Signal Plant 1.2.1.17

F. Plešinger, 2016

INSTITUTE OF SCIENTIFIC INSTRUMENTS
The Czech Academy of Sciences

Index

Introduction

About Signal Plant

Signal Plant is a software tool for viewing, marking and processing of measured signals. It is designed to work with multichannel biosignals like ECG and EEG, but there is not any limitation to process different type of signals. Signal Plant has been developed by MEDISIG group, Institute of Scientific Instruments of the Czech Academy of Sciences.

License

Signal Plant is **FREE, even for commercial projects**. It is published under MIT license. **Using this software you agree with MIT license**.

Key features

- friendly and easily handled user interface
- functionality expandable by plugins (sample plugins with source code are available in Downloads)
- basic plugin set contains filtering, I/O, display, mark editing, batch processing plugins and lot more
- non-destructive editing
- easy work with marks/triggers
- 64-bit architecture for large files
- multi-thread computations (and display-related tasks)
- quering of marks nad channels

Minimal requirements

- OS MS Windows 7+, 64-bit
- 1GB of free RAM (size of RAM defines maximum size of processed files. Expect at least 4GB RAM for processing of 1GB file)
- processor with at least two cores. More cores => better performance

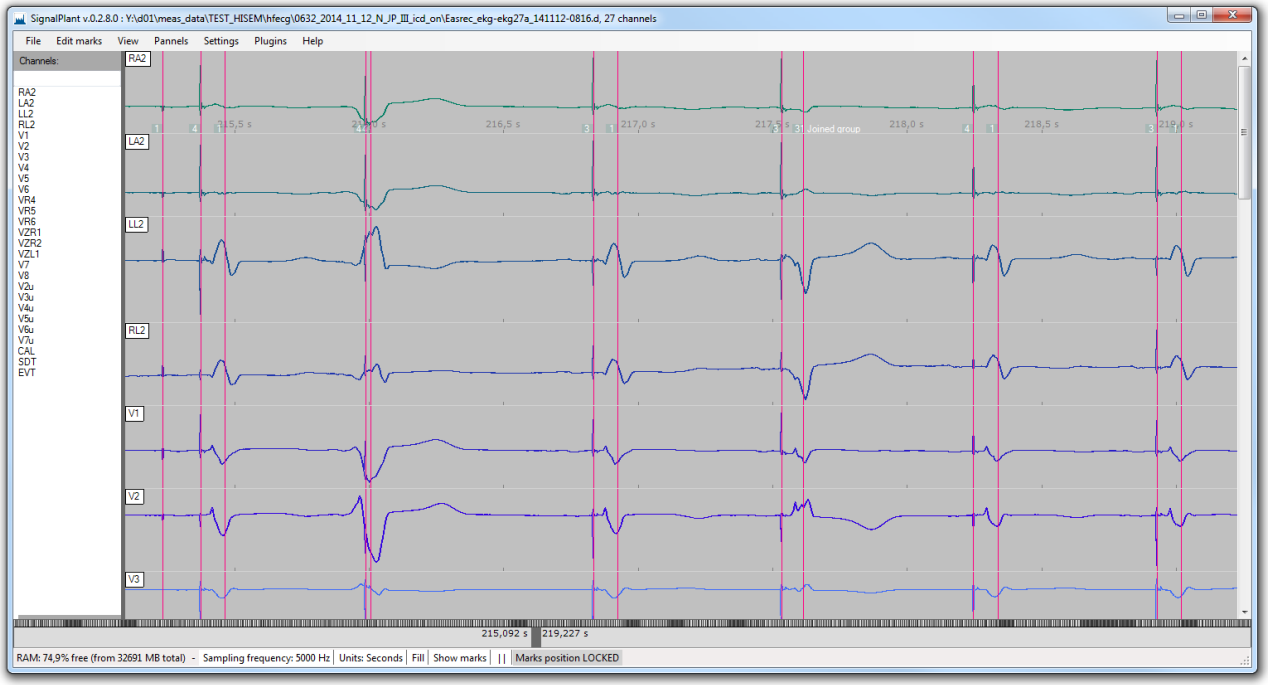


Figure 1 - Signal Plant program

Instalation

Obtain a serial number

Download and unpack *.zip archive to some location, where program has ability to write data.
Then start signalPlant.exe. Following window will appear (fig. 2).

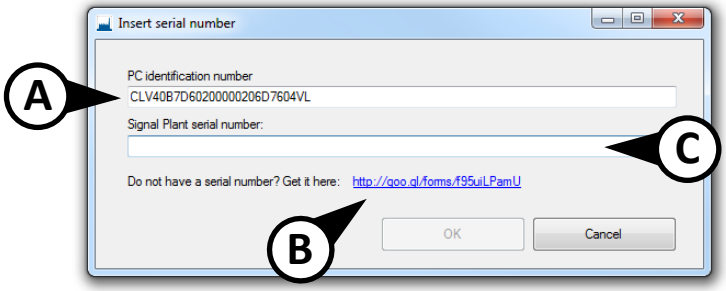


Figure 2 - Serial number form: A - PC identification number, B - link to SN generator, C - Serial number

You will be asked for a serial number (fig. 2). If you do not have one, copy PC Identification number (fig. 2-A) by CTRL+C into clipboard and click on the link (fig. 2- B). Your web browser should open web page with registration form (fig 3).

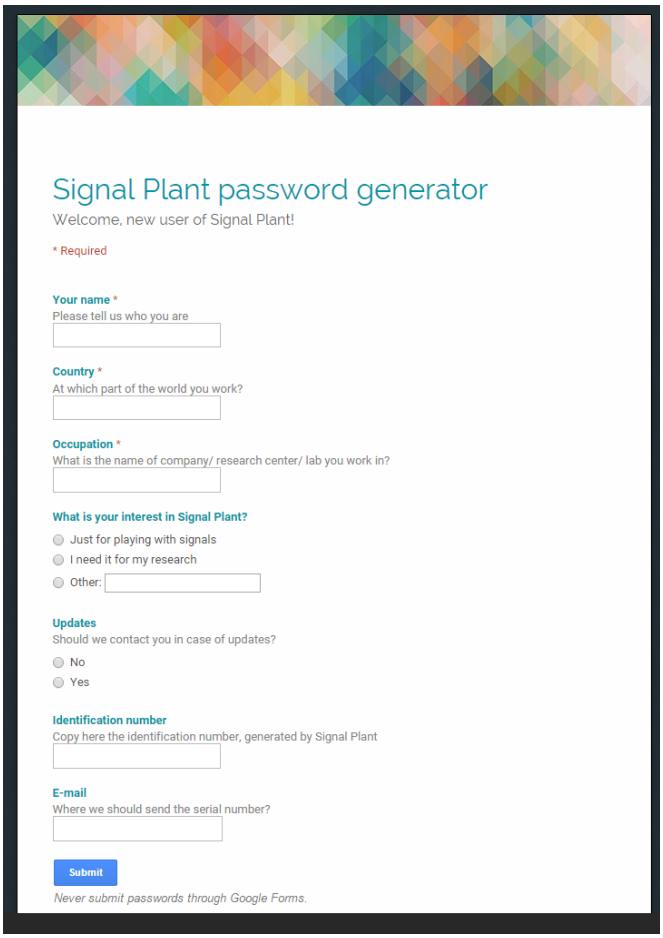


Figure 3 - Web password generator

Fill proper fields and submit form. In a few minutes you should recieve email with serial number.
Copy serial number from incomming message and paste it (CTRL+V) into Serial number field (fig. 2- C).
When OK button change to enabled state, click on it.

Settings

If default paths for plugins, temporary files and scripts does not exist in the same location as SignalPlant.exe, you will be asked to set them in „Settings“ window (fig. 4). Set all three paths (fig. 4 - a,b,c) to existing locations. This window is a kind of cruel - until you set those paths, this dialog will not let you go anywhere else.

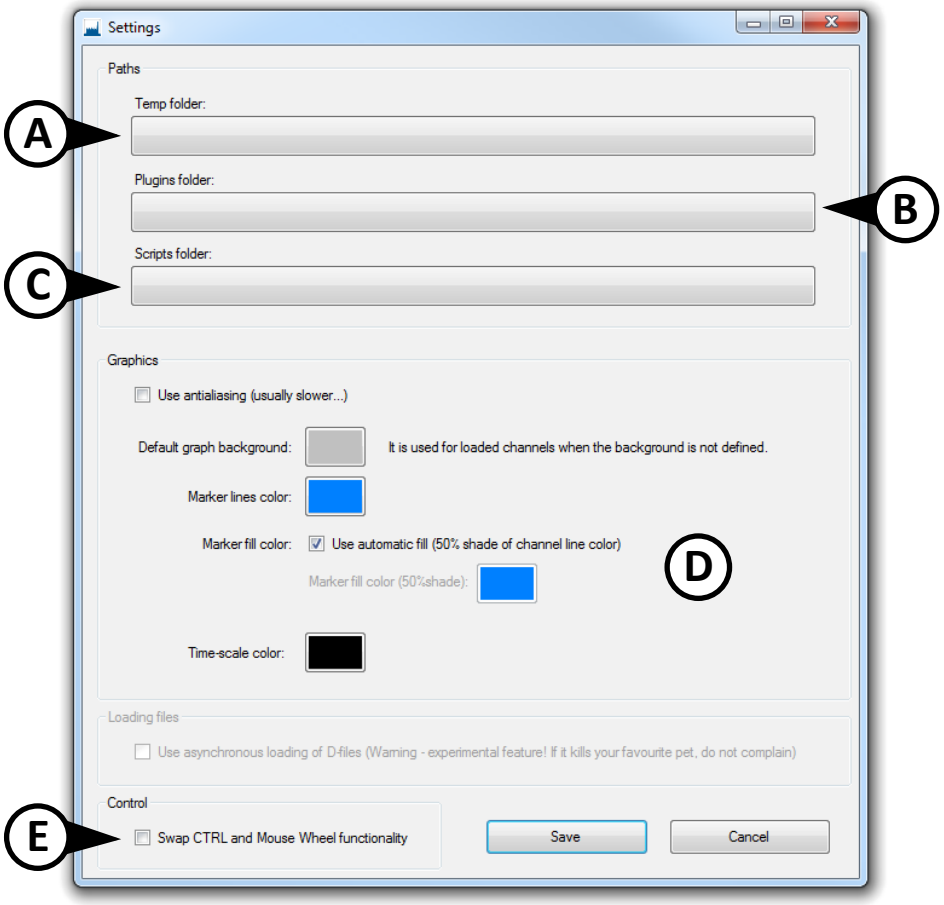


Figure 4 - Settings window: A - Temp folder path, B - Plugins path, C - Scripts path, D - Graphics settings, E - Control settings

Other settings are aimed to graphics (fig. 5 - d) and mouse control (speciality for one grazy neuro-scientist). It is not important to change them now; it can be done later by click on SETTINGS in main menu.

Press SAVE button. Settings dialog will close and Signal Plant will start.

Used file formats

Native file formats

Signal Plant is able to work with several file formats. Next table presents I/O (input & output) abilities, included in basic plugins set.

File format	I/O operation	Data	Marks	Hold datacaches	Note
.w (ScopeWin)	Read	Yes	No	No	
.d (M&I)	Read	Yes	Yes - reduced	No	
EGI files	Read	Yes	Yes	No	
.csv	Read & write	Yes	No	No	
.mat (Matlab)	Read & write	Yes	No	No	Needs Matlab(R)
.sel	Read & write	No	Yes	No	
.edf	Read	Yes	No	No	
.h5 (HDF5)	**Read & write**	**Yes**	**Yes**	**Yes**	

As a native file format able to hold all data including signals, marks and signals variations, HDF5 file format was choosed. To learn more detail how HDF5 files are organized see **appendix A**.

For operating EDF files we uses libraries made by Peter Gordon, available from **https://edf.codeplex.com/**

Reading MAT files requests Matlab installed on the target computer. How to transfer data from Matlab to SignalPlant is describeb in **appendix B**.

In addition to presented data formats it is possible to export both vector and raster images of the main program view with signals as **SVG**, **EPS** or **PNG** file (via FILE|SAVE).

Additional file formats

File format list can be extended by writing specific I/O plugins . Example (save & load simple *.csv file) is available on project web site **https://signalplant.codeplex.com**.

Data structure

Record file

Signal Plant is able to open and process one record file at once. Only exception is the Batch processing plugin.

Channels

Channel is a structure dedicated to keep specific signal. Each record file holds (usually) several channels. Channel is identified by its name (and that name should not contain # ! = > < | symbols). In case of ECG record, there will be probably channels named as V1, V2, V3, V4, etc. Common EEG record holds larger number of channels. Signal Plant is not limited in number of channels in a record; the available RAM is the only limitation.

Channel structure holds other properties as unit or line color and background color (applicable only in standard view).

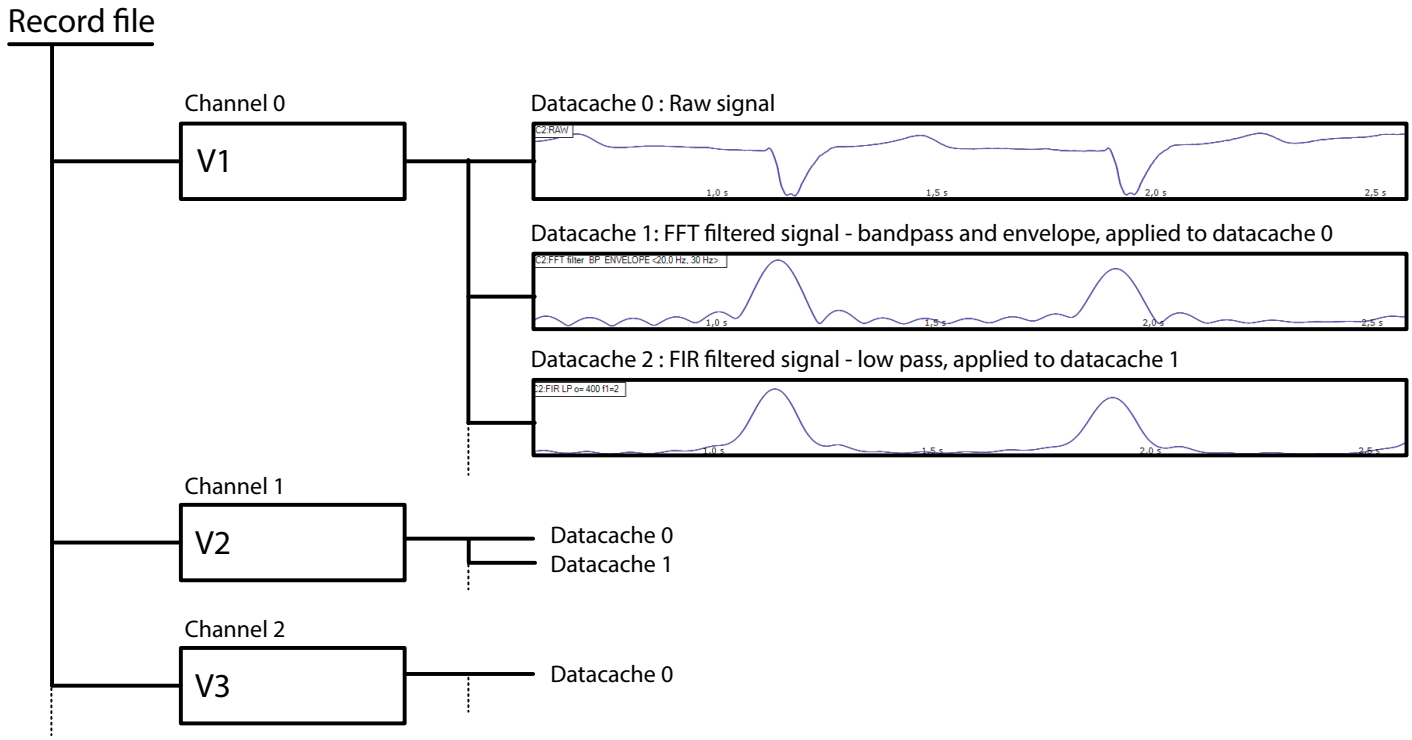


Figure 5 - Datastructure

Datacaches (i.e. layers)

Measured data inside each channel is hold in datacache structure. Each channel is able to hold unlimited number of datacaches and only one of them can be active = visible.

Imported files usually contains only raw signals and each channel contains only one datacache. When user do (for example) filtering, new datacache is added to selected channel with new data and new name. User is able to switch active datacache and examine (in this case) filtered and raw signals.

Important - processing methods (plugins) usually use the active datacache as the source signal.

This datastructure allows **non-destructable editing** ability.

Marks

Marks serve to select specific locations in time like artifacts, stimuli or QRS annotations. Location in time is specified by left and right border of the marker. Both borders can be the same; such marks are displayed as vertical lines (case of stimules or QRS). When they are not the same, the area between borders is displayed as semi-transparent area (case of artifacts).

Mark structure holds several properties, which can be used for mark quering:

- left and right POSITION
- INFO field (i.e. „QRS“ or „PACEMAKER_STIMULUS“)
- LINK to any channel (i.e. V1)
- VALIDITY (i.e. 0.15)
- GROUP (i.e. 1, 2 etc)

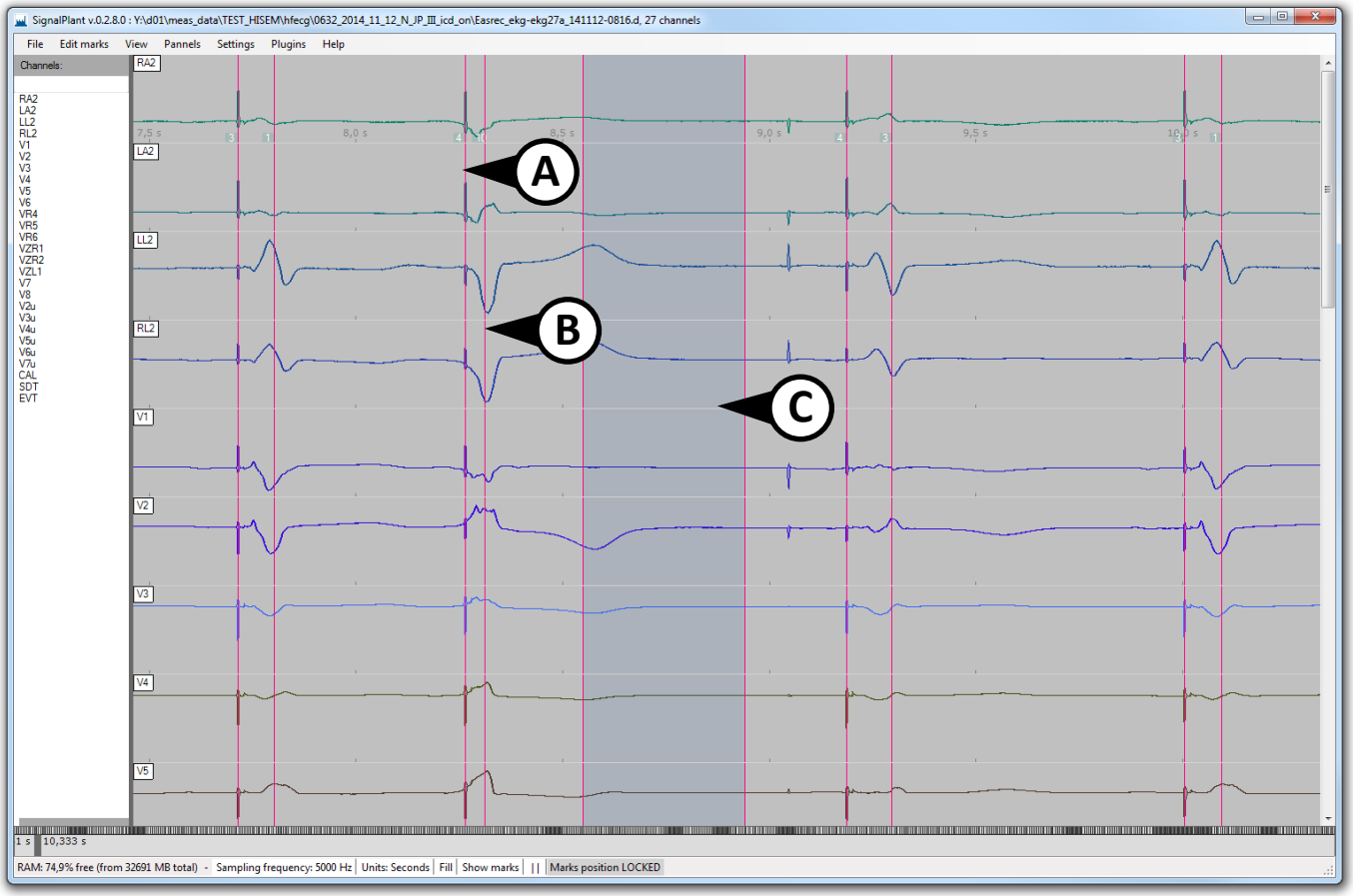


Figure 6 - different mark types : vertical lines (A - pacemaker stimuli and B - QRS complexes) and wide mark in the middle - C

It is upon users will if they uses specified fields. Some plugins fill those fields automatically like the QRS detector (fills POSITION, VALIDITY and INFO) and Sorter (fills POSITION, INFO, VALIDITY and GROUP).

Then it is very easy to do group operations with specific selection of marks.

Marks can be hold separately in *.sel files (containing only marks informaiton).

Quick start guide

Load a file

Drag any suitable file into the main window (or use File | Open to select a file). If there is a *.sel file named exactly the same as the opened file (except the extension), you will be questioned if the marker file should be imported too.

Select channels to display

After loading, all signal channels are displayed in single bars, organized in stack (fig. 7-A). Name of each channel is displayed on the left side of each bar (fig. 7-B).

To show specific channel only, just click on it inside „Channels“ panel (fig. 7-C). Use Click+Ctrl to show more channels together.

You can use small text box (fig. 7-D) in top of the Channels panel to filter specific channels by name (more in Channels quering chapter).

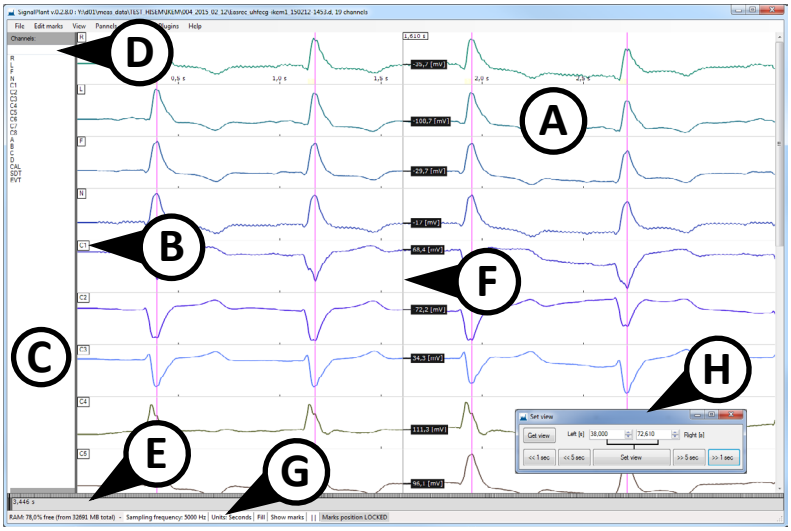


Figure 7 - Signal Plant main window. A - Signals, B - Channel name, C - Channels panel, D - Channels filter box, E - Navigation bar, F-Ruller, G - Units button, H - Set time window

Basic navigation

Use LMB to pan and mouse wheel to zoom. Drag the bottom of each graph to vertical resize. If you press Ctrl while doing this, all channel bars will get new Y-size. Or use the navigation bar (fig. 7-E) to change the visible area. Dragging its center does the panning, dragging its borders does scaling. Other keyboard shortcuts can be used to navigate in time - Home to start, End to end and PgUp/PgDown to paging. If you need to set visible time area precisely, press the RMB on the active part of navigation bar or press CTRL+T to show „Set time window“

Using a ruller for tracing

If you need to measure signal values in a given time, just press Ctrl + LMB in any channel bar and ruller (fig. 7-F) will be placed in desired position as a thick, black line. On the top is the information about time. If you need to switch between samples and seconds, press UNITS button (fig. 7-G) at the bottom of the program window. To show or hide specific values in visible channels, switch View | Ruller | Show value.

Filtering - band stop at 40-60 Hz

Run Plugins | Filters | FFT filter. Plugin window with FFT filter will appear on the screen. Let us select channels to filter. It can be done in several ways, but the simplest one is to drag the name of desired channel (fig. 7-B) to the button SELECT CHANNEL (fig. 8-A). Now we can see frequency spectra and spectral area to remove (fig. 8-B). Below (fig. 8-C) is the source signal (gray) and filtered signal (black). Default settings are set to our purpose, so press PROCESS BUTTON (fig. 8-D) and wait. When the filtering is done, plugin closes itself.

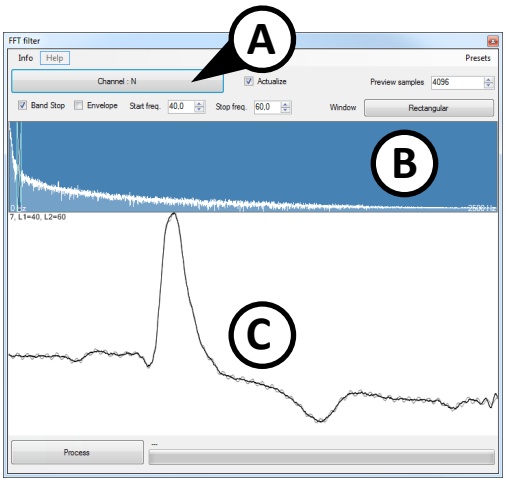
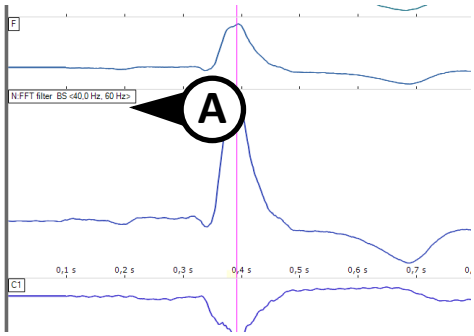


Figure 8 - FFT filter plugin. A - Select channel button, B - Frequency spectra, C - Preview



The filtered channel has new datacache holding filtered data. This new datacache is active now. Such an information is presented on the right of channel name (fig. 9-A). LMB + Shift on that name lets you cycle through existing datacaches - displayed data will reflect active datacache.

Figure 9 - Channel after applying FFT filter. A - New datacache name

FFT analysis

To examine the effect of applied filter, run the FFT analysis plugin at Plugins | Analysis | FFT. Attach filtered channel to it. If you switch the datacache now, you will see the change in FFT spectra immidiatelly.

Left border of area for FFT analysis is defined by the ruller, so try to change its position (Ctrl+LMB on signal) to see a spectra for different time area (width of analysed area is defined by „FFT window" textbox - fig. 10- A).

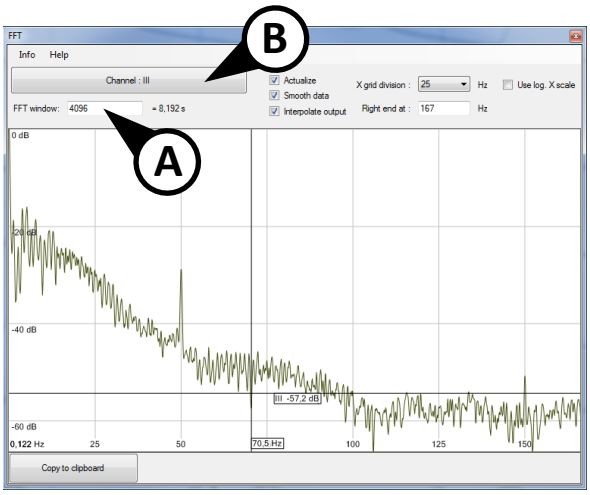


Figure 10 - FFT Analysis plugin, applied to result of filtering. A - FFT window text box, B - Link channel button

How to label some area?

Press the Shift+LMB and drag mouse over that area to **create new mark**. After release of LMB marker properties dialog will pop up (it is one of plugins from basic set). You can set any property of selected mark in this dialog window (that it is an artifact or anything) but now just press Enter to accept implicit values.

Generate marks from signal peaks

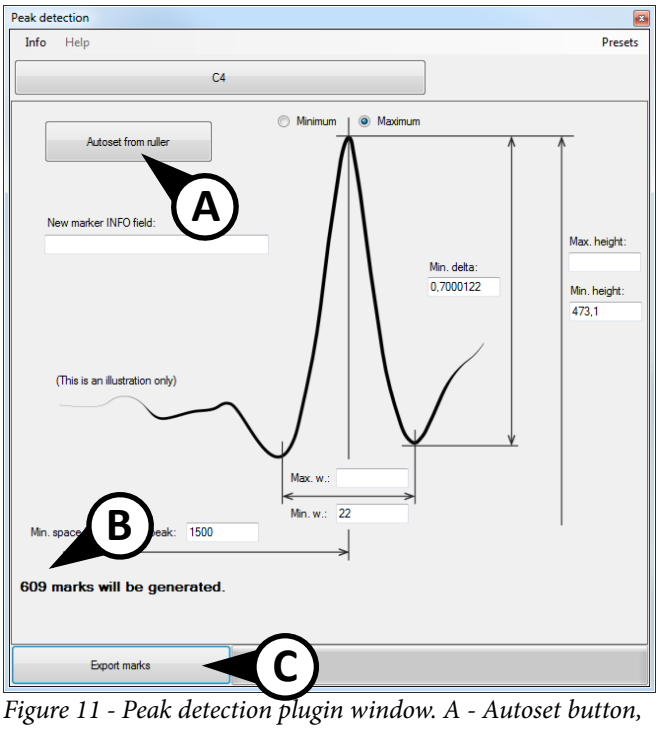


Figure 11 - Peak detection plugin window. A - Autoset button, B - Export marks button

If you want to generate marks in more efficient way, use some automated approach. The simplest one is using the „Peak detection" plugin. Start it by clicking on Plugins | Detection | Peak detection. Place the ruller (Ctrl+LMB) close to a „prototype" peak in a signal.

Link desired channel to detection plugin and then press button AUTOSET FROM RULLER (fig.11-A). Values from peak closest to the ruller in a specific channel are derived and displayed in corresponding text boxes. Change them to achieve satisfying results. At the bottom (fig. 11-B) you can see how many markers will be generated. This value is recomputed in background after you change any of criterias.

When you are done press EXPORT MARKS button (fig. 11-C).

Change mark position

To change mark position, the simplest way is to unlock MARK POSITION (fig. 12-A) and then you are able to drag mark border into the new location.

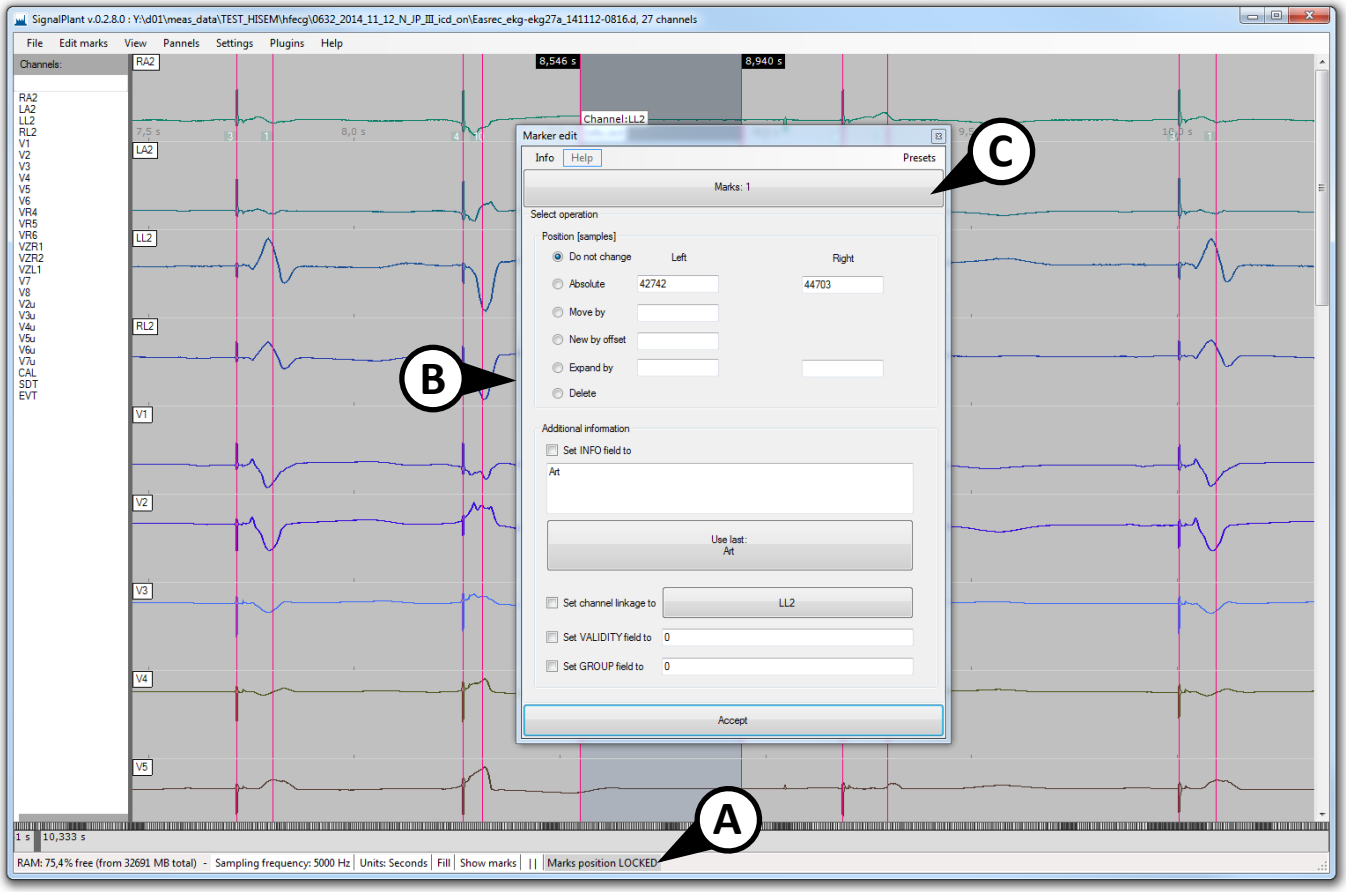


Figure 12 - Marks and editing. A - Marker position lock, B - Marker edit plugin, C - Marks select button

Change all properties of mark

To change any mark properties (including position), just double click LMB on mark. Marks edit plugin will pop up (fig. 12-B) and you can change mark properties. This plugin (found at Pugins | Maintenance | Mark edit) is able to change properties for specified group too - just click on the top large button (fig. 12-C) and define subset of markers (as described alter in chapter Marks Querying).

Export marks to a separated file

If you need to export existing marks to separated file, use File | Export marks. Marks will be exported to a *.sel file, easily readable textfile with description written inside at the top of the file. If you preserve offered name and location, next time you load the data file you will be questioned to load *.sel file too.

Get picture of signals into your favourite word processor

To get picture of displayed signals, press the RMB inside any of channels. This will show pop-up menu. Press Copy picture and system clipboard will be filled with picture of current channel. If you press Shift, all visible channels will be printed into clipboard. Then switch inside your favourite word processor and press Ctrl+v.

Furhermore, image of signals can be saved as an image file - vector formats **SVG** and **EPS** are supported as well as raster format **PNG**. Export image file via FILE | SAVE.

Channels querying

Channel quering in main window

Channels to display can be easily selected by clicking in Channels panel (fig. 4-C). But in cases with a lot of channels (like EEG recordings) the process can be slow and painfull. For such reasons it is possible to select displayed channels by expression, written in small text window (fig. 4-D) and executed by pressing Enter key.

Quering use some reserved words and it is necesarry to avoid them in channel names:

- » ; works as AND operator
- » | works as OR operator (and can be written as Right Alt+w key)
- » ! means NOT operation
- » empty space means ALL (default option)

Examples:
To select all channels C1 to C10 in the record containing channels A1..A10, B1..B10, C1..C10, D1..D10, type „C“ and press Enter. Channels C1 to C10 will be selected.

If we need to exlude C1 and C10 from this set, write „C;!1“

Channel quering in plugins

When user needs to link channels to plugins, it is easy to do this by dragging. Again, in case of large channels to link it can be quicker to select them by an expression. After clicking on CHOOSE CHANNEL button (fig.10-B) the window „Select channel“ appears (fig. 13). Channels can be selected by mouse or by writing an expression (fig. 10-A). Visible channels can be recognized by an „Eye“ icon (fig. 10-B), selected channels can be recognized by low gray background below its name (fig. 13-C).

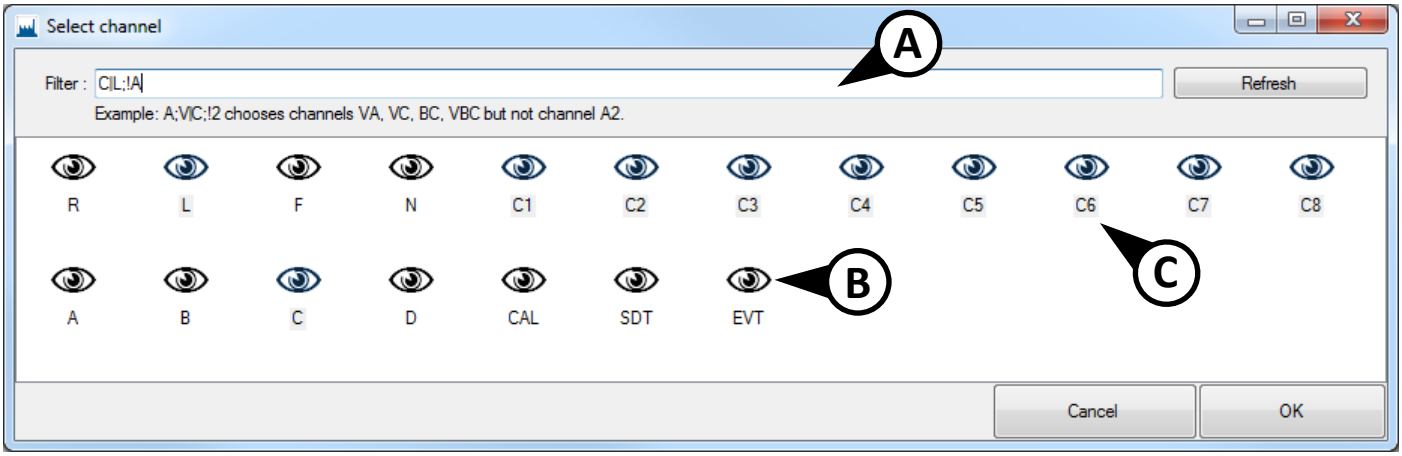


Figure 13 - Select channel window. A - Area for expression, B - Visible channel, C - Selected channel

Marks querying

Marks querying in plugins

Mark querying is used to select subset of all marks in record, for example select only QRS complexes of specific group and specific validity to plot them together. Marks querying uses the same reserved words as channel querying ; | and !, with combination to text information from INFO field. In addition to this, reserved words can be used with # symbol:

- » #GROUP Mark has to be from specified group
- » #VALIDITY Mark validity is examined
- » #TIME Mark position is examined
- » #CHANNEL Mark linked channel is tested
- » #SIMPLE Mark left and right borders has to be same
- » #INDEX Mark specified by its index

Examples:

1) To select only marks with INFO field set to „Art“, type: Art.

2) In file contaning marks for stimulation peaks (pacemaker) and marks for detected QRS complexes, let us select QRS complexes with GROUP field set to 1: QRS;#GROUP=1

3) Choose only marks from sample 1000 to 25000: #TIME(1000,25000)

Choose marks window

Querying of marks is usually done in „Choose marks“ window (fig. 14). There are helpfull buttons with reserved words (fig. 14-A), so you do not have to memorize them. After deffining a filter, **press Enter key**. Then blue bar at the bottom (fig. 14-B) shows marks occurance in the hole record. All available marks in record are displayed in bottom part of the window as dark blue vertical lines. Marks corresponding to current filter are displayed as light blue vertical lines and their count is displayed above (fig. 14-C).

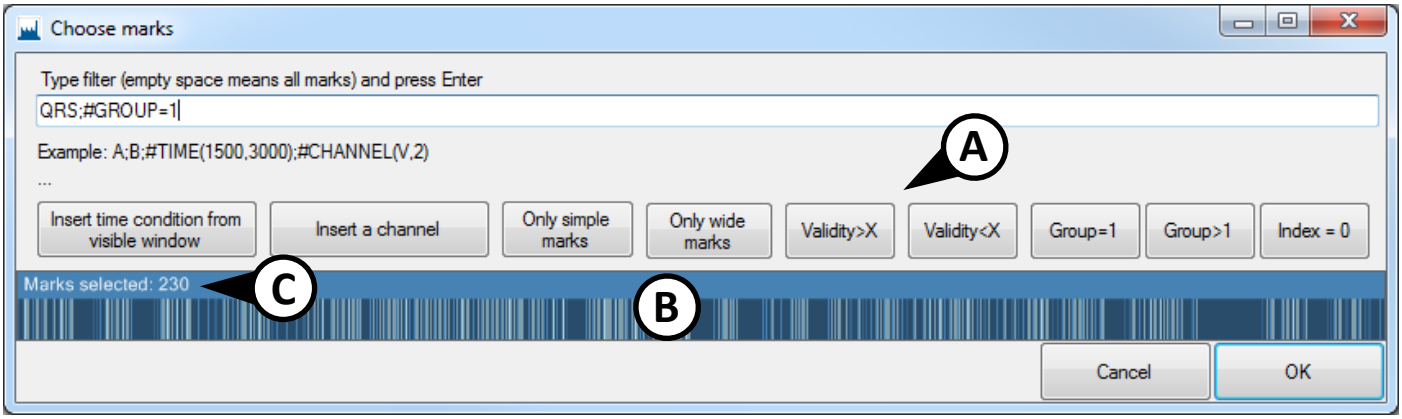


Figure 14 - Choose marks window. A - Buttons with predefined reserved words, B - Marks occurance within all the record, C - Count of resulting marks subset.

Mouse & Keyboard Controls

Legend: LMB = left mouse button, RMB = right mouse button, Move = mouse move. Keybooard shortcuts are blue.

Program controls

» F1 : Display help
» F2 : Open file
» F3 : Close file
» Ctrl+M : Shows panel with markers list

Display controls

» Pan : press and drag with LMB in signal viewport
» : or drag dark slider at the bottom of main window
» : or press Page Up/ Page Down / Home / End keys

» Zoom X-axis : turn mouse wheel
» : or press „/“ or „*“ to zoom in/out
» : or press F7 to fit all data to the screen
» : or press SPACE and drag LMB over area of interest (i.e. - zoom to window funciton).
(If you release LBM prior to SPACE, view will be returned to previous state after release of SPACE.)

» Zoom Y-axis : press F9 to autoscale On/Off
» Change vertical size of channel : drag LMB close to bottom border of signal bar.

» Ctrl+H : Show or hide all marks
» Ctrl+LMB : Show/Hide trace cursor. After release all plugins are refreshed.
» Ctrl + move : Show distance between ruller and mouse cursor
» Ctrl+Shift+move : Show frequency, defined by ruller and mouse cursor
» Shift + LMB : Cycle through existing datacaches

Marker edit

» Shift + LMB : create new single marker in viewport. Left border of the marker is defined when LBM is pressed, the seccond marker is defined when LMB is released. Thus, simple click creates just simple marker (vertical line).
» LMB & drag above existing marker : allows to change border of the marker
» 2xLMB above existing marker : shows plugin for markers editing

Channel edit

» 2xLMB above channel : shows plugin for channel properties editing
» RMB and drag channels over : change channels order

Plugins

About plugins

Plugins are used to extend Signal Plant functionality and are stored inside DLL libraries. Plugins to be loaded by Signal Plant must be placed inside Plugins path folder (see Settings in Instalation chapter).

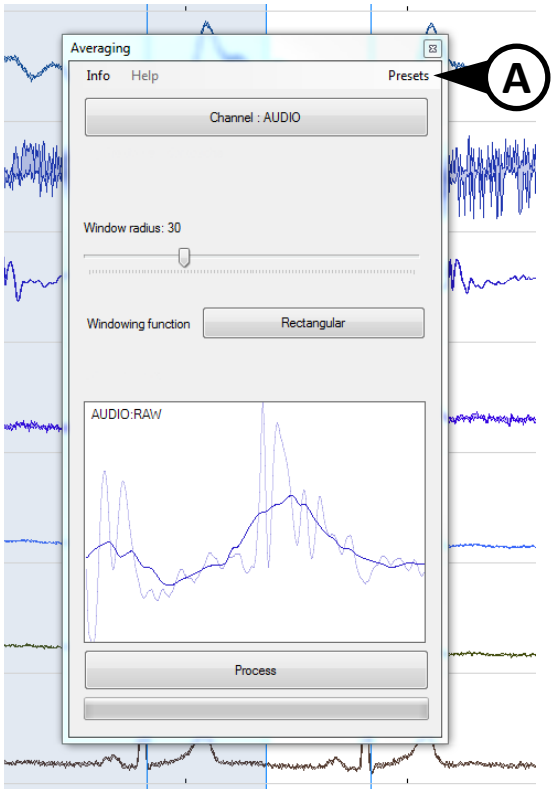


Figure 15 - Regular plugin example: A - Presets menu

Three different types of plugins are available:

» Regular plugins
» Display plugins
» Input/ ouput (IO) plugins

Regular plugins run in single window (fig. 15). They usually serve to specific task like filtering, analysis, detection etc.

To help users plugin are equipped with „Presets“ functionality (fig. 15 - A) (if the developer of plugin had implement it...). Presets functionality is the same through all regular plugins. It stores user values of controls for later invokation. User can create new presets from the current state by click Add new preset. Unused presets can be removed by Shift + LMB on specific preset name in menu.

Display plugins can be found in Plugins | Display. They not run in common window like regular plugins; instead they somehow change channels visualisation in main program window.

IO plugins are used to extend IO functionality and are not present in PLUGINS section of main menu. They are visible when user click on File | Open or File | Save items in main menu as available file formats.

Plugins and commands

Signal plant can be controlled by commands. These commands are available via special plugins as Command or Batch. Plugins have ability to add one or more commands to the SignalPlant command set. This approach is usefull in case of batch operations (when user works with large number of files).

Built-in regular plugins overview

This is the list of plugins, provided by file „plugins_package_1.dll“, which comes with Signal Plant archive. Chapter is divided into several parts, corresponding to categorization of Plugins menu. Plugin categories are derived from loaded plugins, so new categories can come with new plugins.

Filters

- Averaging (signal averaging by specified window type and size)
- FFT filter (Fast Fourrier Transform LP/HP/BP/BS filter with ability to produce envelopes)
- Downsampling (filtering and decimation to lower frequencies)
- FIR filter (Finite Impulse Response LP/HP/BP filter; adjustable order)
- IIR filter (very fast Infinite Impulse Response 2nd order filter)

Mantenance

- Channel detail (edit properties of channel; called by 2xLMB on a channel)
- Commands (command line for direct command; speciality for suiciders...)
- Batch (plugin for batch processing of scripts)
- Switch cache (usefull tool for global switch of datacaches)
- Marker edit (edit properties of marker; called by 2xLMB on a marker)
- Variables (shows window with registered Signal Plant variables)

Detection

- Peak detection (simple detection of peaks; try Autoset functionality)
- QRS detector (detector of QRS complex from multimodal biosignals, awarder in CinC Challenge 2014)
- Simple threshold (distribution of marks related to signal level)

Generate data

- Evaluator (generates new channel or datacache from a given expression)
- Mount (mount data using code for all channels)

Analysis

- FFT (Fast Fourrier Transform analysis with dB scale; switchable smoothing)
- Multisketch (plot signal in some range around specified marks)
- Compare averaged shapes (computation and comparison of an average signal shapes)
- Polar plot (plot two signals in some range around specified marks in polar maner)
- Sorter (plugin, developed for sorting of QRS complexes in UHF ECG signals)

Display

- Multidisplay (an alternative view for neurology specialists)
- Regular view (default view plugin for Signal Plant)

General plugins library

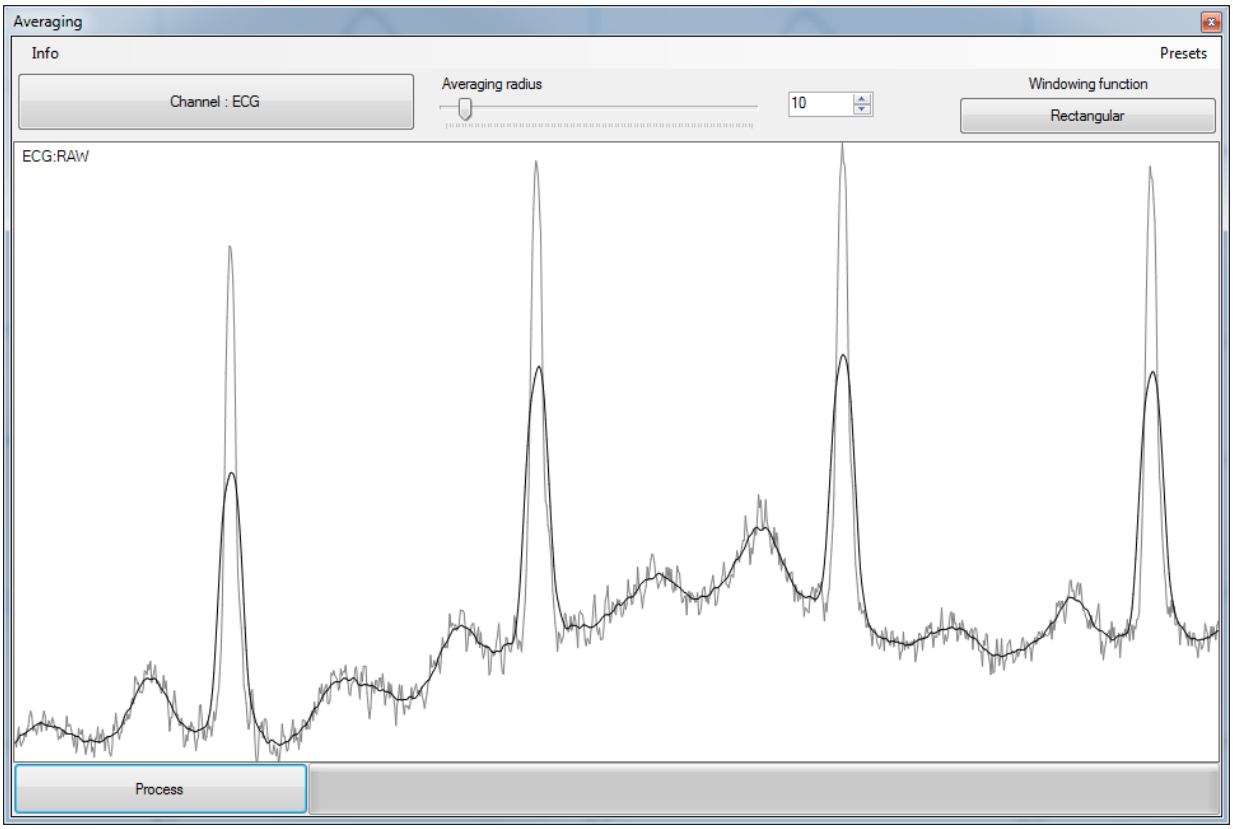
This chapter describes each member of general plugin set in more detail. These plugins are situated in „plugins_package_1.dll“ library, provided with SignalPlant by default. Some of plugins implement command(s) for batch processing. In such case there is always **example command for batch processing** paragraph.

Averaging

Averaging plugin allows to average one or more channels using specified window size and type, usually to remove high-frequency noise in signal.

Usage

Attach channels (by click on button in top-left corner or by drag & drop). Preview should appear (in black) as well as original signal (in gray). Set radius of averaging window by slider (top-middle) and choose window type by button in top-right corner. Implicit range for slider is 4-100, but it can be overided by numberic up-down selector right of the slider (up to 1000).



Example command for batch processing

```
AVG CHANNEL(V1;V2;V3) WINDOW(20,0,0);
```

will provide averaging for channels V1, V2 and V3 with window of radius 20, type 0 (=rectangular) and parameter 0 (it is applied only for Kaiser windowing function).

FFT filter

Most versatile plugin filter in SignalPlant. It uses „Fast Fourier Transform“ and lets user to select part of spectra to remove. Also, it support Hilbert transpform to create amplitude or power envelopes.

Usage

Attach channel(s) to plugin using button (Fig-A). The spectra will be shown in blue „Frequency area“ (Fig - C) from 0 to sampling frequency/2. It is smoothed for better readability (can be switched - Fig- B). Set filter type using switches (Fig - D) and change frequency limits using numeric boxes or by dragging them in frequency area. When using Band-Pass, amplitude envelopes can be produced using ENVELOPE switch. Furthermore, envelopes can be squared to power envelope using ^2 switch. Shape of selected frequency window is by default rectangular and it can be changed using button Fig-E. Switched at Fig-G are used to display.

Start filtering using PROCESS button (Fig - H).

Source and result signals are shown in panel Fig-F.

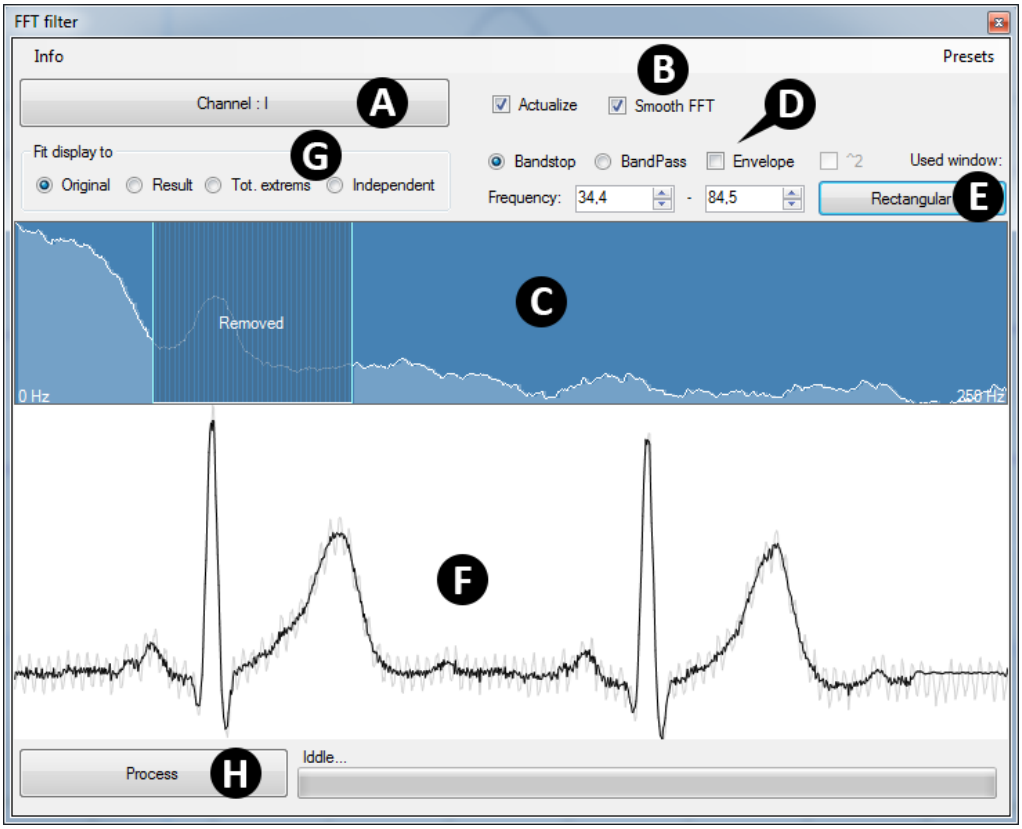


Fig.

Warning: FFT allocates large amount of memory during processing. Check memory size if problem occurs (and use FIR or IIR filter instead).

Example command for batch processing

```
FFTF CHANNEL(V1;V2) FREQ(40; 60)
```

filter out 40-60 Hz from channels V1 and V2.

```
FFTF CHANNEL(V) FREQ(500; 1000; BANDPASS; ENVELOPE)
```

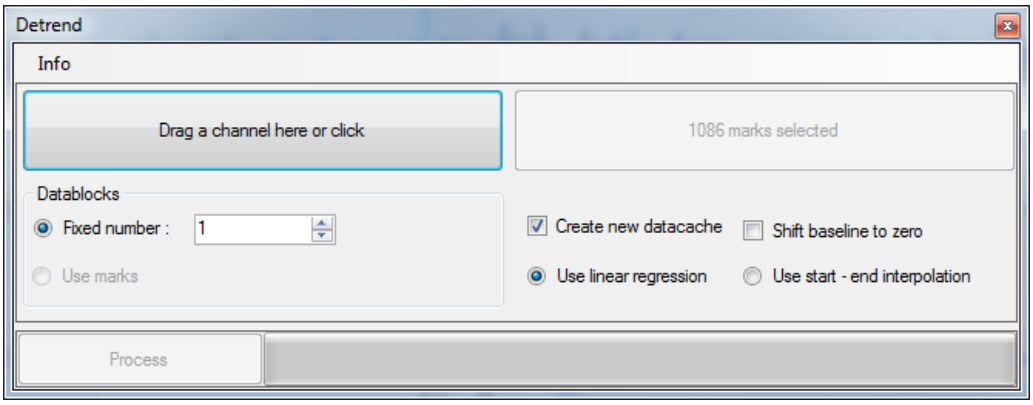
left only frequencies in range 500-1000 Hz, converted to amplitude envelopes in all channels containg „V“ in their name.

Detrend

provides detrendind using linear regression or the first and the last values of datablock(s).

Usage

Number of datablocks can be set using numeric box (Fig-left). Datablocks can be defined by markers (in a future...).

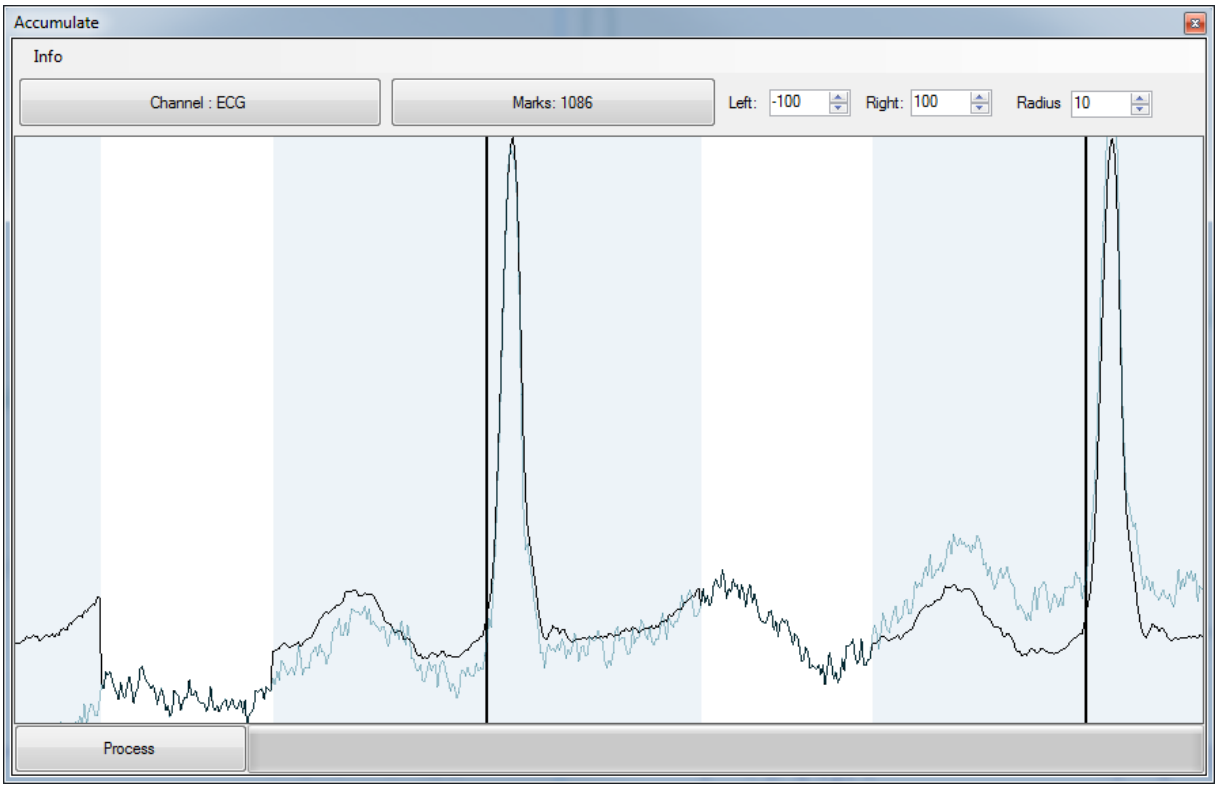


Accumulate

accumulates (averages) signal(s) in a given area using mark positions as triggers.

Usage

Left and Right numeric fields define area (to accumulate) around individual mark (in samples). Size of area around marks is also represented by light blue color in preview panel. Radius parameter defines how many consecutive areas will be accumulated.

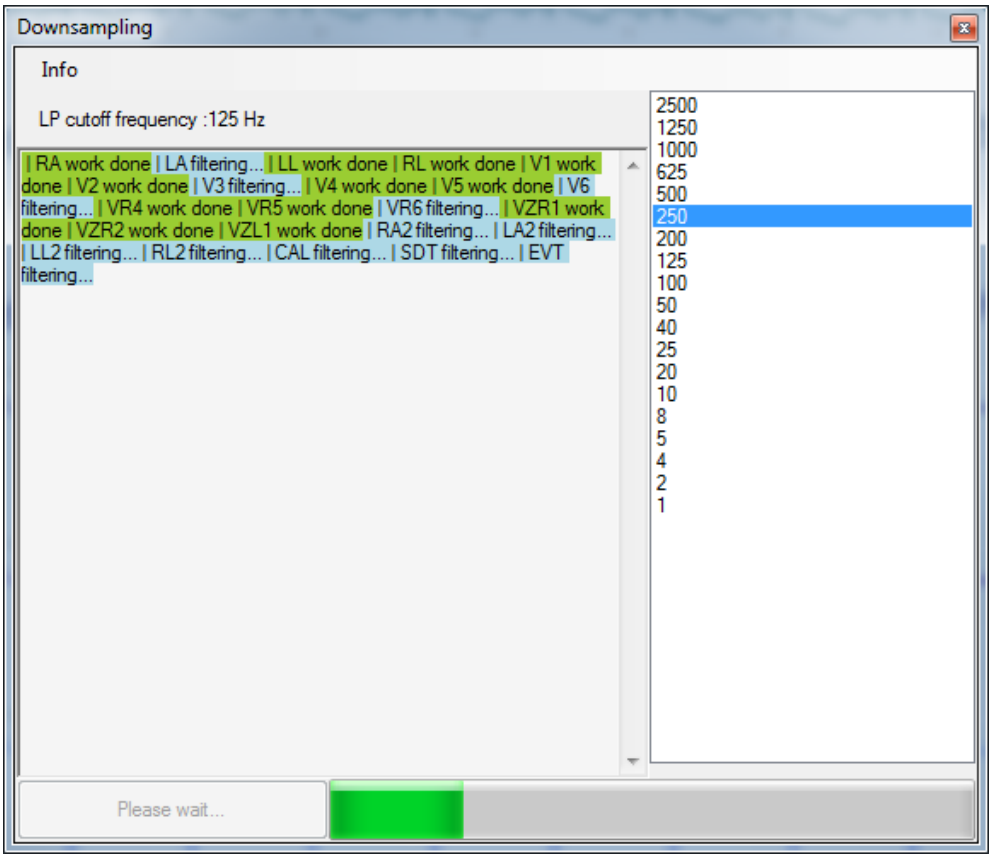


Downsample

This plugin provides low-pass filtering and decimation in order to downsample all signals.

Usage

Use target frequency, press PROCESS button and wait. Signals waiting to process are represents in gray, just processed signals are blue and already downsampled signals are green.



Example command for batch processing

```
DOWNSAMPLE 1000
```

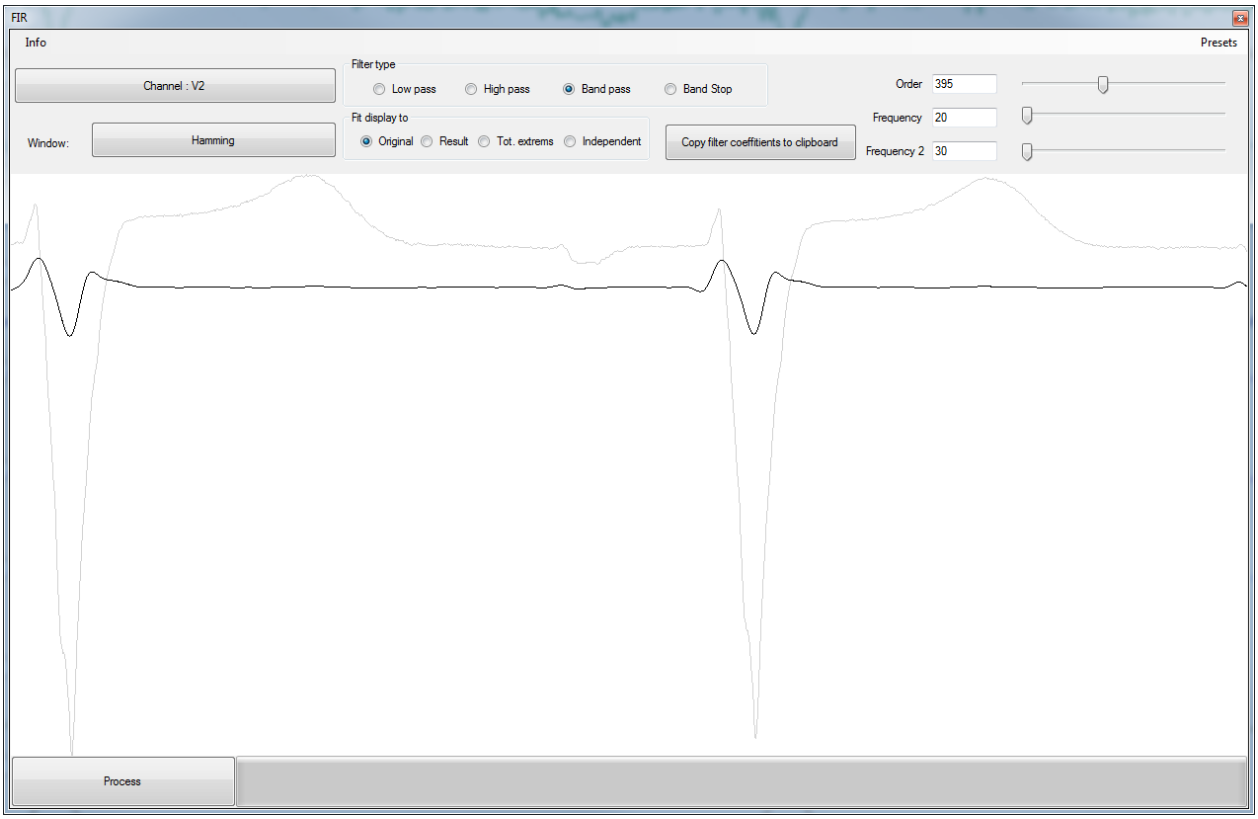
provide downsampling to 1000 Hz. Final frequency must be whole-number division of sampling frequency and must be lower or equal to Fs/2.

FIR filter

Finite infinite filter process data using n-th order FIR filter. Low-pass, high-pass, band-pass and band-stop filters are available. Alghoritm is inspired by book „Digital Signal Processing Tutorial for Braindead“ by Olli Niemitalo.

Usage

Set the filter type, order and frequencies (Fig-top). You can copy filter parameters to Windows clipboard using button COPY FILTER COEFFICIENTS TO CLIPBOARD. Window is used to compute weights for individual sample.



Example command for batch processing

`FIR CHANNEL(V3) PARAMS(BP;50;20;30)` filters V3 by band-pass in 20-30 Hz, 50th order.

`FIR CHANNEL(V2) PARAMS(LP;40;25)` filters V2 using low-pass at 25 Hz with 40th order FIR

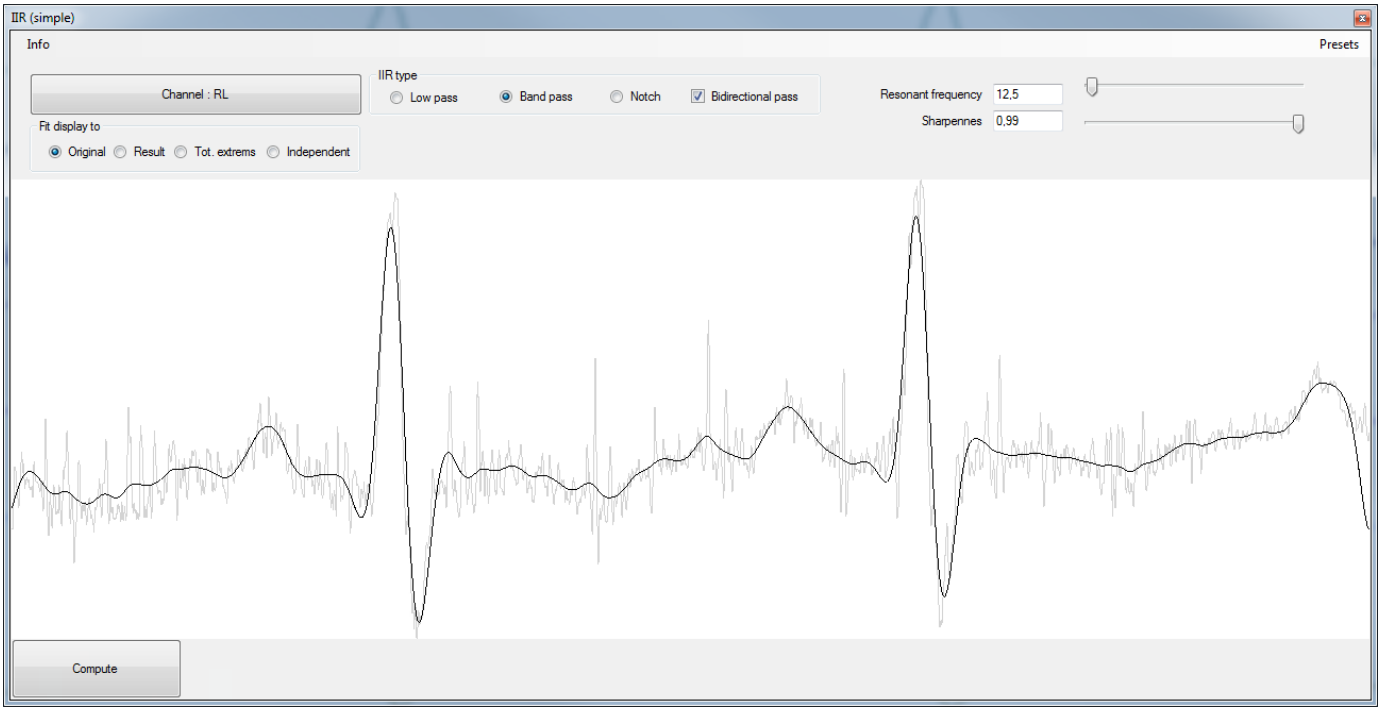
`FIR CHANNEL(ECG) PARAMS(HP;30;10)` filters ECG using high-pass from 10Hz, 30th order FIR

IIR filter (simple)

IIR filter is the fastest type of available filters, but with lower functionality. 2nd order IIR type is used and code is inspired by book „Digital Signal Processing Tutorial for Braindead“ by Olli Niemitalo.

Usage

This 2nd order IIR filter is controled by resonant frequency and sharpness. Because IIR filters (by their nature) shift filtered signal in processing direction, use can use BIDIRECTIONAL switch to avoid this shift.



Example command for batch processing

`IIRFAST CHANNEL(V3) PARAMS(BP;12.9;0.5)` filters V3 by band-pass in 12.9 Hz, sharpness 0.5

`IIRFAST CHANNEL(V2) PARAMS(LP;40;0.1)` filters V2 using low-pass at 40 Hz, sharpness 0.1

`IIRFAST CHANNEL(V2) PARAMS(NOTCH;50;0.8)` remove 50 Hz from V2 channel, sharpness 0.8

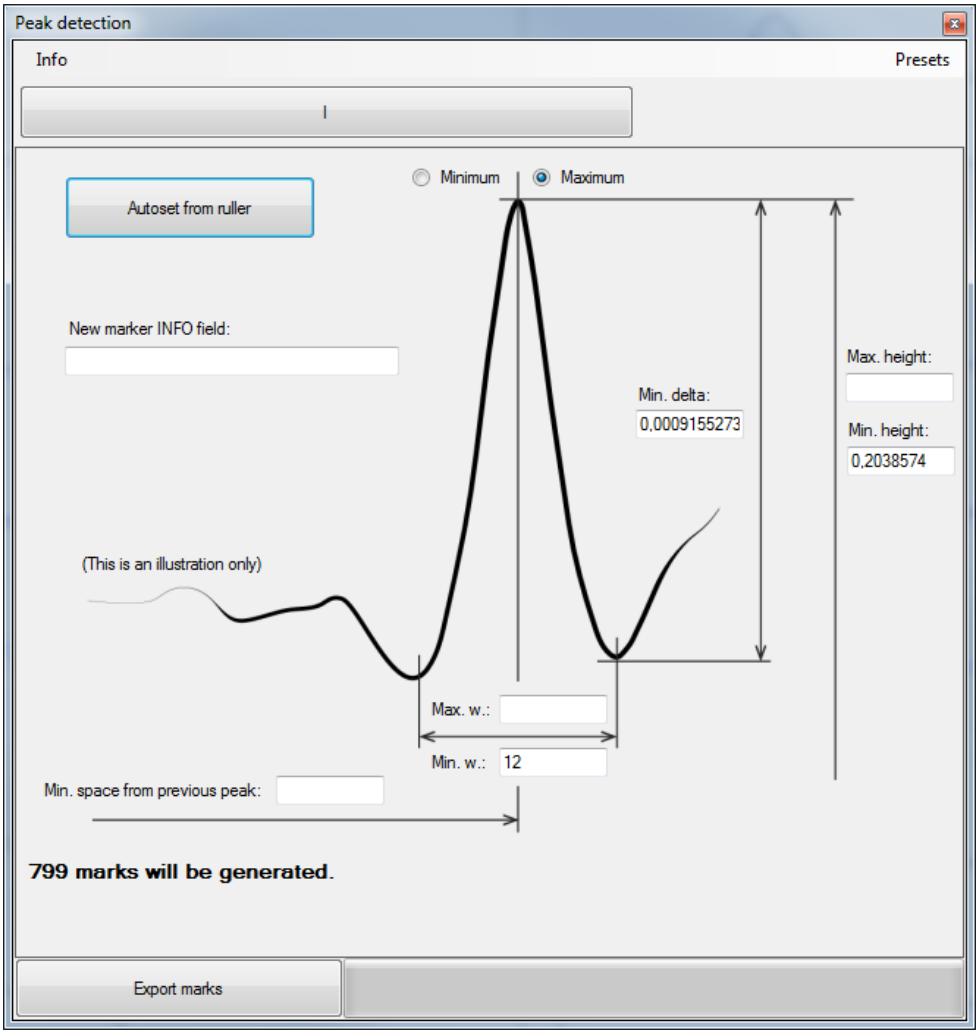
Peak detection

plugin si able to detect peaks and create marks in those positions.

Usage

Link channels to button on the top and define peak properties in plugin window. It is not necesarry to set all available parameters. Number of resultant marks is shown in bottom-left location. INFO field sets „info“ parameter of newly created marks.

Alghoritm can „learn“ peak parameters using cursor position. Place the cursor (CTRL+LMB) just **before** the peak and press AUTOSET FROM RULLER button (any channel must be linked to enable this functionality).



Example command for batch processing

```
FINDPEAK CHANNEL(V1) TYPE(MIN) MIN_HIGHT(0.4) MIN_DELTA(0.2) MIN_WID-
TH(200) MAX_WIDTH(500) MIN_DISTANCE(1000) INFO(PEAKS)
```

searches channel V1 for local minimina of min.height 0.4, minimal distance from peak to root 0.2, of width in range between 200-500 and at least 1000 samples of the last peak. Each generatet mark will be signed as „PEAKS“.

QRS detector

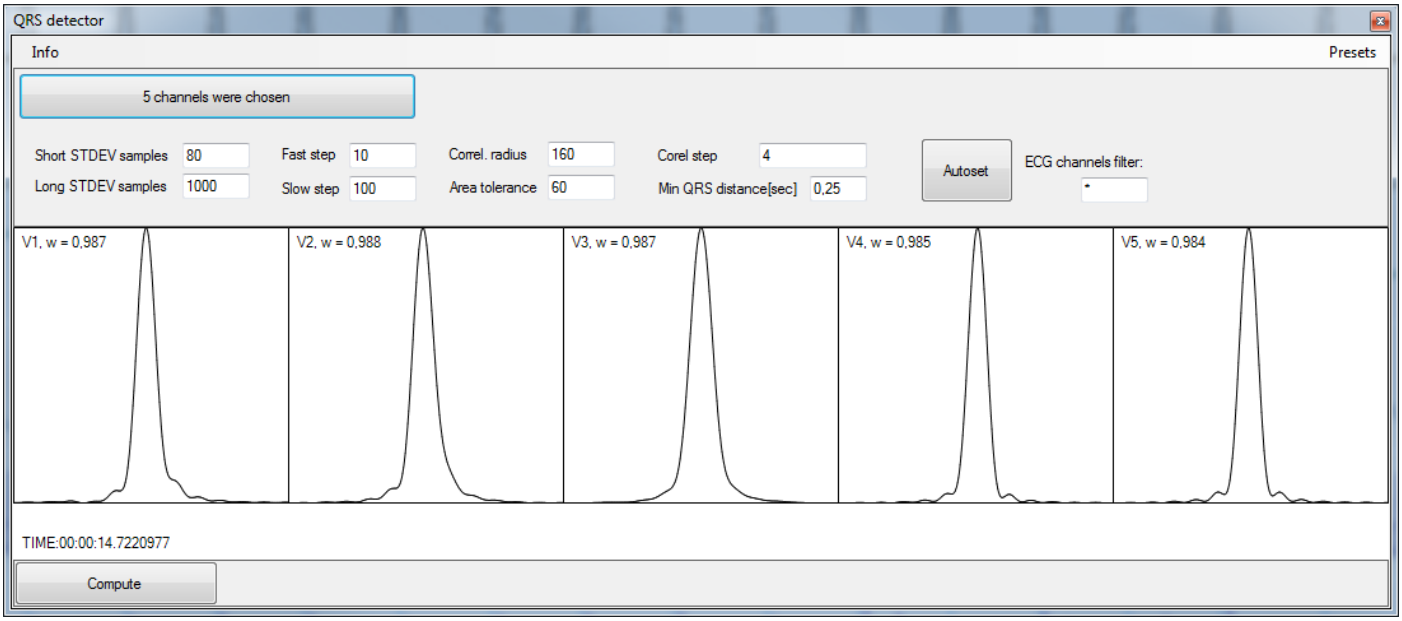
is intended to use with ECG signals to detect QRS in multimodal data, where at least one channel is ECG. It uses our approach from CinC Challenge 2014 (Plesinger, F.; Jurco, J.; Jurak, P.; Halamek, J., „Robust multichannel QRS detection," in Computing in Cardiology Conference (CinC), 2014 , vol., no., pp.557-560, 7-10 Sept. 2014).

Usage

We suggest that ECG channels (for human subjects) are converted to amplitude or power envelopes in range 12-30 Hz prior to QRS detection.

Link any channels to plugin (for human subjects the sufficient number lies between 3-6 channels). You can use only 1 channel, but then the resistence against noise will be greatly limited.

All parameters are auto-set using sampling frequency. If your ECG subject is not human you need to use different settings (for example mice subjects need all parameters to be approx. 6x times lower and power envelopes should be derived from frequency range 40-100 Hz).



When parameters are set, press COMPUTE button. There will be typical sample of QRS displayed in plugin window as wall as weight of that typical sample (there are ECG power envelopes in the figure) . Also, new marks in QRS positions are created.

Example command for batch processing

`SHAPEFINDER` Finds QRS using all channels, expecting all channels are ECG.

`SHAPEFINDER CHANNEL(V)` Find QRS in human ECG from all channels including V in name (V1-V6)

```
SHAPEFINDER CHANNEL(*) ECGSOURCE(V) PARAMS(15;150;2;20;30;12;2;0.1)
```

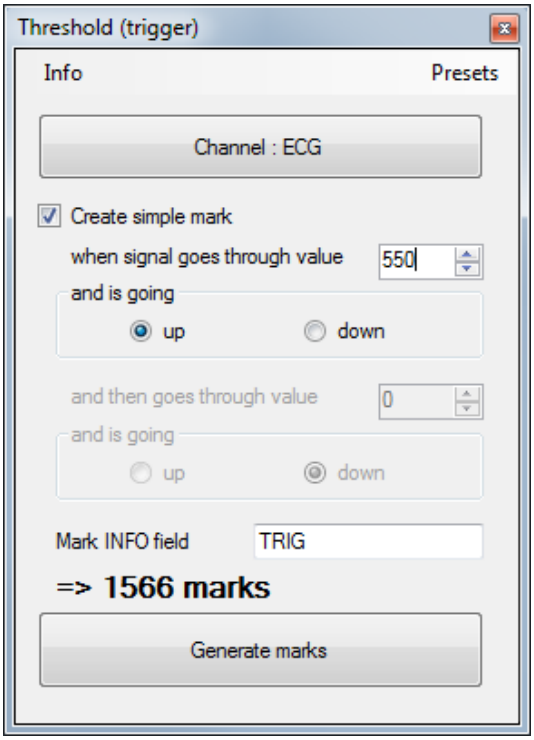
uses all channels from record, defining any channels with „V" in name as V1-V2-V3 etc as ECG channels. Eight parameters are as in Figure - Short st. dev samples, Long st. dev samples... to min. QRS distance.

Threshold (trigger)

is intended to generate marks using input channel as trigger.

Usage

CREATE SIMPLE MARK checkbox sets if the newly created mark will react to only one threshold value or to two threshold values. Set the threshold value in numeric box. In both cases,the user can set if reaction to the threshold value should be accepted when signal si growing (i.e. UP) or decreasing (DOWN)

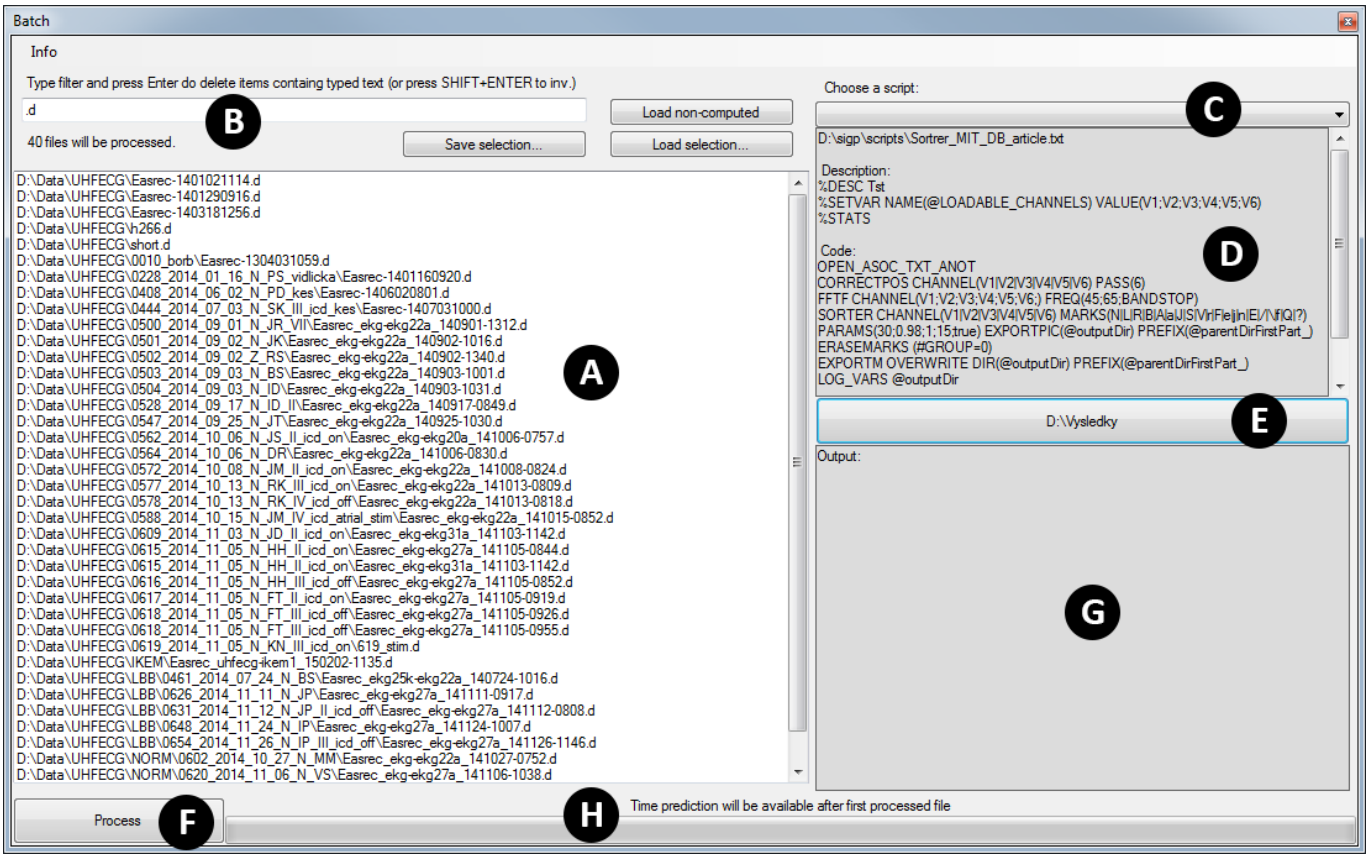


Batch

It is useful to automatize repetitive operations when working with large number of files. For this reason SignalPlant allows to run scripts on groups of files. Script consists of commands, registered by plugins. More about SignalPlant scripts is written in appendix B: **Scripting**.

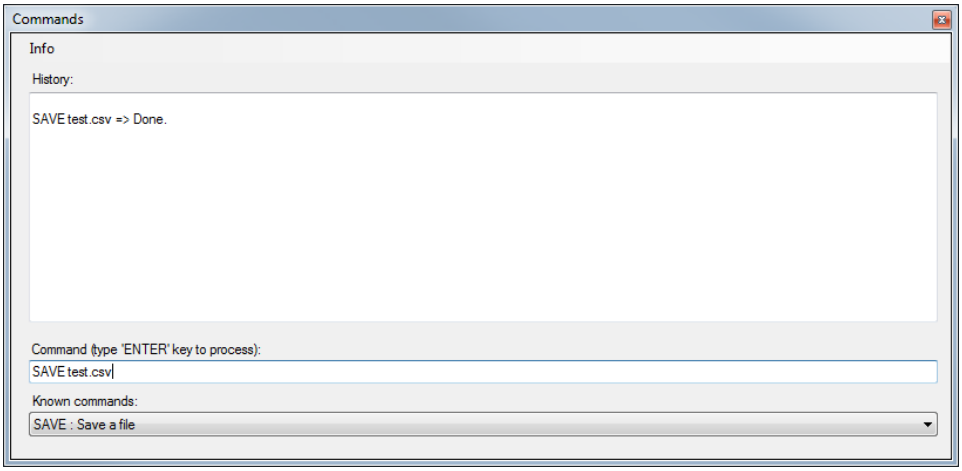
Usage

First, create script and place it into SCRIPTS subdirectory (see **Settings** chapter to find out that location). Open Batch plugin and drag files or directories into large area for files Fig-A. Filter out files you do not want to process using fitlering text box (Fig - B). You can save final list into text file for later use. Now choose the script (Fig - C), it will be listed in preview area Fig - D. Set output directory (Fig - E) and press PROCESS BUTTON (Fig - F) to start batch processing. Text output from individual commands is printed to output area Fig-G. Overall progress is shown using label and progres bar Fig-H.



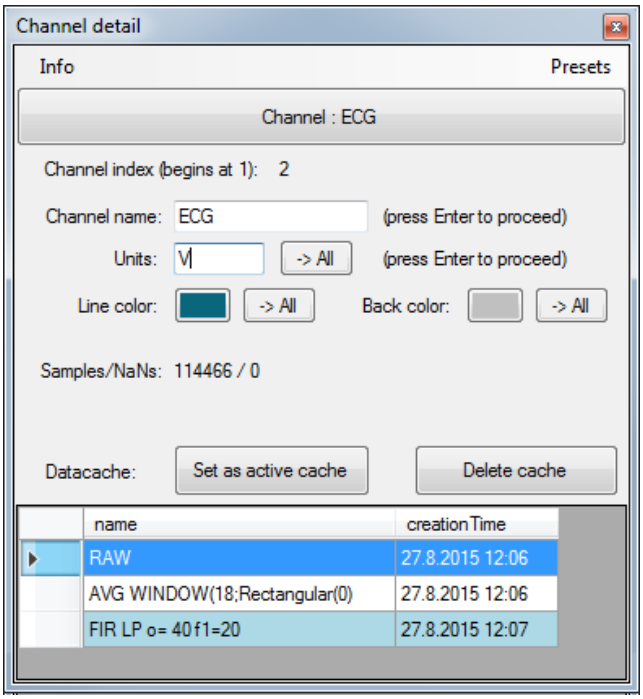
Commands

allows to execute single commands. It useful when building scripts to test the whole command sequence.



Channel detail

This plugin is called whenever the user double clicks on any channel. It allows to rename it, rename its unit, change foreground and background colors or apply them to other channels. Furthermore, it shows all available datacaches and allows renaming or deleting them.

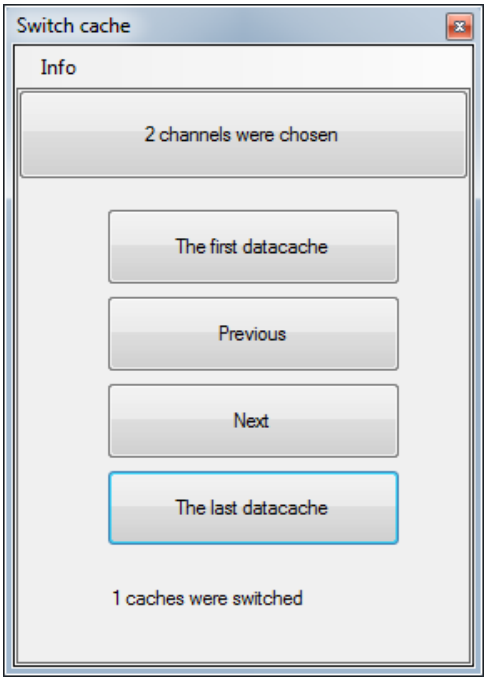


Usage

Open the channel detail window. After modifing text properties press enter to proceed. To delete datacache, select it in table at the bottom and press DELETE CACHE button.

Switch cache

allows to switch selected channels to specific datacache, especially useful when working with large number of channels (for example with EEG signals).



Example command for batch processing

```
SETCACHE CHANNEL(V1) PARAMS(0)
```

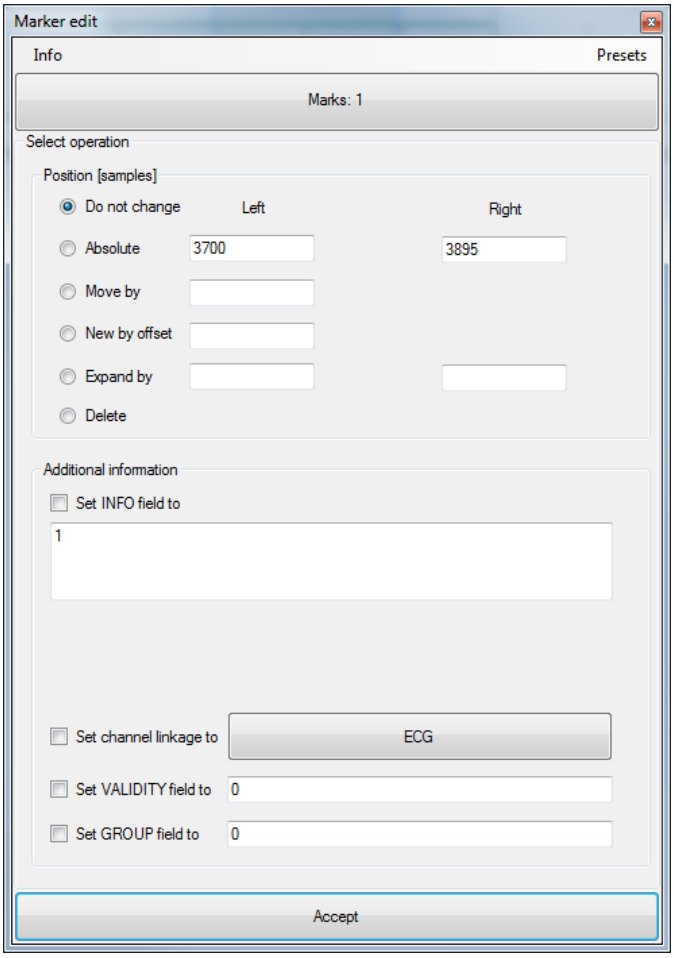
sets channel V1 to the first datacache.

Marker edit

is called whenever user double clicks on a mark or when defining new mark using SHIFT+LMB. It is intended to change any of mark properties to one or more marks.

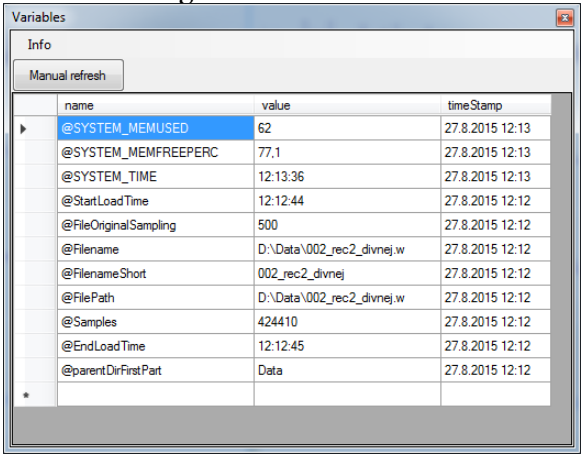
Usage

Open the plugin and set involved marks using the button on the top. If the plugin is opened by double click on mark, then the plugin is already linked to it. Choose the operation and press ACCEPT buton.



Variables

plugin shows existing SignalPlant variables, which are usualy used for loging system information during batch processing. Any plugin (and user) can insert any new variable. Some Variables are defined when file is loaded into SignalPlant.

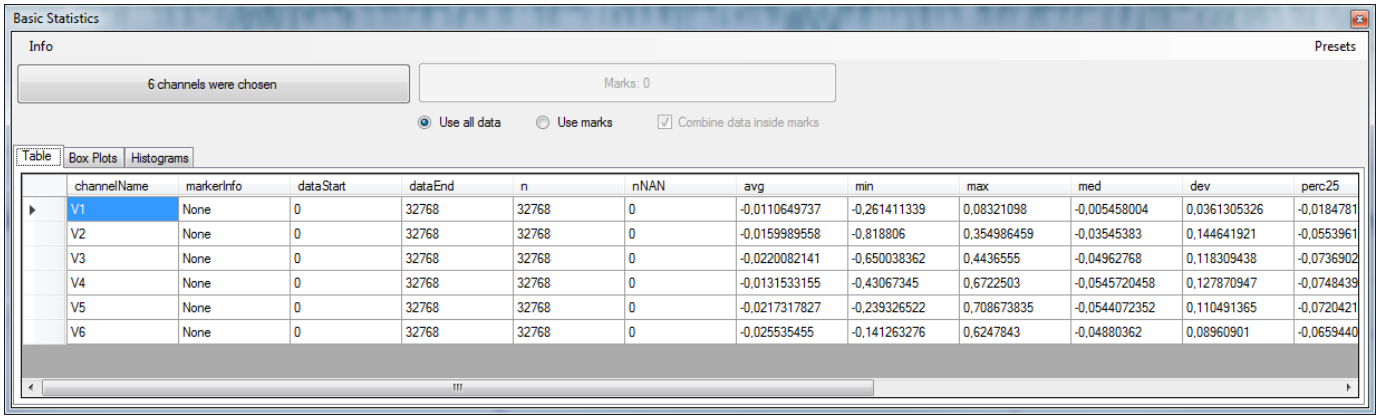


Basic statistics

plugin allows to find out descriptive statistics parameters of selected channels or their parts. It shows results in table, box-plot or histograms.

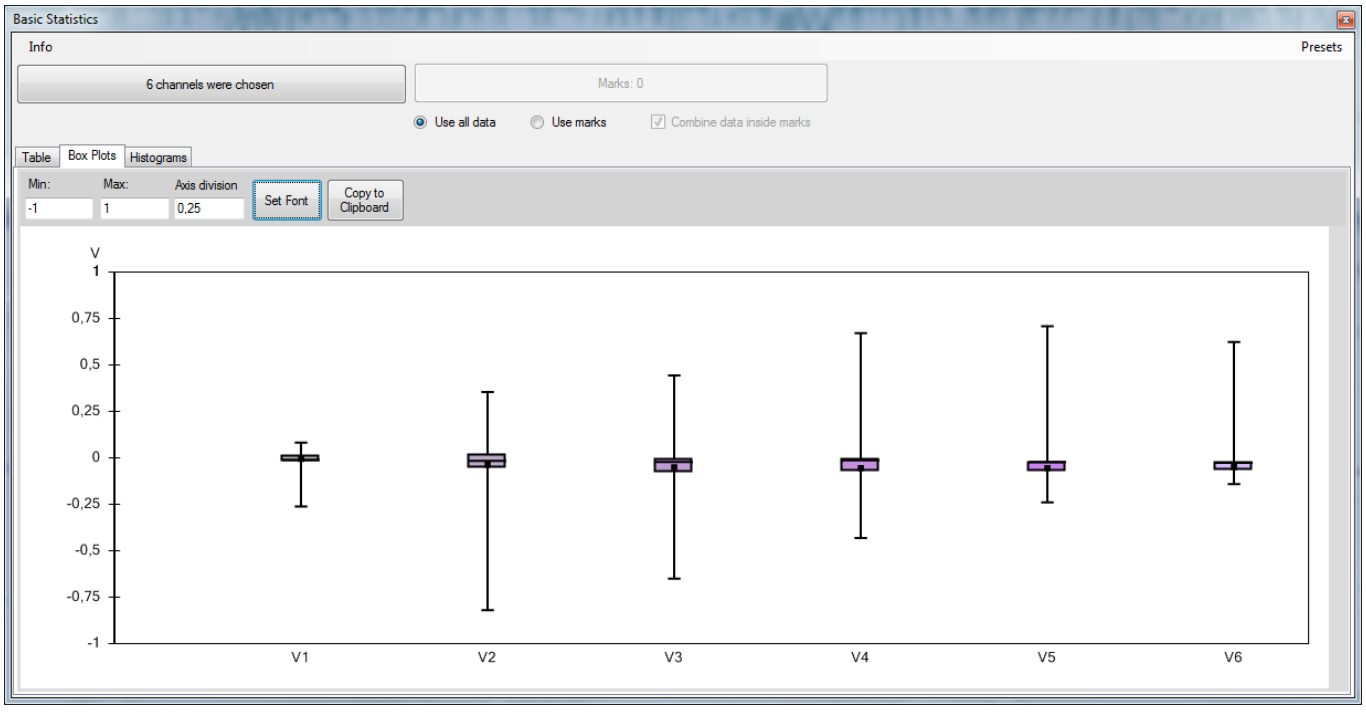
Usage

If you need to find out parameters of all signals, just open the plugin and attach individual channels into it. Statistic properties are computed (in parallel) and shown in table.



channelName	markerInfo	dataStart	dataEnd	n	nNAN	avg	min	max	med	dev	perc25
V1	None	0	32768	32768	0	-0,0110649737	-0,261411339	0,08321098	-0,005458004	0,0361305326	-0,0184781
V2	None	0	32768	32768	0	-0,0159989558	-0,818806	0,354986459	-0,03545383	0,144641921	-0,0553961
V3	None	0	32768	32768	0	-0,0220082141	-0,650038362	0,4436555	-0,04962768	0,118309438	-0,0736902
V4	None	0	32768	32768	0	-0,0131533155	-0,43067345	0,6722503	-0,0545720458	0,127870947	-0,0748439
V5	None	0	32768	32768	0	-0,0217317827	-0,239326522	0,708673835	-0,0544072352	0,110491365	-0,0720421
V6	None	0	32768	32768	0	-0,025535455	-0,141263276	0,6247843	-0,04880362	0,08960901	-0,0659440

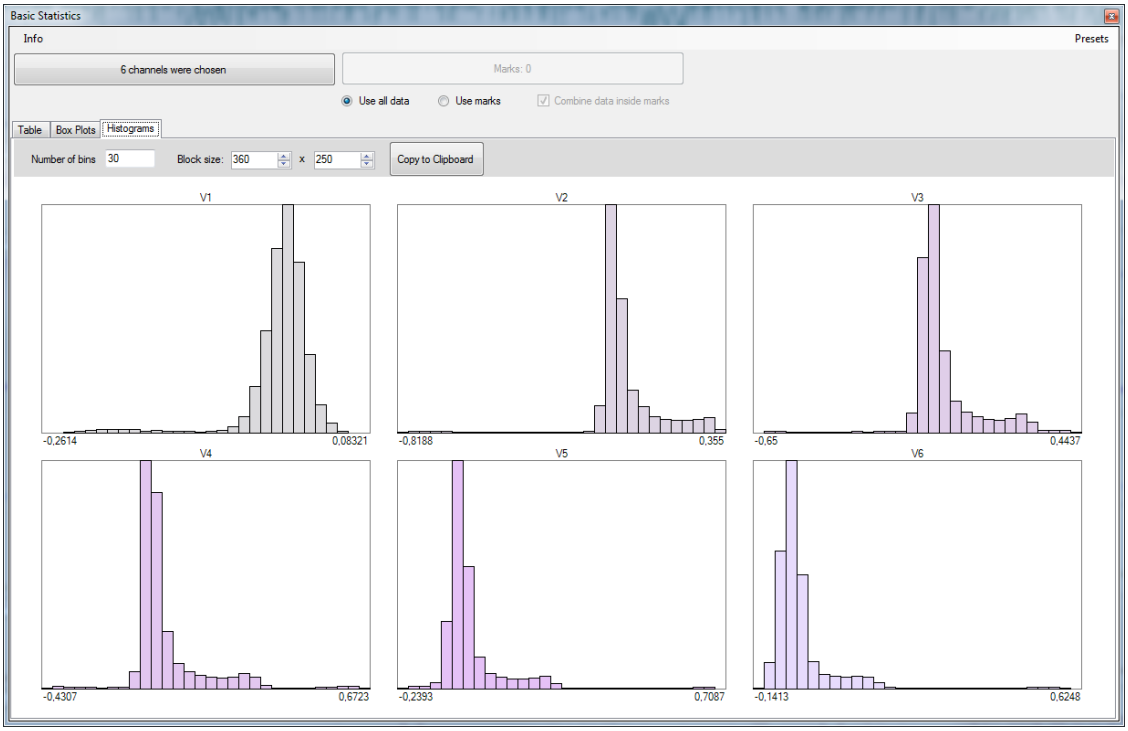
Contents of the table can be selected (top-left empty cell selects all the table) and copied using CTRL+C. Box plots tab switch to different representation of computed statistical properties:



If you need to find out statistical properties of specified area(s) in channel(s), then at first create marks over those areas of interest and set their INFO field for better results recognition. Use the Marks button (in Fig it is disabled, because there were not any marks in project) to refine mark selection.

Plugins are interactive, so you can see changes in boxplots/ histograms while you change linked mark.

Finally, histograms can be produced to show values distribution in selected channels:



Pearson correlation

plugin allows to compute correlation between selected shape (by mark) and specified channel. Shape can originate from any channel.

Usage

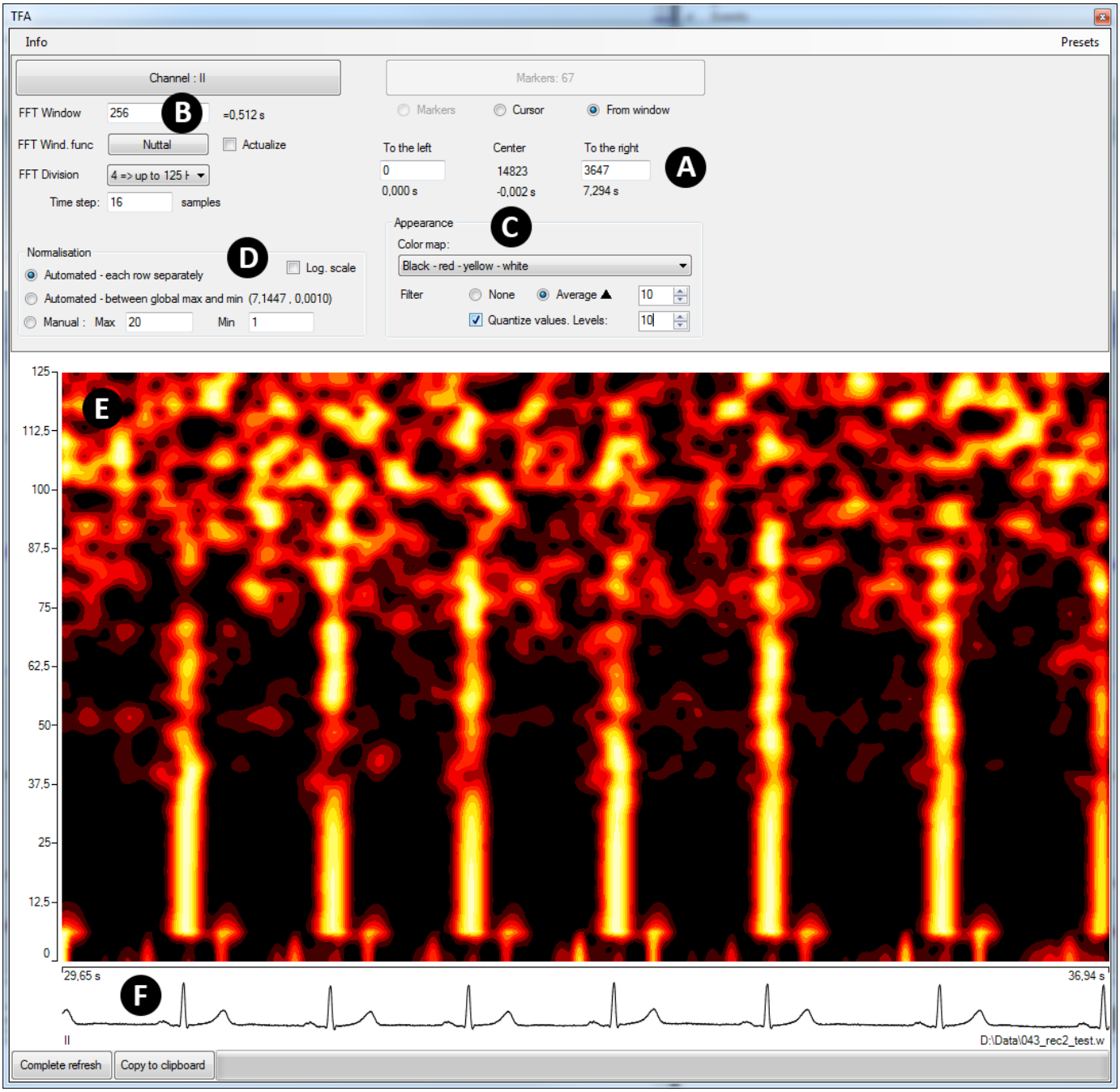
Create mark describing desired shape. Then open Pearson Correlation plugin and link the channel where you will seek the shape (Fig - A) and set prototype channel and mark (Fig - B, C). The prototype shape will be displayed in shape preview area D and correlation preview (for currently displayed time range) will be shown in viewport E. Set if the correlation result will be datacache or new channel (Fig - F) and start process (Fig - G).

TFA - Time Frequency Analysis

is tool showing continuous signal in frequency domain when X axis represents time and Y axis represents frequency. It usefull when analysing frequency range of some event as shown in figure with human QRS, starting above 8Hz and with visible traces up to 85 Hz.

Usage

Area to compute TFA is (by default) taken from visible window in SignalPlant, but can be overriden (Fig-A). Set FFT window size (Fig - B). Usually, the window size should be shorter than period between examined events. FFT division defines the top limit of displayed frequency range. Time step defines period between individual FFT analysis. Time step=1 means that FFT will be computed for each sample.
User can choose different color maps (Fig-C) installed with SignalPlant and switch on smoothing (Averaging) and/or quantization to improve visual apearance of resultant image.
Three modes of value-to-color normalizaiton are available (Fig - D): Automated - each row (normalize each row separatly), Automated between global maximum and minimum and Manual. The source signal is shown at the bootom (Fig - F)

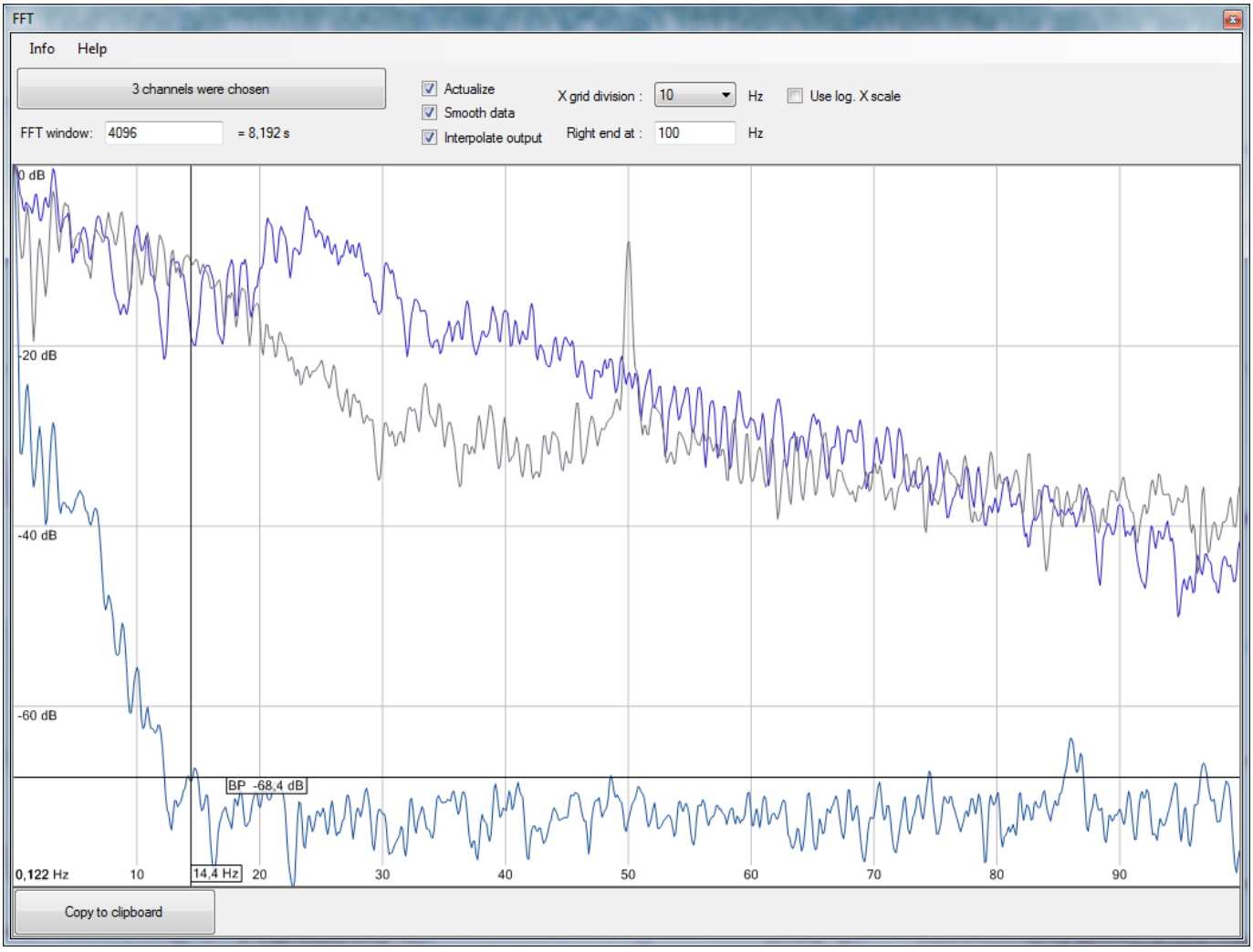


FFT - Fast Fourier Transform

shows FFT analysis to selected number of channels from specific time window. Values are converted to dB.

Usage

FFT window starts at the current **ruller** position and it size is defined at the top-left corner of the plugin window. It is posiible to choose between linear and logaritmic scale (top-right most checkbox).
User can measure FFTs using LMB.

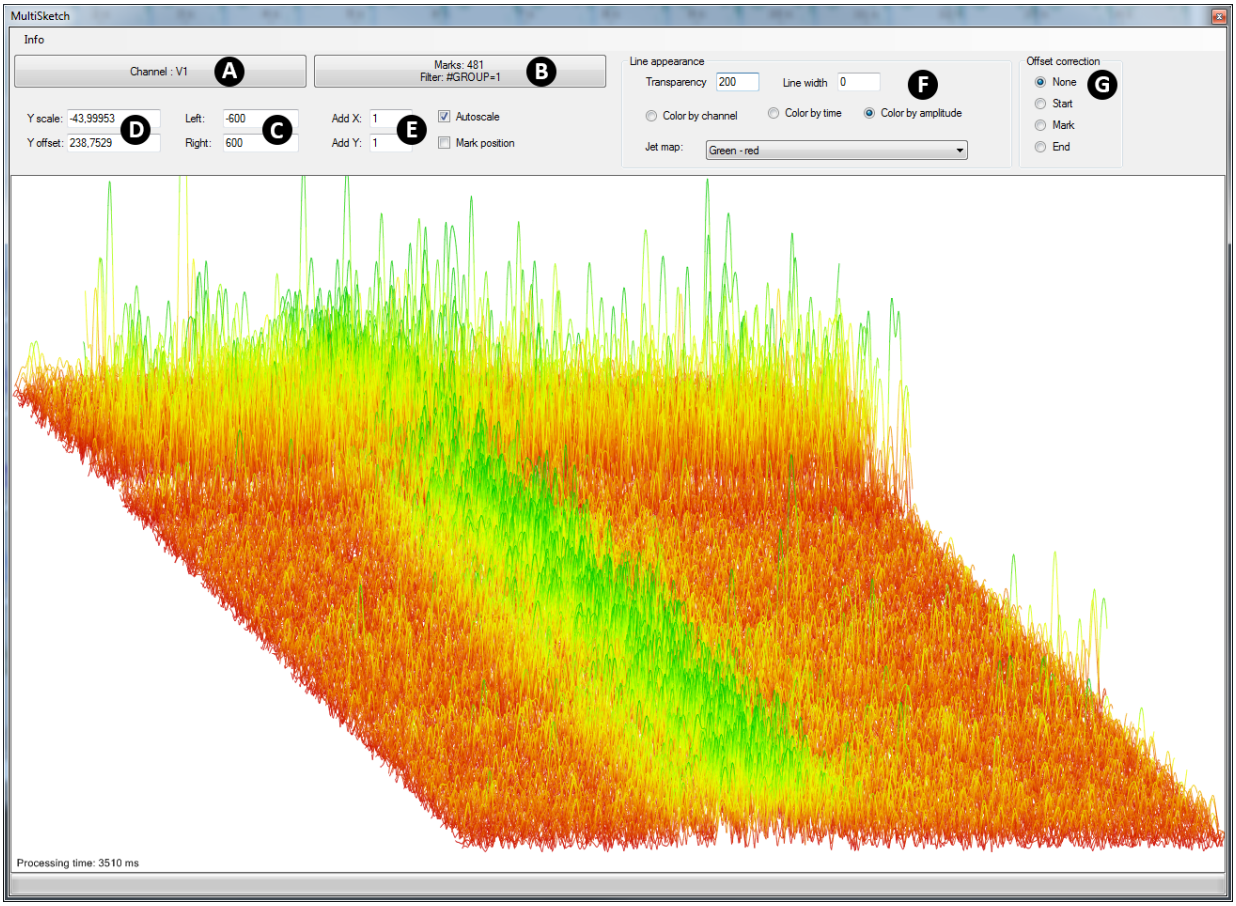


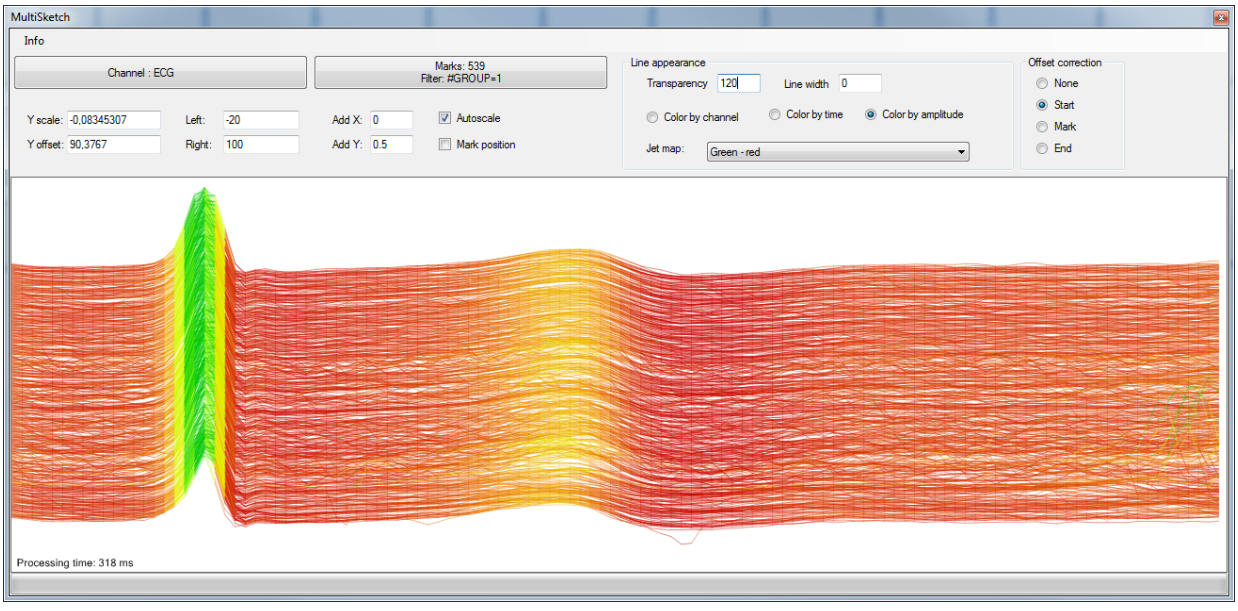
Multisketch

draws epchos from one channel over themselves, using marks as triggers.

Usage

Select channel (Fig-A) and triggering marks (Fig - B). Area before and after each mark is defined by textboxes in Fig-C and Y-scale and offset is defined in Fig - D. Offset can be overriden by mouse drag in the image. Each epoch can be shifted in X and Y values to produce 3D-efect (Fig - E). Line width, transparency and color can be set using controls in Fig - F and line offset mode can be set in Fig-G.



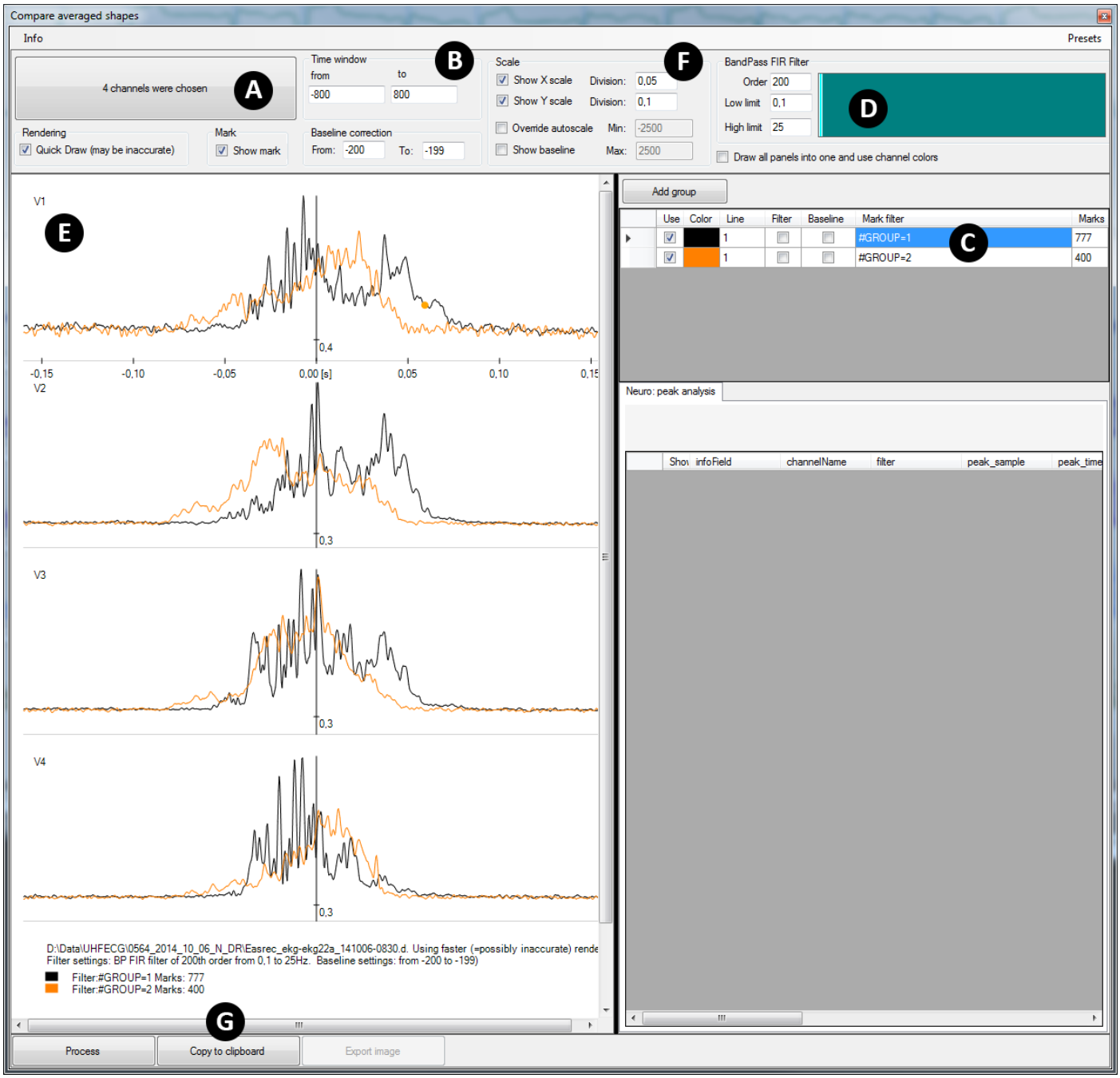


Compare averaged shapes

is sofisticated tool for signal accumulation (averaging). For example, using this plugin it is possible to compare 500-1000Hz amplitude envelopes, accumulated through large number of previously categorized QRS. User can see results in several leads together. **Marks** are used as triggers. This tool alows SAECG processing.

Usage

Attach channels using BUTTON (Fig - A). Signal will be accumulated in predefined area to the left and right from each mark (Fig - B). To define group of marks as triggers, click on „Mark filter" cell (Fig - C) and in **quering dialog** set proper group of mark. Accumulated signal can be filtered using FFT band-pass filter (Fig - D) and base-line corrected. Resultant image (Fig-E) can be scaled by dragging the x-axis up or down. Scale and labels apearance can be modified by controls in Fig- F. If you need to copy picture, use COPY TO CLIPBOARD button (Fig - G).

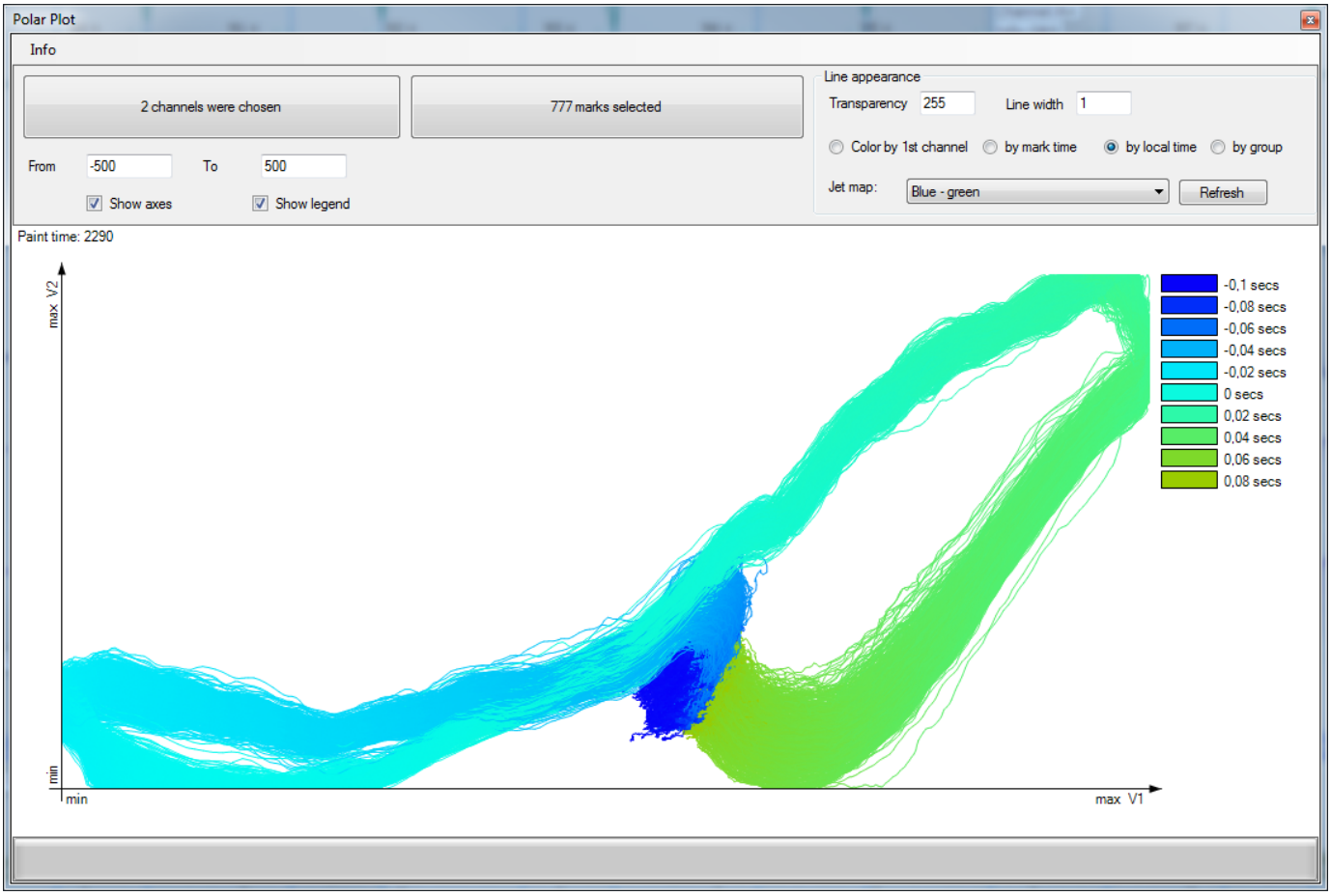


Polar plot

is plugin to display signal from to channels when the first acts as X-coordinates and the second as Y-coordinates source.

Usage

Select just two channels and drag them to CHANNELS button. Select marks, defining triggers and set the time window (below the channels button). Every iteration is normalized to minimum and maximum values. In the Line appearance box, you can set color mode.

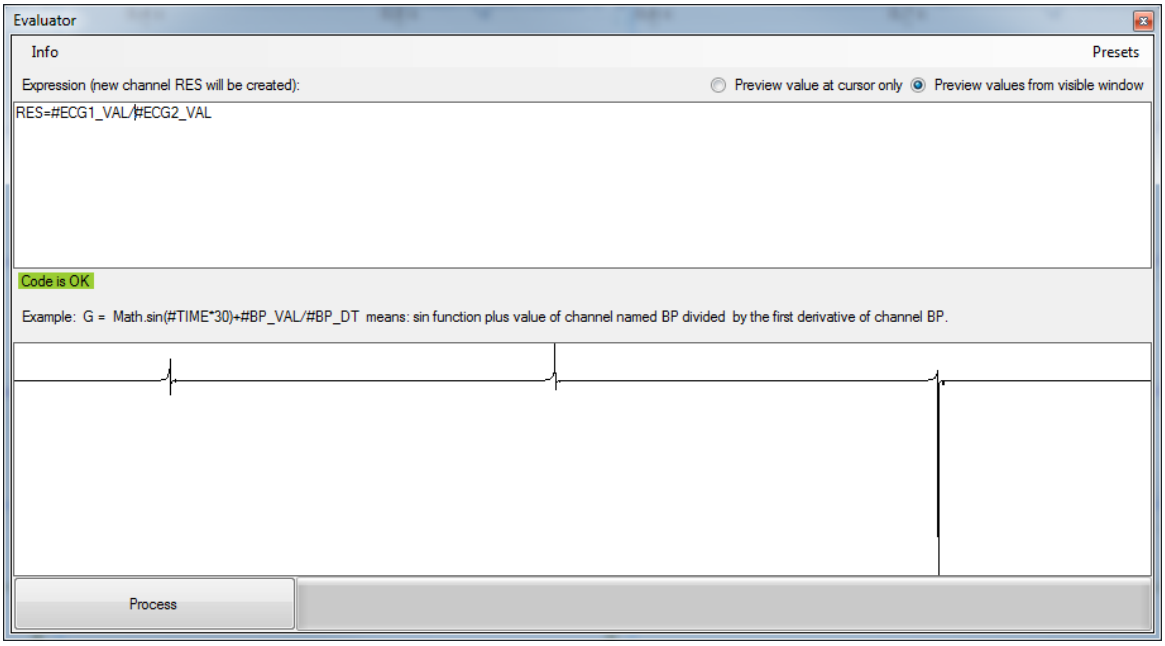


Evaluator

Plugin „Evaluator“ is intended for computing values of new channel, based on an equation. Mathematical expressions must respect JAVAscript language (except SignalPlant reserved words)

Usage

Write the equation into the top part of plugin, beginning with name of new channel. Code validity is checkech by green or red (if code is invalid) label below. The result preview is shown in bottom part of plugin window.



Reserved words always begin with #. They are:
#(Name of channel)_VAL: value of a given channel (use drag & drop instead of writing)
#(Name of channel)_DT: the first derivative of a given channel, computed using time
#(Name of channel)_DN: the first derivative of a given channel, computed using sample index
#TIME: time in seconds from the file start
#SAMPLE: sample number
#TRACE_TIME: cursor position in seconds
#TRACE_SAMPLE: cursor position in samples

Samples:

RES=#ECG1_VAL/#ECG2_VAL	Create channel named RES with division of ECG1 by ECG2
DV=Math.abs(#ECG1_DT)	Absolute values of the first derivative of ECG1
SINUS=Math.sin(#SAMPLE*0.02)	Draw sinus function with scaled argument

To compute large number of channels (e.g. when mounting) use **Mount** plugin.

Example command for batch processing

```
EXPRESSION RES=#ECG5_VAL*#ECG6_DT
```

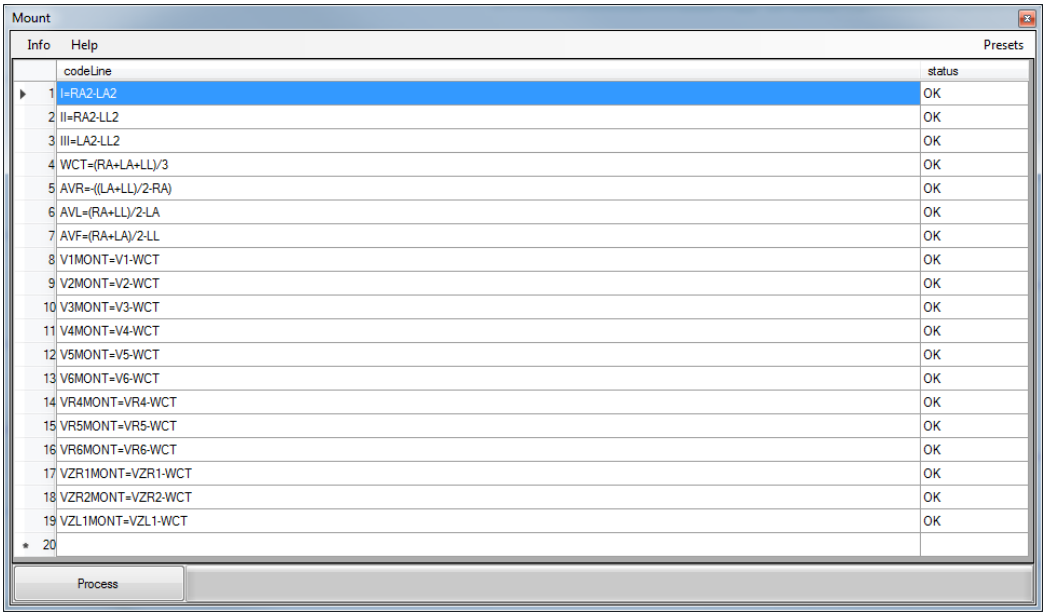
creates new channel name RES, holding multiplication of ECG5 and ECG6 derivative.

Mount

This plugin provides basic mounting operations, i.e. mounting of ECG AVR/AVL/AVF to ECG I/II/III.

Usage

User must define equations for each channel to be mounted. Allowed operators are + - * /; also simple brackets ( ) can be used. After editing each code line is validated. If it is ok, „status“ column on the right is green, else it is pink. Proces is started by pressing „Process button“. Note that this button is enabled only when all lines are accepted as correct.



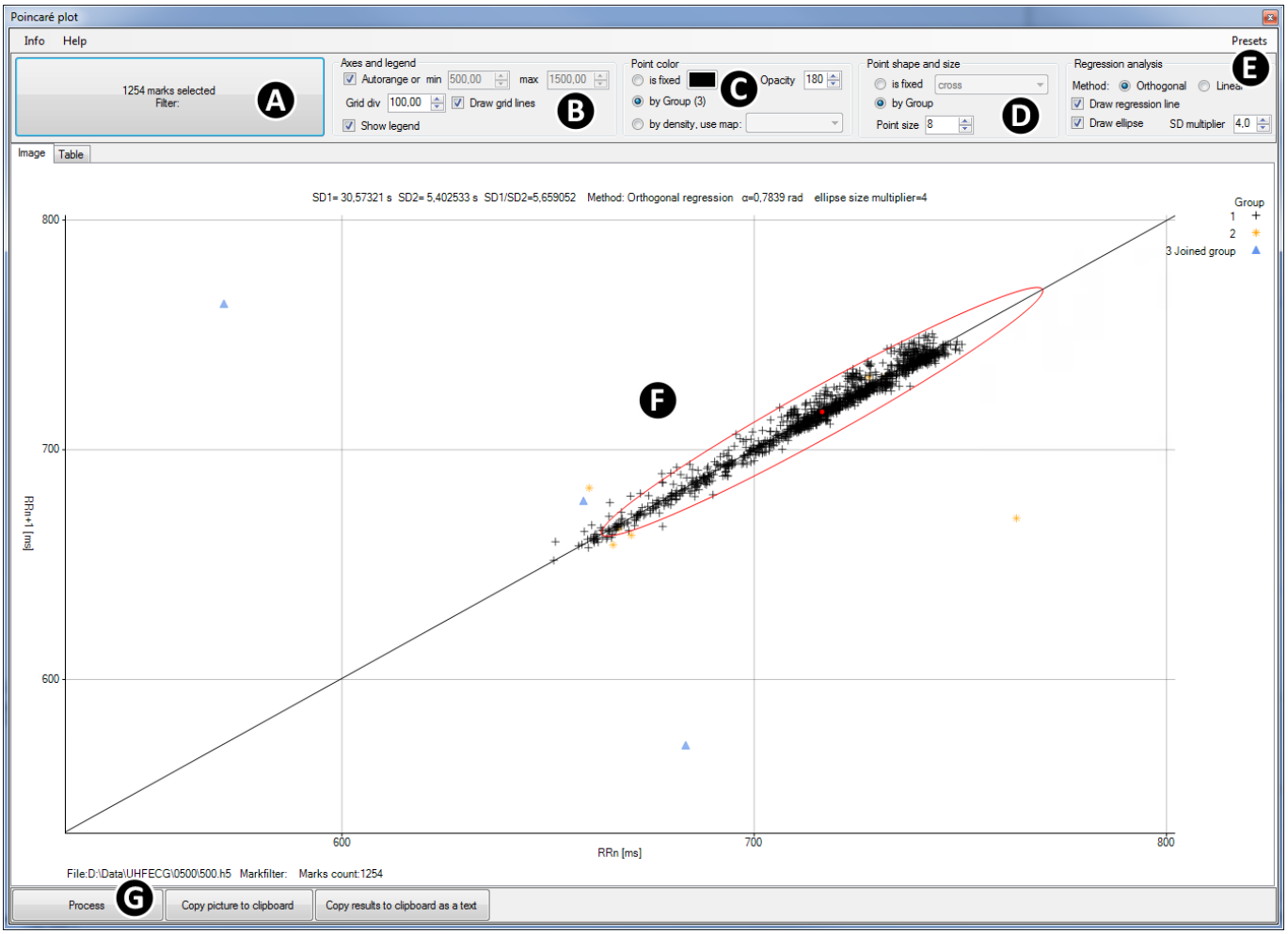
Processing is done assynchornously in parralel. All channels which can be computed are processed in the first itteration (FIG lines 1 to 7). Newly computed channels are added and if there are channels to be computed, next iterations are made (lines 8 to 19 will be computed simultaneously in the 2nd iteration). Obviously, processing speed will be dependant on number of available cores.

Poincaré plot

Poincaré plot is usually used for HRV analysis. It is kind of a „delay map“; the name „Lorenz plot“ is also used.

Usage

User must have **marks** defined (usually QRS marks if the HRV Poincaré plot is the goal). Marks selection may be refined using button A. Display properties as display range, grid division or legend visibility can be set using controls B. Point color can be defined in area C and their shapes and size in area D. Analysis is done automatically or manually using Process button (G). Two algorithms may be used - orthogonal (default) and linear regression. Coordinates of resultant points (F) can be copied from tab „Table“ (bellow marks Button - A).



AmpToColor

This plugin translates amplitudes of multiple channels to color and render it as an image. This approach allows displaying of large number of channels (usefull with EEG recordings) over given time period. In the case we are interested in event in specific frequency range it may be good to translate raw signals into amplitude envelopes in a given range (using **FFT filter** plugin).

Usage

Link channels into Channels button (A). Set the time range and method (B) and press the Process button (D). Resultant image (E) is colorized using predefined gradients („jet-map", C). Max and Min values for selected area is computed, but the scale can be overrided by changing „Sensitivity" value (F). Image can be copied to clipboard or saved as a file (to the right from D).

Please note that while using this plugin, you have to press PROCESS button (D) each time you change settings.

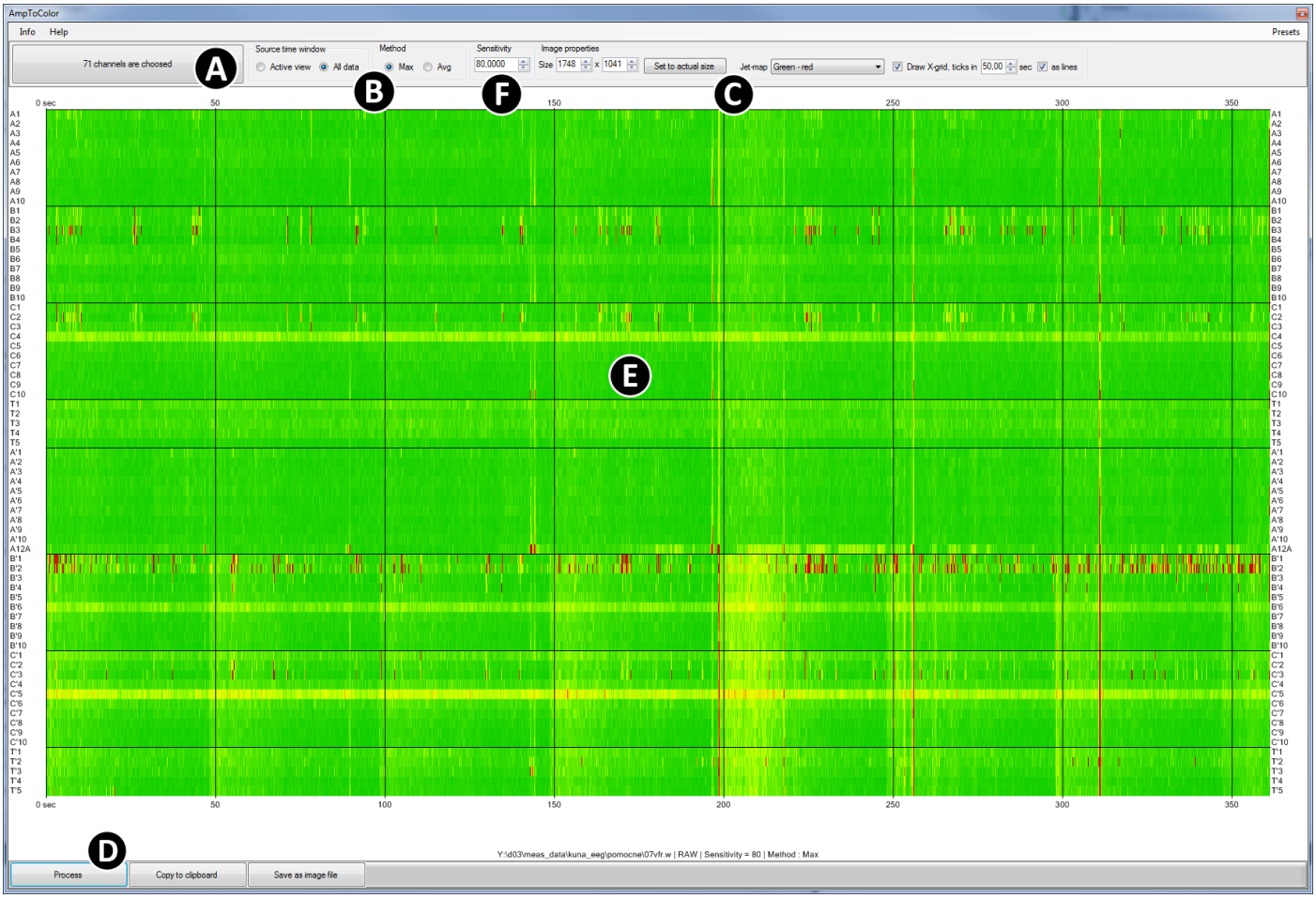


Figure shows EEG record of 72 channles at 5 kHz and length over 6 minutes. All channels were preprocessed using FFT filter - amplitude envelopes in range of 500-1000 Hz corresponding to „very fast ripples". This image shows such an activity in B´1, B´2 and B3/C2 leads.

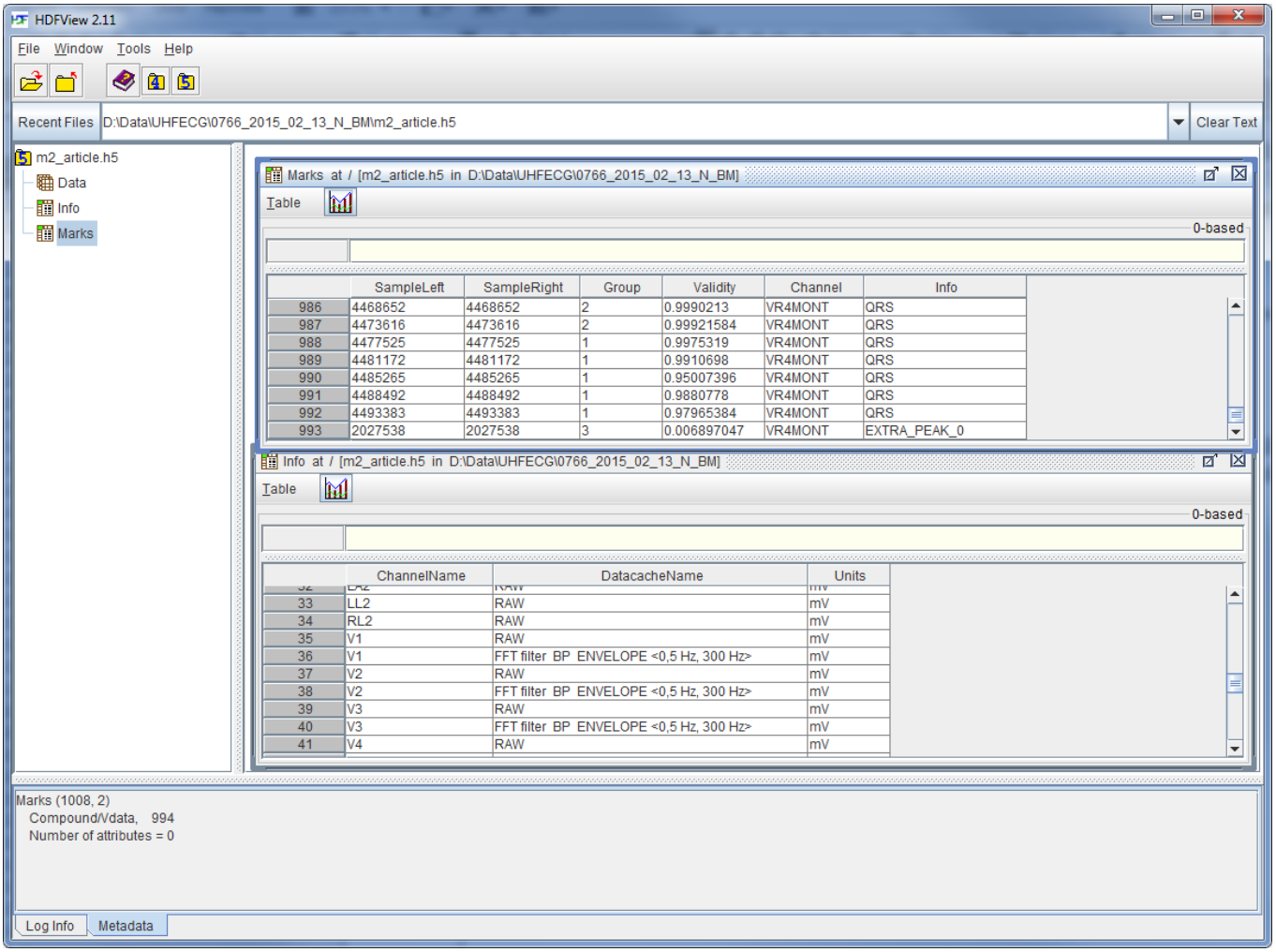
A - HDF5 files structure

For working with HDF5 files the HDF5FotNet library is used (**http://hdf5.net/**). HDF files are saved as „chunked“ files (see **https://www.hdfgroup.org/HDF5/doc/Advanced/Chunking/**). Thanks to chunking, you may achieve better I/O performance after resaving file from other source by Signal Plant.

HDF data organization for files is as follows:

Root - each HDF file has its base named ROOT. Sampling frequency and source are set as ROOT attributes.
|__ Data (dataset with 2D matrix of floats [channels, samples])
|__ Info (dataset describing channel names and units)
|__ Marks (dataset describing marks)

We strongly encourage users interested in inner HDF5 files structure to load SignalPlant h5 file into free HDFVIEW (FIG) - **https://www.hdfgroup.org/products/java/hdfview/** and investigate the file‘s strucure directly. In that tool its possible to find exact description of use datatypes.
For managing HDF files SignalPlant uses classes, provided by HDF Group.



B - Matlab -> SignalPlant

SignalPlant is able to load data saved in Matlab(r). Please note that you must have Matlab installed on the target computer. Data must be in form of 2D matrix of float or double; complex number cannot be read.

The simplest way to transfer raw data without channel names, units and sampling frequency is to save the record as a 2-dimensional matrix. The name of variable with data is unimportant now, it is preselected by its type. Let us say that we hold 4-channels record of 75 000 samples in 2D matrix named Signal [4,75000]. Save:

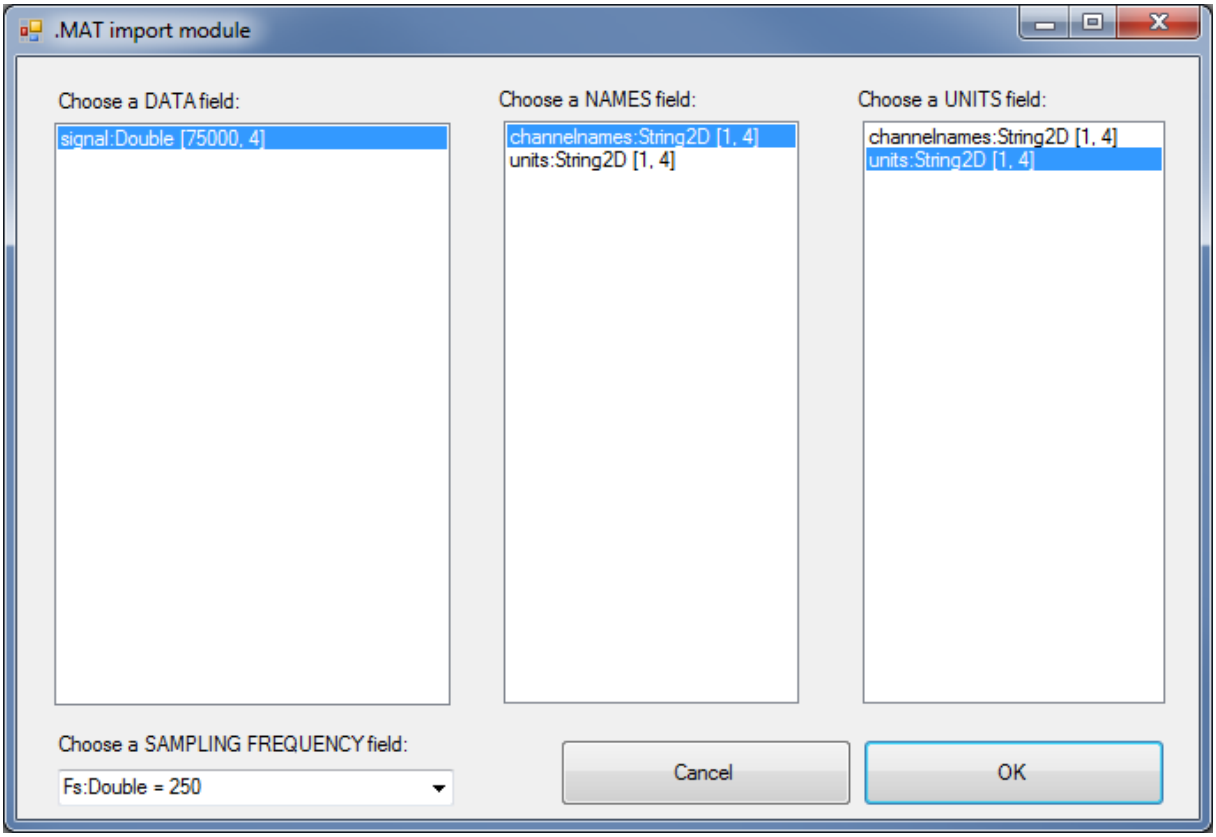
```
save('D:\signalsSimple.mat','signal')
```

and load it into SignalPlant. SignalPlant will ask for sampling frequency.

If you need to transfer channel names, units or sampling frequency, prepare such variables:

```
names={'ECG_1','ECG_2','ABP','RESP'}
units = {'mV','mV','mmHg','mV')
Fs = 250
```

and save them to the file. Variables for names and units are preselected (FIG) by SignalPlant and it is easer for user to name them as „names“ and „units“. Variable for sampling frequency is preselected using its type; name is unimportant.



C - Scripting in SignalPlant

Scripts in SignalPlant are usually used for **batch processing** for apply the same command sequence to multiple files.

Structure

Scripts are defined in .TXT files where each row is one command and % signalize commented line. Some-commands are defined in SignalPlant itslef (as LOG_VARS for logging) or can be registered by plugins (as FFTF for filtering using FFT). You can see in this help file is specific plugins also register any commands.

Note 1 - first is command, folowing by space symbol and additional parameters. Which and how many paramters command may read depends only on the developer of the specific command.

Note 2 - %DESC signalizes that from this line the name of the script will be derived.

Note 3 - using **variables** (@...) is very strong tool. Each varible is replaced by its current value prior to processing specific script line.

Example

Here is a script for QRS detection and clustering.

```
%DESC QRS detection
FFTF CHANNEL(ECG|AV) FREQ(12;36;BANDPASS;ENVELOPE)
SHAPEFINDER CHANNEL(*) ECGSOURCE(ECG|AV)
SETCACHE CHANNEL(*) PARAMS(0)
SORTER CHANNEL(ECG) EXPORTPIC(@outputDir) PREFIX(@parentDirFirstPart_)
EXPORTM OVERWRITE DIR(@outputDir) PREFIX(@parentDirFirstPart_)
LOG_VARS @outputDir\logInfo.txt;@FilenameShort;@Samples;@SHF_MARKSCOUNT;@SRT_GROUPS;@
SRT_GROUP1COUNT;@SYSTEM_MEMFREEPERC;@SYSTEM_MEMUSED;@SYSTEM_TIME
```

Line 1: comment with script description
Line 2: generate amlitude envelopes from 12-36 Hz for channels containing ECG or AV in names. This command will create new datacaches and they will also be set as active
Line 3: run plugin for QRS detection, named „ShapeFinder“. Active datacaches will be used (i.e. amplitude envelopes in case of (ECG|AV) channels). This plugin will generate QRS marks.
Line 4: switch all channels to their first datacache
Line 5: run Sorter - it is plugin for shape categorization (clustering). It is by default set to human QRS categorization, thus we need only to link correct channels (ECG) into it and we ask sorter to export results image into specific directory (@outputDir), set in batch plugin GUI.
Line 6: marks are exported to text file with *.sel extension. The filename is combined from PREFIX, original source datafile name and it is placed in @outputDir folder.
Line 7: some of variables are logged into text file. Some variables are generated by system itself (as SYSTEM_...), some are generated when loading file (as @Samples, @parentDirFirstPart) and some of them are generated by plugins (@SHF_MARKSCOUNT by ShapeFinder, @SRT_GROUPS by Sorter).