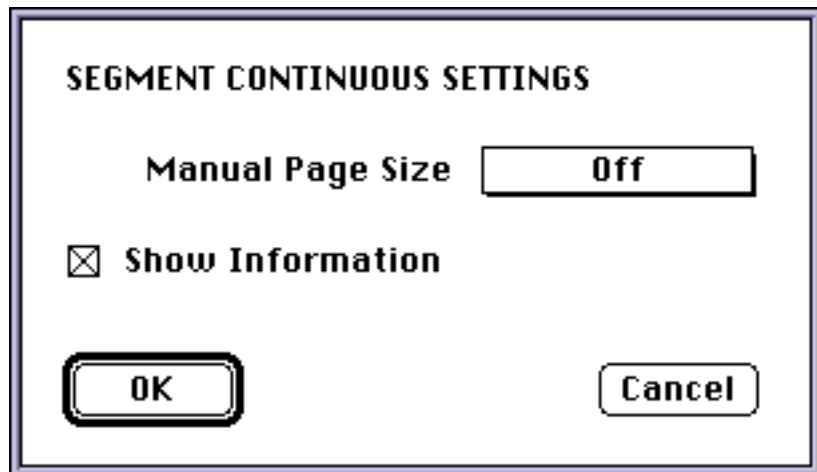
Most analysis features (see below) are restricted to non-continuous data. The option "Segment Continuous" can be used to generate a set of non-continuous data Sweeps of one continuous data Sweep. As a new time base the total duration of all segments which precede the continuous segment are taken (this corresponds to a page on the oscilloscope). Alternatively, a block size (e.g. 16kB) can be defined. The segmenting works sweep-by-sweep. From each input sweep to be segmented a new group with two series is generated. The first one contains a sweep of the non-continuous part only, the second series contains all sweeps representing the segments of the continuous segment. Note that the size of the last output sweep can be shorter than the page size.

Note: The second trace of a continuous Sweep is segmented as well.

Segment Continuous Settings Dialog

Manual Page Size: With this option a page size can be selected manually (maximum is 16k). If *Manual Page Size* is "off", the page size based on the non-continuous segments in the *Pulse Generator* file are used.

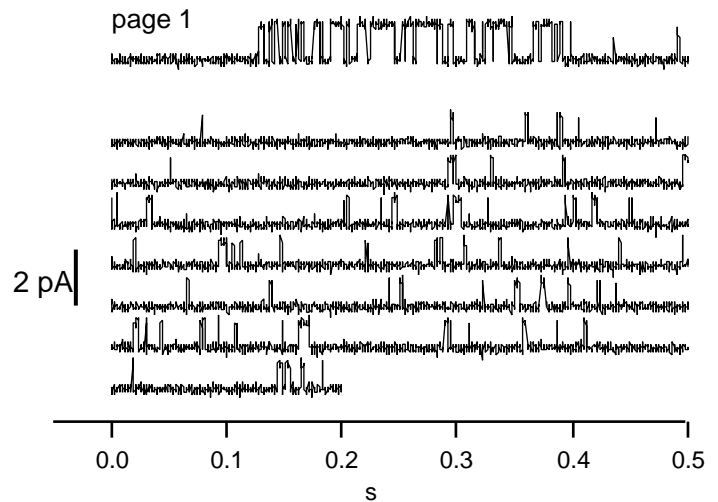
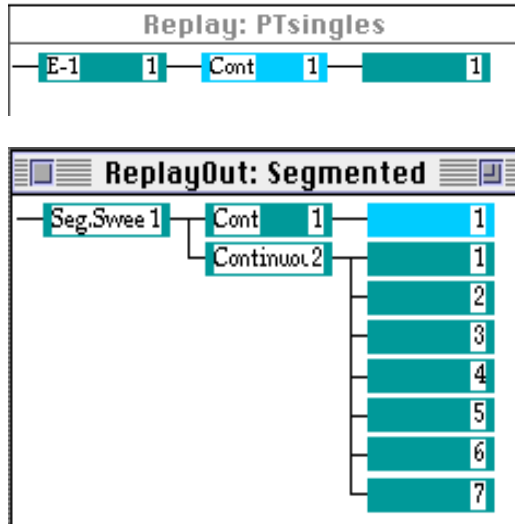
Show Information: As the segmenting of data can take quite a lot of time, information about the progress of the segmentation can be shown.



Note: Segment Continuous Sweep is also performed on all marked input sweeps. In this case, the highlighted selection in the input tree is ignored.

Example

A data sweep, based on four segments, with the last one being a *Continuous Segment* is segmented into one series with one sweep ("Cont"), and one series with 7 Sweeps ("Continuous") which represent the segmented continuous part of the input sweep.

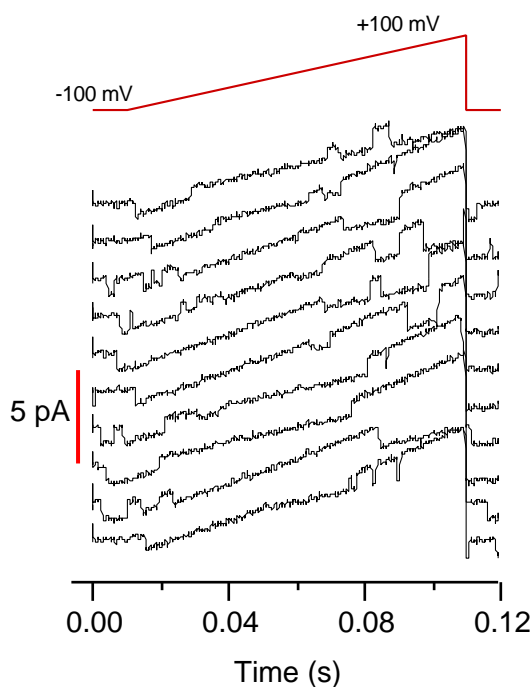


4. Conditional Averaging

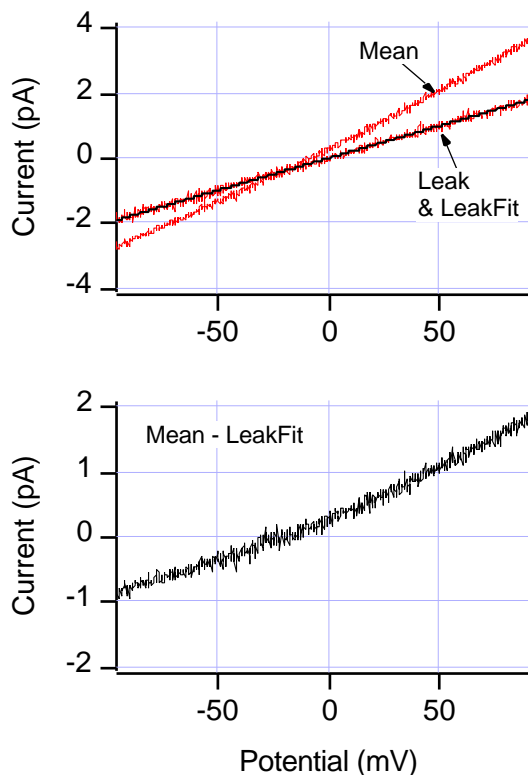
This function allows to compile averages of data sweeps, selected either in the input or in the output tree. The averages as well as the traces that contain the number of selections for each data point (*Counts*) can be stored in the output tree. When blank sections in single-channel traces are compiled, further analysis allows to perform leak correction by using the subtract feature of the *Copy* function (see *Chapter 3 - Data Edit Tools*). Another major application is to compile single-channel iV curves based on single-channel recordings as responses to ramp voltage stimulation. Below, the analysis window is shown as it could appear during such an application. The averaged currents are displayed on-line as *Present Results* during analysis using colors which can be specified in the configuration dialog. (see *Pulse Manual, Chapter 7 - Configuration*).

Example

Raw Data: Single-channel ramps



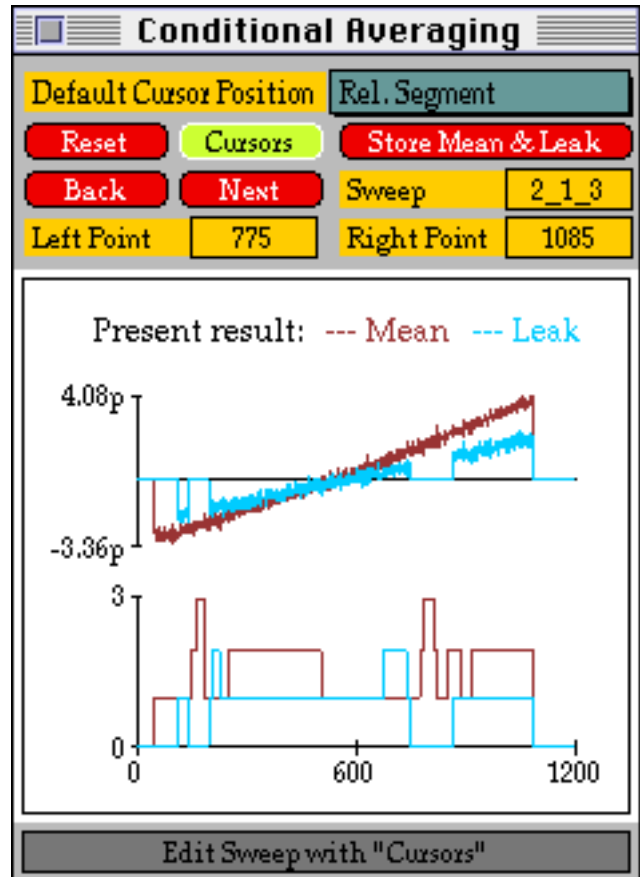
Analysis: Single-channel iV



Conditional Averaging Dialog

When the *Conditional Averaging* dialog is called, the sweep which is currently selected in the *Replay* window of the input tree is loaded for analysis. If the last selection has not been a sweep, all analysis functions are disabled and the user is prompted to select a sweep first. The features for averaging are:

Cursors: Switch to the *Oscilloscope* dialog window and activate cursor lines. These cursor lines are not identical with the cursors of the Oscilloscope but they are controlled in the same way by dragging with the mouse or by using the arrow keys on the keyboard. The data of a cursor-selected time region can be added either to the mean trace or to the leak trace by typing or , respectively. If a region shall be considered multiple times, the corresponding number can be used instead of . (e.g., if it is assumed that two channels are open in the region, the region can be added with). For averaging, the summarized traces are divided by the number of selections (*Counts*) for each data point. Graphs show the averaged traces as well as the counts during analysis (*Present Result*).



Default Cursor Type: When a new sweep record has been selected, the cursors are set to this default position. The default positions are:

- **MinMax** - Data point 0 and maximum data point.
- **Rel.Segment** - Start and end of the relevant pulse segment.
- **LastPosition** - Cursor positions of the previous selection.

Reset: Set all values of the mean and leak trace to zero.

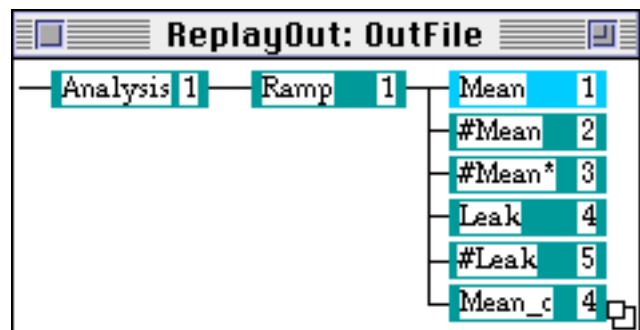
Back / Next: Select the previous or next sweep in the *Replay* window of the input tree. These buttons can be used instead of switching to the *Replay* window and selecting the sweep in the tree.

Since all *Oscilloscope* items are disabled when the cursor lines are active, selections can only be performed by using the keyboard. In addition, several helpful functions are available via “hot keys” during averaging which are summarized in the table below.

Key	Function
0	Add the data of the selected time interval to the <i>Leak</i> trace.
1 2 3 ...	Add the data of the selected time interval to the <i>Mean</i> trace. The number indicates how often the selected data is added to the <i>Mean</i> .
B	Select and display the previous tree entry.
G	Activate the <i>Gain 1</i> item of the <i>Oscilloscope</i> dialog.
O	Activate the <i>Offset 1</i> item of the <i>Oscilloscope</i> dialog.
H	Help text menu.
N	Select and display the next tree entry.
P	Execute “R” button of the <i>Page</i> items of the <i>Oscilloscope</i> dialog, i.e., reset all display settings of the <i>Oscilloscope</i> (<i>Gain</i> , <i>Time</i> , <i>Page</i>).
Q	Exit <i>Cursor</i> mode.
R	Execute “R” button of the <i>Time</i> items of the <i>Oscilloscope</i> dialog, i.e., reset <i>Time</i> settings to 0 and 100 %.
T	Activate left <i>Time</i> item of the <i>Oscilloscope</i> dialog.
Shift T	Activate right <i>Time</i> item of the <i>Oscilloscope</i> dialog.
Z	Undo last accumulation, i.e., subtract the selected data.

Store Mean / Leak: Results are stored in the output. The output sweeps are:

- **Mean:** Accumulation of *Mean* trace.
- **#Mean:** Number of “channel openings” considered for each point of the *Mean* trace (*Counts*).
- **#Mean*:** Number of additions made to *Mean*. The distinction between *#Mean* and *#Mean** is important when the *Leak* is to be subtracted from the *Mean*. One has to consider that an accumulation of a double-channel opening contributes two channel current amplitudes but only one leak amplitude (see below).
- **Leak:** Accumulation of *Leak* trace.



- **#Leak:** Number of averages considered for each point of the *Leak* trace (*Counts*).
- **Mean_c** - *Mean* trace corrected according to the description below.

Leak Correction

The accumulation of regions with multiple-channel opening leads to a wrong average of the *Leak* amplitudes which are contained in the *Mean*. In order to correct a *Mean* for an averaged *Leak*, the *Mean* is corrected according to the following equation:

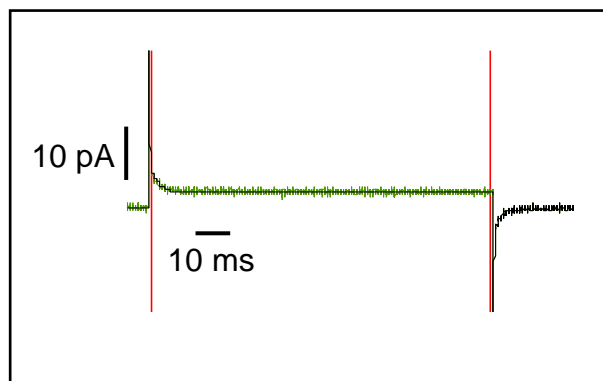
$$Mean_c = Mean - \frac{(\#Mean^*) Leak}{\#Mean}$$

5. Trace Fit

Polynomials (exponential functions in preparation) can be automatically or manually fitted to the data of selected sweeps in order to smooth blank records, for example. For this purpose, sweeps to be fit can be selected either in the input or in the output tree. The fit trace as a function of time can then be stored as a sweep branch in the output tree for further analysis such as for leak and capacitance correction of single-channel traces (see *Chapter 4 - Conditional Averaging*).

Example

Smoothing: Raw data trace of a blank single-channel recording acquired with *Pulse*. For the time range specified by cursor settings (vertical lines) a polynomial fit has been performed and superimposed.



Trace Fit Dialog

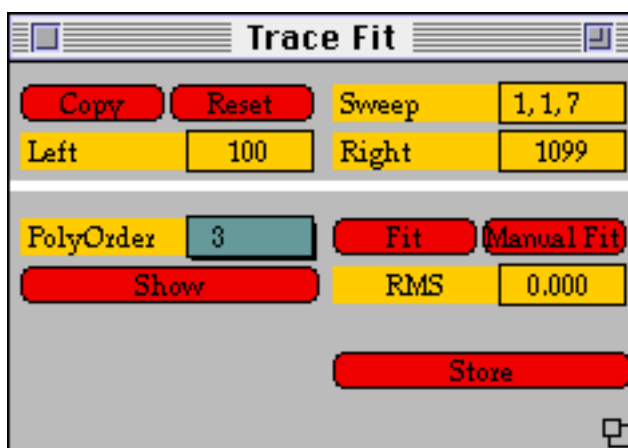
Copy: Copies the data of the selected sweep to the fit trace.

Reset: Sets fit trace to zero.

Sweep: Number of the group, the series, and the selected sweep, respectively.

Left / Right: Range of data points used for fit, as selected with the *Oscilloscope* cursors.

PolyOrder: Order of the polynomial function that is used for fit.



Fit / Manual Fit: Executes an automatic fit by using the least squares method, or calls a dialog for adjusting the fit parameters manually (see dialog below).

Show: Superimposes the fit trace on the selected sweep trace in the *Oscilloscope* window. The colors of the traces are specified in the *Configuration* dialog (see *Pulse Manual, Chapter 7 - Configuration*).

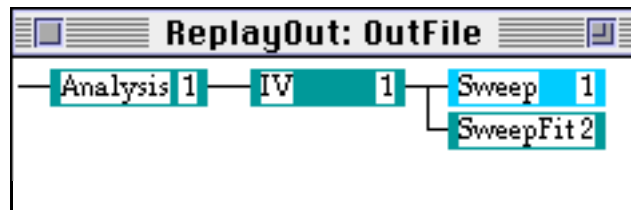
RMS: Root of the mean squared deviation between the values of data and fit points.

Store: Stores the fit trace together with the sweep that was used for fit in the output tree. Series information is copied from the input. If no new group has been created, the new series is appended to the last group of the output tree.

$I(t) = \sum_{i=0}^n a_i t^i$

AutoFit

PolyOrder n	5
a 0	5.91p
a 1	-415.49p
a 2	20205.14p
a 3	-4.31E+05p
a 4	4.16E+06p
a 5	-1.49E+07p



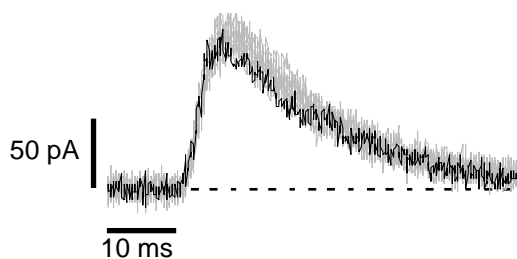
6. Noise Analysis

Non-Stationary Noise Analysis

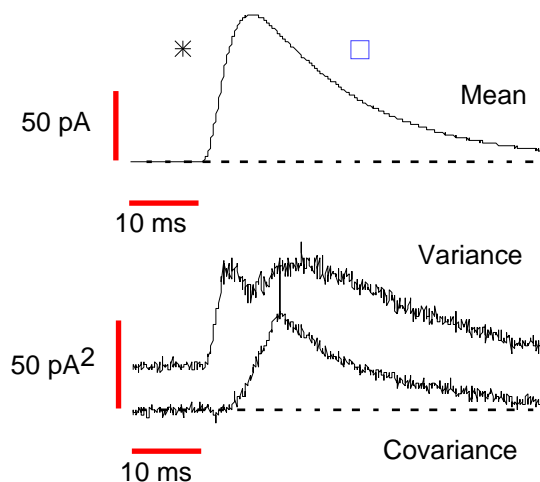
Non-stationary noise analysis is performed on data files as described by Heinemann & Conti (1992). Per marked data range a mean and leak current trace, an ensemble noise trace, and an arbitrary number of covariance traces at pre-selected times are compiled and stored as a series of the output tree.

Example

Input data: The figure on the right shows simulated ionic currents based on a kinetic Hodgkin & Huxley model including sequential inactivation. Each one of the traces is the accumulation of 100 single-channel traces (1.5 pA). Gaussian-distributed artificial baseline noise of 5 pA RMS was added and 10 of such traces superimposed in this example. The aim of non-stationary noise analysis is to determine single-channel parameters such as the single-channel current (i), the maximal open-probability (p -open) and the number of active channels (N) from such macroscopic data.



Non-stationary noise analysis: 750 traces as shown above were used as test records for non-stationary noise analysis. The results of the analysis are the mean current (upper panel), the variance, and the autocovariance as a function of time (lower panel). The variance offset of 25 pA² results from the baseline noise of 5 pA. The dialog used for this analysis is shown below indicating the analysis parameters. The results can be used for further analysis such as *Variance Fit* to obtain estimates of the single-channel parameters.



Noise Analysis Dialog

In the following, the parameters and output of the non-stationary noise analysis are described briefly. For a more detailed description of the theory as well as of the individual steps during analysis, please refer to the *Appendix*.

Test Pulse Series: Name of a series to be used as test pulse series. *PulseTools* is expecting a mixture of test pulse and leak pulse records in the marked data range. The name given in this field is used in order to identify those records to be counted as test pulses. The default entry is *Noise*. All sweeps of those series with other names will be considered to be leak pulses.

Gamma / G-max / Reversal: Estimates for the single-channel conductance (γ), the total conductance (G_{max}), and the reversal potential (E_{rev}). Default values are 20 pS, 1 nS, and 40 mV, respectively.

RMS: Specifies the maximal baseline RMS noise that is tolerated. This value is used to discard very bad records. The default value is 15 pA.

Sigma Base: Factor to specify a tolerance for the baseline variance. All records which are most likely affected by some artifactual extra noise are discarded. The default factor is 4.0.

Sigma Pulse: This factor is used to discard records, for which the difference to neighboring records exceeds certain limits, for which *Sigma Pulse* serves as a criterion. The default factor is 5.0.

Seg. Offset: If a segment other than the relevant segment is analyzed, *Seg. Offset* has to be different from zero.

Base Offset: Normally, the segment prior to the number of the relevant segment plus the number of segment offset serves as baseline segment. If another segment is

Non-Stationary Noise			
Test Pulse Series	Noise		
Gamma	20.0pS	Reversal	-100.mV
G-max	1.00nS	RMS	15.0pA
Sigma Base	4.00	Sigma Pulse	5.00
Seg. Offset	0	Base Offset	0
Record Skips	0	Var. Gain	Off
<input checked="" type="checkbox"/> Successive	<input checked="" type="checkbox"/> Show	<input type="checkbox"/> Online Cursors	
<input checked="" type="button" value="Analyse"/>	<input checked="" type="button" value="Covariance"/>	<input type="checkbox"/> Cov. Matrix	
First Sweep	0_0_0	Last Sweep	0_0_0
Test Pulses	0	Leak Pulses	0
Skipped Records (too large deviation)	0		
Discarded Couples (too big difference)	0		
Leak Reversal Potential	0.000 V		
Single-Channel Current	0.000 A		
Correlation Time	0.000 s		
Number of Couples	0		
<input type="button" value="Mark sweeps for noise analysis"/>			

used as baseline segment, e.g., for noise analysis during deactivation, *Base Offset* has to be different from zero.

Var. Gain: The gain factor for storing the mean and leak current corresponds to the gain factor of the fluctuating current traces. Var. Gain is used as the gain factor or storing the variance and autocovariance functions. If it equals zero (“*Off*”), the gain factor is calculated automatically according to the maximum value of the variance.

Var. Gain

Online Cursors: If this item is not selected, the user is prompted two times for setting a cursor range during analysis. If Online Cursors is selected, as shown here, the analysis is performed without interruption. In this case the pre-defined cursor ranges of the online analysis dialog (Cursor 1 and Cursor 2) are used.

Successive Show Online Cursors
 Analyse Covariance Cov. Matrix

Cov. Matrix: The autocovariance function is calculated for each point in time. They are used in Variance Fit for determining the error-covariance matrix of the variance as described in Steffan & Heinemann, 1997 (see below).

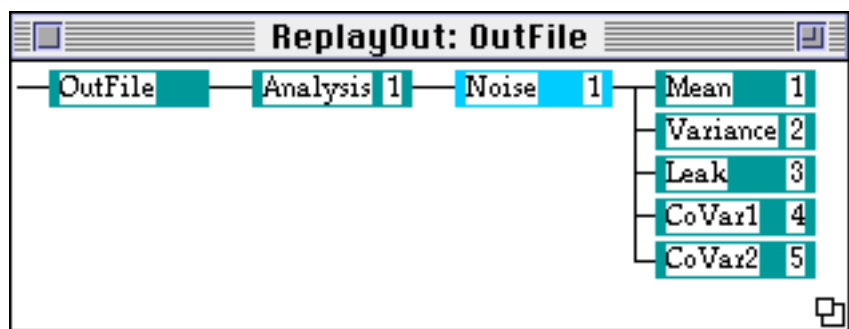
Record Skips: To eliminate the effect of initial drifts in a series of test pulses, this variable determines, how many sweeps have to be skipped at the beginning of each series. This feature is used if, for example, cumulative channel inactivation causes a reduction of peak current in the first few sweeps of a series until a “steady-state” is approached for the given sweep interval.

Note: “Records Skips” does not have any effect, if the first sweeps of a series, which are relevant to be skipped, are not marked.

First Sweep / Last Sweep: The first and the last sweep marked in the input tree. *First Sweep* and *Last Sweep* indicate the data range that is used for the non-stationary noise analysis.

Note: Since it is possible to “unmark” single sweeps in a marked series, not all sweeps that are neighbors for the noise analysis need also be neighbors in real time.

Mean & Variance: Performs non-stationary noise analysis and stores the results in the output tree. The output series created contains sweep records for the *Mean* and *Leak* current trace and the *Variance*.



Covariance: Computes the autocovariance function of the previously analyzed number of *Couples*. This operation can be repeated several times for different center points, specified by cursor settings, until the analysis dialog is exited. After switching to the dialog again, the analysis is reinitialized, so that the data has to be analyzed with *Mean & Variance* beforehand. The *Covariance* results are stored in a sweep which is appended to the series of the corresponding analysis.

Test Pulses / Leak Pulses: Number of traces of test and leak pulses which were found in the selected data range after a check for incompatible sweeps has been performed.

Skipped Records: Number of leak records which did not pass the criterion for the baseline stability and which were therefore discarded.

Discarded Couples: Difference records of neighboring sweeps which did not pass the criterion for discarding records with excessive noise.

Leak Reversal Potential: Estimate of the leak reversal potential.

Single-Channel Current: Estimate of the single-channel current based on the analysis of the test pulses.

Correlation Time: Estimate of the correlation time of the last determined *Covariance* function. The correlation time can be used in *Variance Fit* as a criterion for the estimation of the error margins of variance-current bins.

Number of Couples: *Couples* of test records which passed all the tests and were used to calculate the variance and autocovariance.

Variance Fit

Variance Fit

Auto Bounds: The cursor range of the online analysis (Cursor 1) is used for plotting the mean-variance data. If Auto Bounds is selected, the first cursor range of non-stationary noise analysis, which is stored in the data structure, is automatically used. In the previous release of *PulseTools*, this feature was coupled to the Auto Scale item.

Current Bins	Max. Current	Max. Variance
20	40.0pA	1.0E-23A2
<input checked="" type="checkbox"/> Auto Scale	<input checked="" type="checkbox"/> Auto Bounds	Show

Fit Approach: Three linear regression methods for fitting the single-channel parameters are added. They are used for calculating the uncertainties of the parameter estimates as described in Steffan & Heinemann, 1997 (see below). The fit approaches are:

Fit Approach	Iterations		Mean Sq. Error
Linear EGLS	2	0	2.21E-53A4
Settings	Fit	Show Fit	Export
	Estimate	Uncertainty	
Unitary Current	749.fA	fit	44.3fA
N Channels	50.1	fit	3.44
Baseline Noise	1.5E-27A2	fit	6.0E-26A2

- **Linear OLS:** Linear ordinary least squares.

- **Linear WLS:** Linear weighted least squares.

- **Linear EGLS:** Linear estimated generalized least squares.

- **Nonlinear WLS:** Weighted least squares using the iterative Simplex method.

Iterations: The left item is the maximum number of iterations, the right item is the iteration counter. The default value for the maximum number of iterations is 1 for Linear OLS and Linear WLS, 2 for Linear EGLS and 200 for Nonlinear WLS.

Mean Sq. Error: The mean squared difference between the mean-variance data and the fitted parabola function.

Settings: Shows a subdialog for setting the following flags:

- **Manual Fit:** See *PulseTools* manual.

- **Show Results:** The binning and fit results are written to the notebook.

- **Successive Records:** Must equal the corresponding setting for non-stationary noise analysis.

- **Predict covariance matrix:** The covariance matrix of the variance is determined from all the previously calculated autocovariance functions of the current.

If Predict Covariance Matrix is selected, the matrix is estimated using the correlation time of a single autocovariance function (see Steffan & Heinemann, 1997).

- **Estimate variance:** The variance is calculated by using the time course of the mean current as well as the pre-defined single channel parameters according to the theoretical parabola function. This feature can be used to roughly predict the uncertainties of the parameters even before performing non-stationary noise analysis (see Steffan, 1998).

VARIANCE FIT SETTINGS

- Manual Fit
- Show Results
- Successive Records
- Predict Covariance Matrix
- Estimate Variance

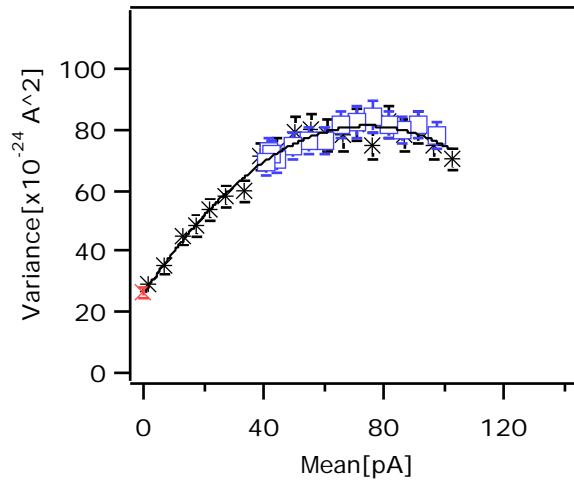
OK Cancel

Note: Steffan R, Heinemann SH (1997) Error estimates for results of nonstationary noise analysis derived with linear least squares methods. J Neurosci Meth 78: 51-63

Steffan R (1998) Modellierung und Simulation nichtstationärer Stromantworten spannungsabhängiger Ionenkanäle. Dissertation University of Stuttgart. Submitted

Example

The output series created by the non-stationary noise analysis is taken as input for this analysis. The variance can be plotted versus the mean current according to a cursor-selected time interval. The variance points are sorted into evenly-spaced current bins, in order to calculate the mean variance for each current bin. The statistical errors are superimposed as error bars. After plotting, a fit to these data can be performed with the background variance, the single-channel current, and the number of channels as fit parameters. The single-channel conductance is then calculated from the given reversal potential, the test potential, and the single-channel current assuming a linear iV-relationship.



Note: For plotting a mean-variance diagram, the first two sweeps of a selected series will be used, where the first sweep is taken as the “Mean” and the second sweep is taken as the “Variance”. Therefore, not only a “Noise” series with sweeps as specified above will be accepted.

The mean-variance diagram can be exported as an *Igor* graph, an example of which is shown above. The diagram corresponds to the previously described example of application, where the meaning of the symbols is described in the following.

Variance Fit Dialog

Show Data: Plots variance versus current (mean-variance diagram) in the *Oscilloscope* window as a temporary graph. Data points of rising current are plotted as black asterisks, while data points of falling current are blue squares. The measured background noise is shown as a red cross.

Export: The displayed graph is either stored as an “IgorText” file or is sent to the printer. The export mode can be specified in the *Tree* menu (see *Pulse Manual, Chapter 3 - Menus*).

Max. Iterations / Step Size / Iterations: Specifies the parameters for the “Simplex” fit algorithm.

Selected Series: Selected series in the data tree.

Analyzed Segment: Segment that was analyzed with the non-stationary noise analysis.

Reversal Potential / Single-Channel Conductance / Measured Baseline Noise / G-max / Correlation Time: Estimates obtained by the non-stationary noise analysis, which are now used for calculating initial values of the fit parameters and error margins of the variance-current bins. The values can be edited (e.g., error bars are avoided by setting *Correlation Time* to zero). *Get* retrieves the old values from the input file.

Current Bins: Number of current bins for sorting the variance points. The options are: Binning according to min-max of mean current (*Current Bins*), according to time delimited by the cursor settings (*Time Binning*) and *No Binning*, i.e., variance is plotted versus mean current point-by-point.

Max. Current / Max. Variance: These values set the limits for the variance-current diagram.

Variance Fit		
Selected Series	2, 1	
Analyzed Segment	2	
Reversal Potential	<input type="button" value="Get"/>	-100.mV
Single-Channel Conductance	<input type="button" value="Get"/>	20.0pS
Measured Baseline Noise	<input type="button" value="Get"/>	2.5E-23A
G-max	<input type="button" value="Get"/>	1.00nS
Correlation Time	<input type="button" value="Get"/>	2.10ms
Current Bins	Max. Current	Max. Variance
20	200.pA	2.0E-22A2
<input type="checkbox"/> Auto Scale	<input type="button" value="Show Data"/>	<input type="button" value="Export"/>
Max. Iterations	Step Size	Iterations
200	250.m	201
<input type="checkbox"/> Manual Fit	<input type="button" value="Fit"/>	<input type="button" value="Show Fit"/>
Single-Channel Current	1.49pA	fit
Number of Channels	99.9	fit
Baseline Noise	2.4E-23A	fit

Auto Scale: If this box is selected, *Max. Current* as well as *Max. Variance* are computed from the selected traces (within the cursor range). The cursor positions for specifying the range of data points for plotting the variance-current diagram are retrieved from the data file; these default cursor positions were specified during the non-stationary noise analysis. If *Auto Scale* is off, the limits of the variance-current diagram can be set manually (*Max. Current* and *Max. Variance*) and the considered data range can be selected with the *Oscilloscope* cursors.

Manual Fit: When this flag is on, a fit function will be drawn in the variance-current diagram after each editing/dragging of the fit parameters (see below).

Note: The fit function is shown for open-probabilities from “zero” to “pmax”. However, when “Manual Fit” is selected the fit function is shown for the entire range of open-probabilities, i.e., from “zero” to “one”.

Fit: Runs an automatic *Simplex* fit and shows result in the variance-current diagram.

Show Fit: Superimposes the calculated fit curve to the displayed variance-current diagram.

Single-Channel Current / Number of Channels / Background Noise: Fit parameters that are used to calculate the theoretical variance according to the following equation:

$$(\sigma^2 - \sigma_b^2) = i I - \frac{I^2}{N}$$

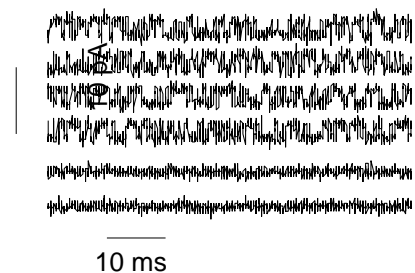
where σ^2 is the variance, σ_b^2 the baseline variance or background noise, i the single-channel current, I the mean current and N the number of active channels. In this example, all parameters were used for the fit. Alternatively, after selecting the option *hold*, individual parameters can be set manually and are then not varied during the automatic fit.

Power Spectrum

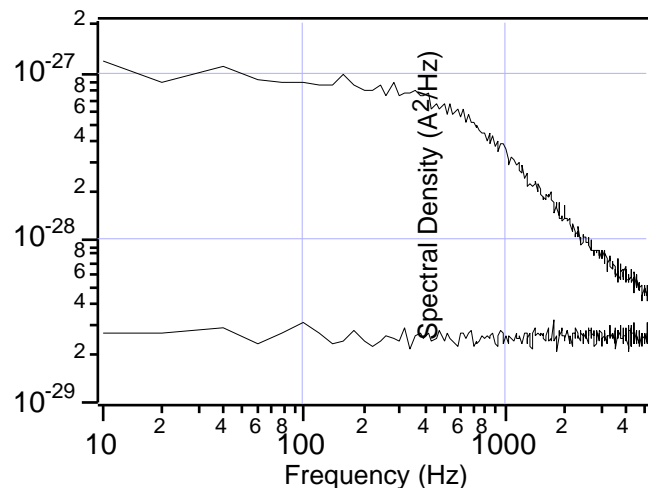
The spectral analysis dialog is used to compute and to accumulate the power spectra of raw data traces. Data are selected by specifying a series or a sweep in the input data tree. In addition, the cursors have to be set such that they delimit the data range of interest. The power spectrum is then computed in sections of the indicated number of data points as they are displayed (i.e., after digital filtering, leak and zero subtraction). Points have to be a power of 2 between 8 and 1,024. For example, if the cursors delimit a data range of 1,000 samples and *Points* is set to 256, then three power spectra are computed and accumulated (i.e., # *Avg.* is 3). The trailing 232 sample points are not considered for analysis.

Example

Raw Data: Simulated current records consisting of a superposition of the random fluctuation of one channel (2 pA) and Gaussian-distributed background noise (200 fA RMS). The potential was constant throughout the simulation such that the data serve as an example of “stationary current noise” suited for spectral analysis.



Power Spectrum: Power spectra of channel fluctuations and background noise were calculated using each 200 of the raw data traces shown above. While in *PulseTools* the spectral density is displayed in the dialog, the picture on the right hand side is a plot generated with *IgorPro* after exporting the results from *PulseTools* in “Igor Text” format.



Power Spectrum Dialog

The power spectrum is plotted on a double-logarithmic scale in units of A^2/Hz as a function of the logarithmic scaled frequency in Hz. The scaling of the graph is done automatically according to the limits of the spectrum.

Reset: Sets spectrum and number of averages to zero and clears the graph.

Run: Computes, accumulates, and shows power spectra.

Notebook: Writes index, frequency, and spectral density of the spectrum as a table into the *Notebook*.

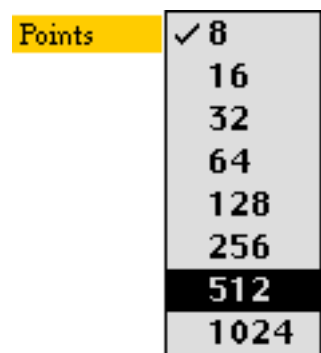
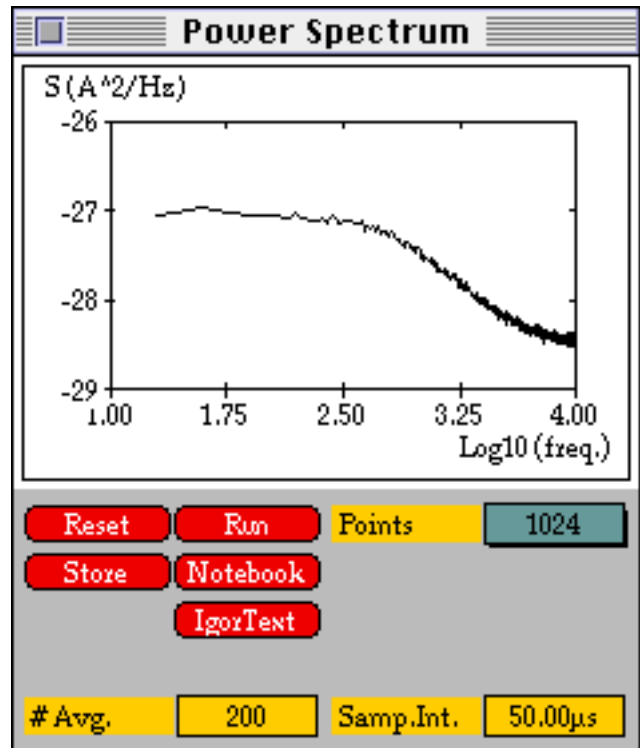
Store: Stores the power spectrum in the output data tree. The sample interval is interpreted as frequency increment. Spectral density values are stored in “Real32” format. For replay and export, the unit is automatically set to “Hz”.

Note: The number of data points in the spectrum is $Points/2$, resulting in frequency ranges from 0 to $(0.5/Sample\ Interval)$.

Igor Text: Export spectrum in “Igor Text” format. Upon load of this file with *IgorPro*, a graph with the correct scaling and units is generated.

Points: Number of data points to be taken for computation of the power spectrum. The pull-down menu containing the possible numbers that can be selected is shown on the right.

Avg.: Number of data blocks which were used for the computation of the spectrum. This number is stored in the output series record as “Number of Averages”, as used for on-line-averaging in *Pulse*.

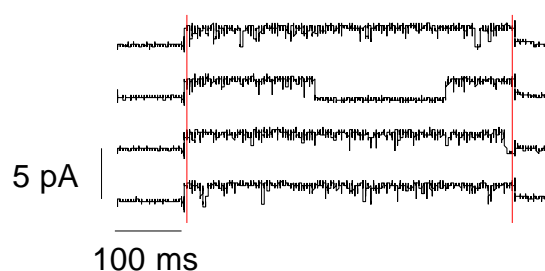


7. Amplitude Histogram

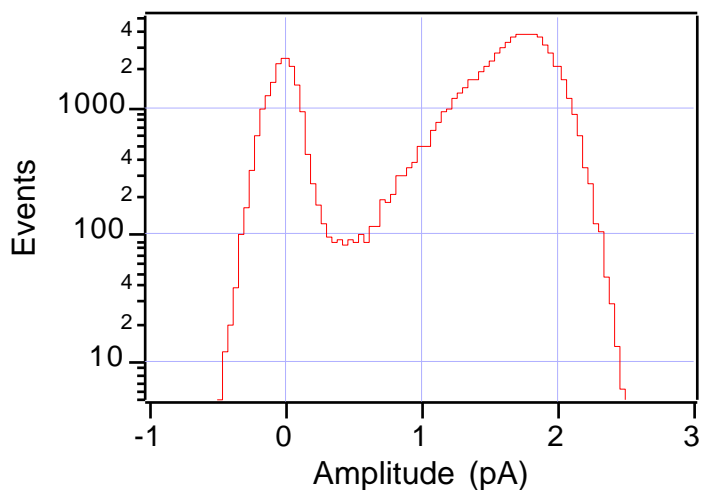
This option compiles amplitude histograms on a sample-to-sample basis. The result of such an analysis is a plot of the number of counts versus the left bound of an amplitude bin which can be stored in the output tree. For further analysis, such as fitting of distribution functions to the histogram data, the stored histogram can be exported in a format as specified in the *Tree* menu (see *Pulse Manual, Chapter 3 - Menus*) or the graph can be exported as an *Igor* graph directly from the amplitude histogram dialog.

Example

Raw Data: Raw data traces of single-channel recordings acquired with *Pulse*. Histograms were compiled for the time range specified by cursor settings (vertical lines).



Amplitude Histogram: The compiled histogram can either be stored in the output tree or exported as an *Igor* graph, which will appear with the same axis settings as the graph displayed in the amplitude histogram dialog. The example on the right corresponds to the histogram displayed in the dialog below but was exported with the option *Log. Events* on.



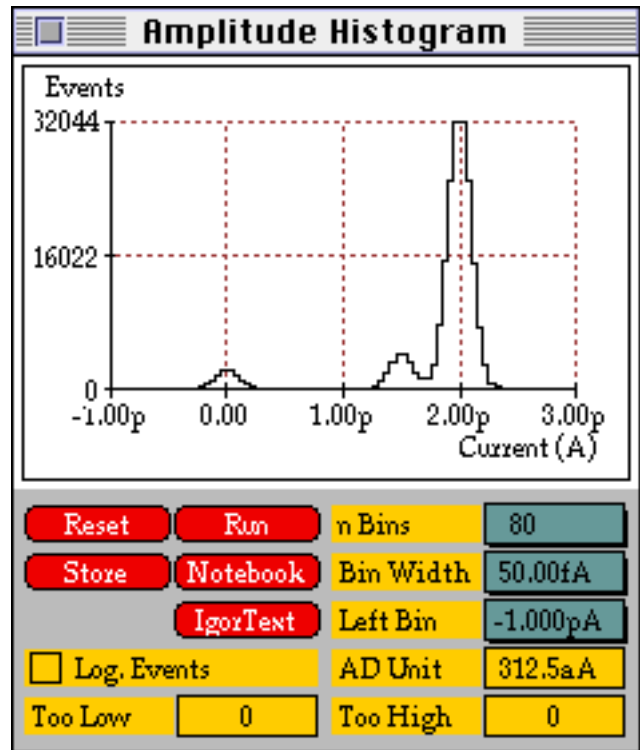
Amplitude Histogram Dialog

Reset: Sets the counts of all histogram bins to zero and erases the graph.

Run: Accumulates the number of data points per current bin within the data range delimited by the cursor settings. This operation is performed either on individual sweeps or on a series, depending on the selection in the input tree.

Store: Stores histogram values as a sweep trace in the output tree. Sweep and series information is copied from the input tree. For replay, *Zero Subtraction* has to be turned off since the left bin is stored as *Zero Current*. In the PGF-dialog, time has to be interpreted as current.

Note: Since a bin can contain many more than 64,000 counts, the raw data format of amplitude histograms is 32-bit integer rather than 16-bit.



Notebook: Prints a list of the counts and the current bin values (left edge) in the *Notebook* window.

IgorText: Exports the amplitude histogram graph as an “IgorText” file. The graph is setup such that the event wave is shown in “Cityscape” format, i.e., the Y-value is shown as the left edge of a bin (the same is done in “HistogramBin” format). When functions (e.g., Gaussians) are fitted to these histograms in *IgorPro*, the resulting fit function has to be shifted by $Bin\ Width/2$ to the right. Also note, that when showing a histogram in *IgorPro* in “Cityscape” or “HistogramBar” format, the last bin is not completely shown (i.e., n data points yield n-1 bins). This inconvenience can be fixed by duplicating the last data point and appending it to the histogram waves.

n Bins: Number of current bins.

Bin Width: Current range of a single bin. The overall current range for the X-axis of the histogram is therefore the *Bin Width* multiplied by *nBins*.

Left Bin: Left edge of the first current bin.

AD Unit: Current resolution specified by the hardware used for data acquisition.

Note: In order to avoid binning artifacts, “Bin Width” should be a multiple of “AD Unit”.

Too Low / Too High: Number of events detected outside the range of the histogram.

Log. Events: Shows the number of events on a *Log10*-scale. Scaling is done automatically, i.e., the lower limit is one, the upper limit is scaled according to the peak of the histogram. The histogram is shown in linear scaling when *Log. Events* is off.

8. Event Detection

Event Detection

Detects transitions according to a 50% threshold criterion within the cursor-specified time region. This function finds transitions by a very simple algorithm. The transition time is determined by linear interpolation between the adjacent data points being below and above the 50% threshold. For level amplitude determinations the data point at which a transition was detected is discarded. The current level of the baseline (0) and the first open channel (1) have to be specified manually. The search algorithm then detects transitions between these levels and multiples thereof (e.g., levels -1, 2 etc.). Thus, there is no correction for drift in the baseline performed, neither are sublevel events considered. The transitions are accumulated in an event table. This event table can be exported for further analysis with *TACFit*. A subset of the table, restricted to levels 0 and 1 only, can be displayed as histograms (amplitude, dwell times and first latencies) in *PulseTools*. Such histograms can be output in *Pulse+PulseFit* format for further analysis with, e.g. *PulseFit*.

Reset: Clear event table. Set all counters to zero.

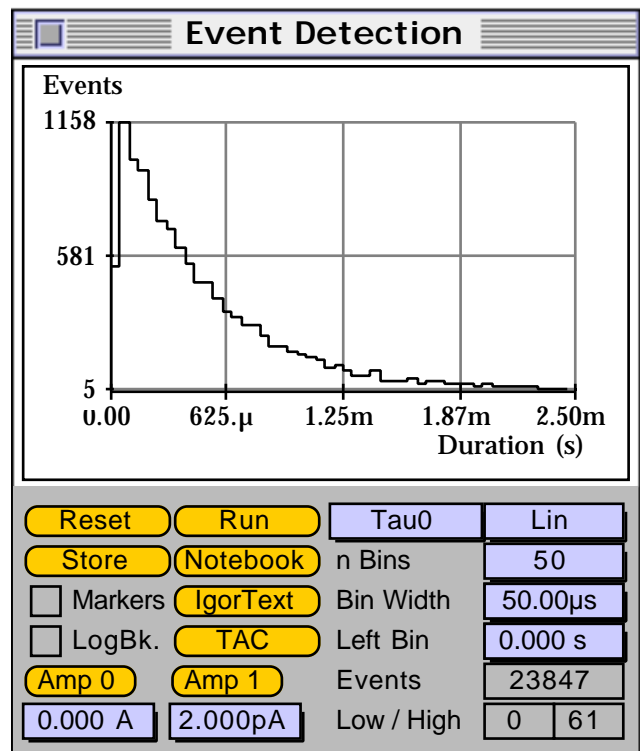
Run: Start analysis of data range selected in the input tree (e.g. individual Sweep, Series, or Group). The results are appended to the event table.

Store: Store histogram in *Pulse+PulseFit* format.

Notebook: Write event table into *Notebook* window.

Markers: If this item is selected, transitions are indicated in the *OsciWindow* by blue markers.

IgorText: Output histogram in *IgorText* format.



LogBk.: Write events into *Notebook* window during analysis.

TAC: Export ASCII file containing the event table. This file can be loaded by *TACFit* for further analysis of the event table.

Amp 0/1: Activate horizontal cursors in the *OsciWindow* for specification of zero and first channel current level. The resulting values are displayed in the fields below. Alternatively, these values can be entered manually.

Histogram content: This list is used to specify which parameter is going to be displayed in the histogram. Options are: Amp (all amplitude values), Amp0, Amp1 (amplitude values of levels 0 or 1), Tau0, Tau1 (dwell times at levels 0 or 1), and FirstLatency (latency until first transition to level 1).



Y-axis mode: This list specifies the scaling of the y-axis. Options are: Linear, Square root, Logarithm.



nBin: Number of bins.

Bin Width: Width of bins (in seconds for dwell times, in Amperes for amplitudes).

Left Bin: Start of the histogram (in seconds for dwell times, in Amperes for amplitudes).

Events: Total number of events presently stored in the event table.

Low/High: Number of events which are outside the specified histogram range.

Appendix I: Examples

Nonstationary Noise Analysis & Variance Fit

Introduction

In cases where excessive background noise obscures the resolution of single-channel current events or when a preparation contains many active channels, noise analysis can be applied in order to obtain information on single-channel parameters such as the single-channel current (i), the maximal open-probability (p_o) and the number of active channels (N). For signals which do not show significant alterations of channel activity (steady-state case), spectral analysis or analysis of the time variance can be applied. For transient signals like the activity of voltage gated ion channels, however, ensemble variance of successive iterative stimulation have to be considered. As introduced by Sigworth (1977, 1980), the ensemble variance σ^2 plotted versus the mean current I is described by a parabolic function which is characterized by the slope at zero, i , and the number of active channels, N :

$$(\sigma^2 - \sigma_b^2) = i I - \frac{I^2}{N}$$

where σ_b^2 is the baseline variance (background noise). This equation is based on statistically independent ion channels for which hold:

$$I(t) = N i p_o(t)$$

and

$$\sigma^2(t) = N i^2 p_o(t) [1 - p_o(t)]$$

Pulse Protocol for Data Acquisition

The aim of non-stationary noise-analysis is to determine i , N , and the maximal open-probability for a given potential. In order to constrain the parameters of equation (1) it is of advantage when $p(t)$ undergoes a large variation during the test pulse interval, ideally exceeding the value of 0.5. Experiments in which high $p(t)$ values are yielded therefore provide most reliable estimates.

If no leak correction is to be performed, a series of identical test depolarization can be acquired. For correction of leak and capacitance currents test pulses have to be nestled with leak pulses. The leak pulses must have the same temporal pattern as the

test pulses; the voltages must be in a range where channels do not open. In order to avoid mistakes in designing the noise pulse protocols in the Pulse Generator, one may proceed according the following scheme (based on the example):

Generate test pulse sequence: Name "Noise"

- Turn off the on-line leak correction, NumberLeaks=0
- Make three segments, e.g. -100, +80, -100 mV of appropriate duration and sample interval. Do not use Vmembrane flags for determination of the potentials.
- Set trigger to the middle of the first segment in order to avoid capacitance artifacts.
- Set relevant X and Y segments to 2.
- Select number of sweeps: here 25.
- Select sweep interval: slow enough to avoid cumulative inactivation, but fast enough to yield many data traces within a minimal time.

Generate leak pulse sequence: Name "Leak"

- Copy "Noise" sequence and name it "Leak" (names are arbitrary).
- Set segment potential, e.g. -130, -100, -130 mV
- Adjust number of sweeps: here 6
- Adjust sweep interval to zero in order to save time during acquisition.
- Link this sequence to the "Noise" sequence: enter "Noise" in the linked sequence field.

Link "Noise" to "Leak"

- Click on LINK within "Leak" sequence; this brings you to "Noise"
- Enter "Leak" in the linked sequence of "Noise".

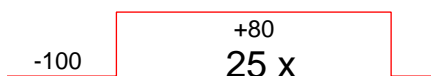
Now you have a cyclic linked noise sequence: 25x Noise - 6x Leak - 25x Noise - etc. Depending on the single-channel current and the background noise one has to repeat this cycle several times. For a low background noise and i on the order of 1-2 pA, 10-

20 cycles should be sufficient (see example); for the measurement of gating noise signals several hundreds of cycles are required. Stop execution by BREAK or STOP. In order to ease analysis one should create a new Group (or new Experiment) after acquisition.

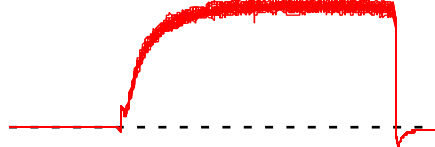
Note: Instead of only one Leak protocol one can use multiple protocols: e.g., "Leak_p": -120, -100, -120 mV; "Leak_n": -120, -140, -120 mV. One would then link this sequence to "Noise" - "Leak_p" - "Leak_n" - "Noise" etc.

Example

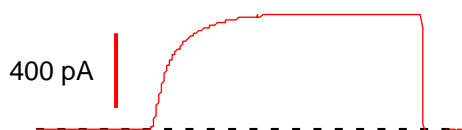
Test



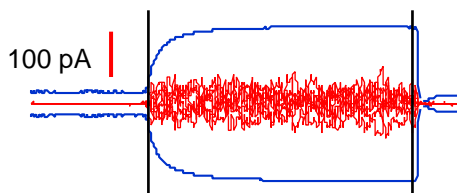
25 x test



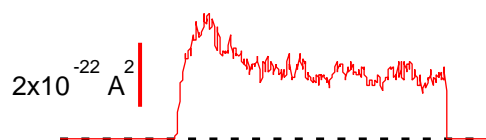
mean test, leak corrected (256)



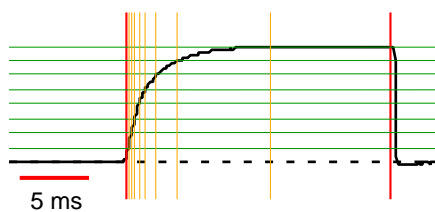
difference records (8)



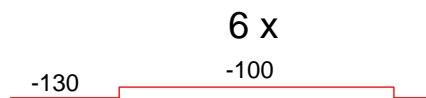
variance (243)



binning for variance-vs-current (8)



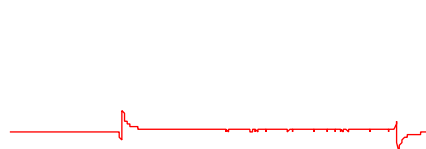
Leak



6 x leak



mean leak, scaled (60)



variance versus current
20 bins

