

Contents

- [News](#) – Updates & Meetings
- [Scripts Spotlight](#) – Using dialogs
- [Spike2](#) – Burst mode sampling
- [Spike2 Script](#) – Average multiple result views with error bars
- [Signal](#) – Single channel analysis
- [Signal Script](#) – Duplicate idealised traces
- [Did you know...?](#) – Keyboard shortcuts to add cursors
- [Recent Questions](#) – Play two waveforms simultaneously

News

European Training Days: Thursday 4th and Friday 5th April 2013

[Back to top](#)

Join us for two days of training in Cambridge and learn how to use Spike2 and Signal for your research needs. These sessions are suitable for both existing and prospective users.

Attendance is limited so please [register](#) early to reserve your place on the appropriate course.

Future meetings and events

Australian Neuroscience Society 2013

Melbourne Convention and Exhibition Centre,
Melbourne, Australia
February 3rd - 6th 2013

CED European Training Days 2013

St Johns Innovation Centre,
Cambridge, UK
April 4th - 5th 2013

BNA 2013 Festival of Neuroscience

The Barbican Centre,
London, UK
April 7th - 10th 2013

Experimental Biology 2013

Boston Convention & Exposition Center,
Boston, MA, USA
April 20th - 24th 2013

Local Training Sessions

In addition to our annual UK and US training days, we are also happy to offer on-site local training sessions for Spike2 and Signal. These smaller training sessions can be tailored towards particular requirements for research groups or departments. If you are interested in organizing a local training seminar for your group or department, please contact us.



Latest versions of Spike2 and Signal

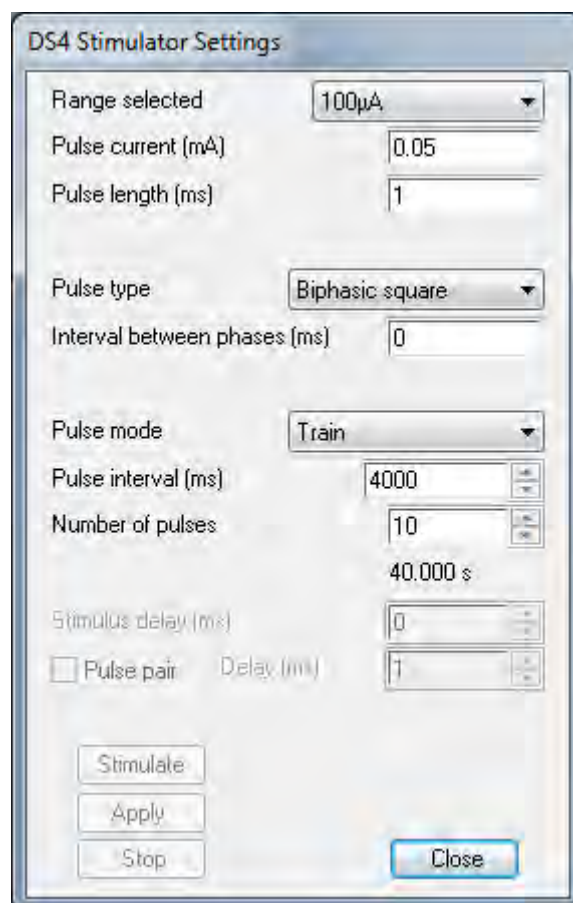
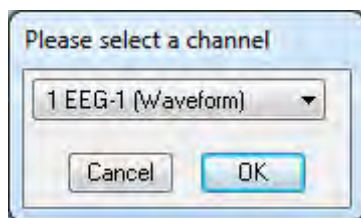
Updates for Spike2 and Signal are available from the [CED Downloads](#) page, or by clicking on the links in the table below. Demonstration versions of the latest software are also available.

Spike2 downloads	Released	Signal downloads	Released
Spike2 version 7.10c	09/12	Signal version 5.08	11/12
Spike2 version 6.17	10/12	Signal version 4.10	10/12
Spike2 demo	07/12	Signal demo	11/12

Scripts Spotlight

[Back to top](#)

Most scripts that you write will inevitably reach a stage where some form of user interaction is required to determine 'what happens next', be it selecting which channel to analyse, what type of analysis to apply and settings for the designated analysis task. This is something we see all the time in our favourite software packages, like Spike2 and Signal, in the form of dialogs. The script language has its own family of functions which allow you to build dialogs for use in a script. These can range from simple drop-down channel selection lists right through to multiple-item dialogs for the control of external devices, for example.



A simple example for channel selection (left) alongside a more complex example which provides control of the Digitimer DS4 stimulator (right)

For someone who has never approached a dialog before, the number of commands and arguments for the various `DlgXXX()` commands can appear quite daunting. Luckily help is at hand in the form of a script called `DlgMake`, which can be found in the scripts folder of your Spike2 and Signal installation folder.

This script lets you 'build' dialogs by adding items such as channel lists, checkboxes, buttons and number fields from a toolbar. Once you've added the required features you can use the Output button in the toolbar to generate the script code for your dialog and add this in to your own script projects.

Spike2

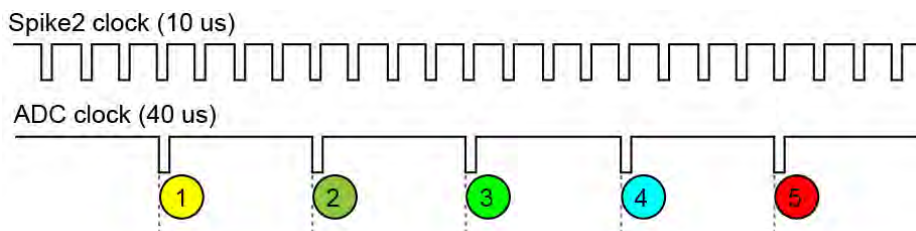
[Back to top](#)

- Q. I am recording multiple channels of spike data (at 25kHz) with a Power1401 and find that the longest run time for my data file is just under 3 hours. Is there any way I can adjust my settings to increase this without lowering my sample rates or number of channels?
- A. The burst mode sampling option in Spike2 can allow longer run times with the same sample rates or higher sampling rates at the same clock resolution. This option is most effective when sampling a large number of waveform and WaveMark channels.

The Burst mode check box is enabled in the Resolution tab of the sampling configuration if you have any type of Power1401 or a Micro1401 mk II or later. With the Burst mode option selected, instead of sampling the ADC at equal intervals of the 1401 clock tick, it samples a burst of channels as fast as the 1401 hardware allows.

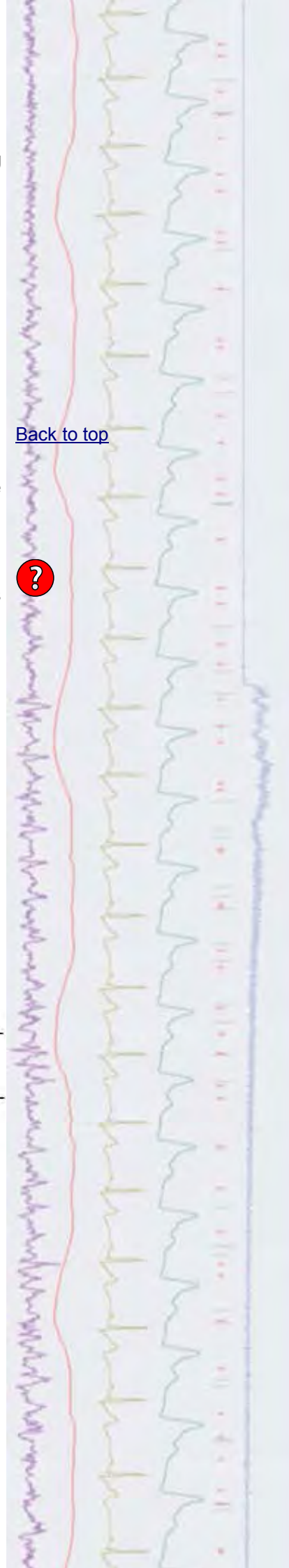
Non-burst mode

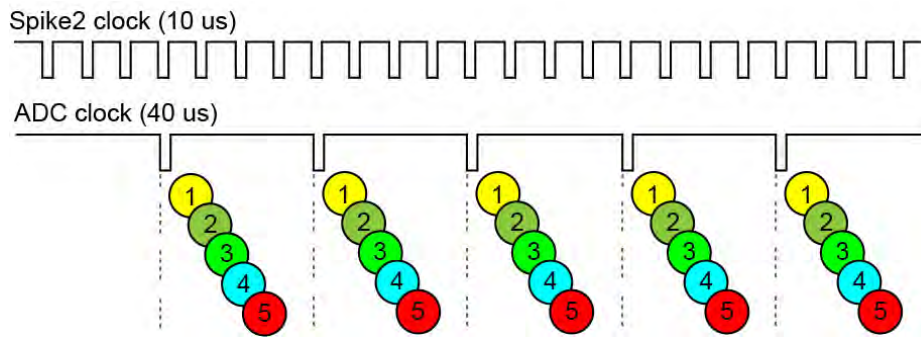
In non-burst mode, the Time units per ADC convert field sets how often the ADC samples in units of clock ticks. For example, with 10 microseconds per tick and the Time units per ADC convert field set to 4 (once every 40 microseconds), the ADC sample rate is 25 kHz. In a simple case with five waveform channels, the fastest each channel could be sampled is 5 kHz. Each sample is synchronised to the Spike2 clock and the pattern repeats every five samples.



Burst mode

In burst mode, the Time units per ADC convert field sets how often a burst of ADC samples is taken. With the same settings of 10 microseconds per tick and the Time units per ADC convert field set to 4, the burst rate is 25 kHz. The same five channels as in the example above would be sampled in a burst, each channel sampled at 25 kHz. The interval between samples in a burst depends on the 1401 hardware and is typically the interval implied by the maximum sampling rate for the 1401 hardware.





As shown above, the ADC samples no longer fall exactly at Spike2 clock times. The times between samples on a channel are exact, but the entire channel may be displayed time shifted by up to half a Spike2 clock. In this case, channels 1 and 2 would probably be timed for the tick just before them, and channel 3, 4 and 5 for the next tick. In many sampling applications this will not matter and the benefits of a longer run time or better sampling rates will outweigh this slight time shift.

You can find further information and technical details regarding the burst mode sampling option in the Spike2 on-line help.

Scripts: Spike2



- Q. I have created a number of result files during analysis of my data and would now like to create an average of all of these. Is there any type of 'batch process' script available for this?
- A. The script, `AvResults.s2s` can be used to create a new average with error bars based on a series of result views. When run the script prompts the user to select a folder which should contain at least two result views of the same number of bins and bin size. The script then creates a single channel average based on channel 1 from each result file.

[Back to top](#)

Right-click the script icon and save to disk

Signal



Show me How!

Click here to view a video tutorial on single channel analysis

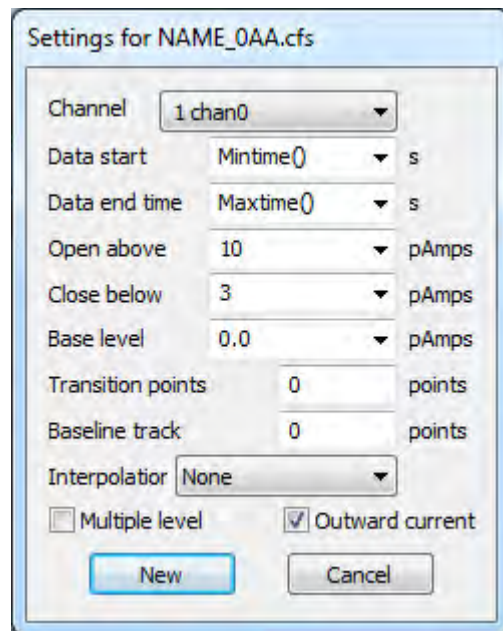
- Q. What features does Signal have for analysis of single channel data? I am interested in open/closed time detection based on threshold measurements.
- A. If you have enabled the `Signal` features for patch, voltage and current clamp, either during installation or from the `Clamp` tab of the `Edit` menu `Preferences` dialog a number of features for single channel patch clamp data analysis are made available from the `Analysis` menu. These include idealized trace formation using threshold detection or `SCAN` analysis¹, and generation of dwell time and amplitude histograms. An idealized trace is formed by converting the raw data trace into a series of open/closed times (events). Each time (or event) has an amplitude, a duration and a set of 'flags' to identify it. This article describes how to generate idealised traces using the threshold method.

[Back to top](#)

To create a new idealised trace select `New idealised trace (Threshold)` from the `Open/Closed times` option in the `Analysis`



menu. Set the channel to be fitted and the time range for the idealised trace in the dialog. If you are analyzing voltage activated channels you would set the time range to match the start and end time of the given stimulus, the first open/closed time found in the idealised trace is flagged as the first latency.

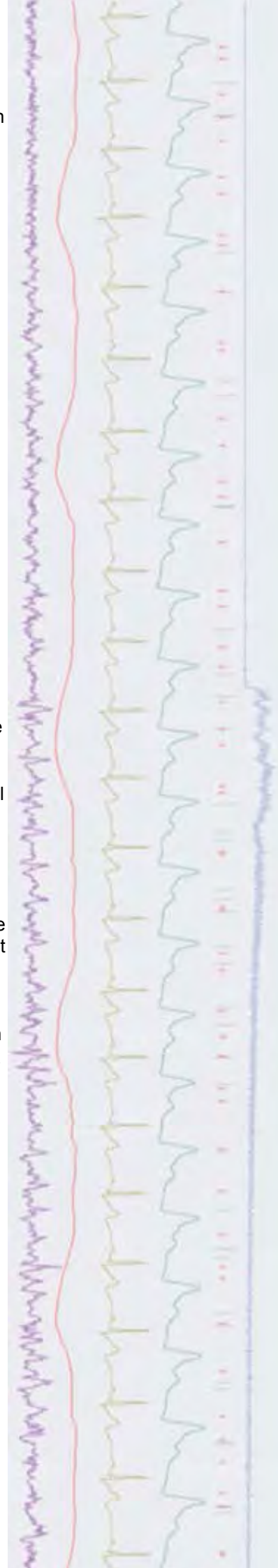


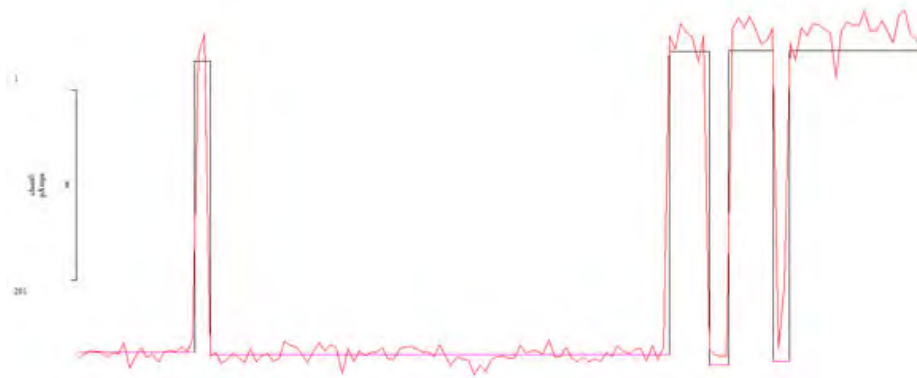
Idealised trace settings using the threshold method

The next two fields are used to set the thresholds for detecting openings in the ion channel. These are labelled *Open above* and *Close below* for outward current measurements, and *Open below* and *Close above* for inward current measurements. Outward current means that an opening of the channel produces a more positive current i.e. an upward direction in the time view, and inward current would produce a more negative current. You select outward or inward current using the checkbox in the lower right-hand area of the dialog.

The *Base level* and *Baseline track* settings are used for multiple level analysis. Sometimes the 'patch' of membrane being recorded will contain more than one ion channel, giving multiple current values if the channels are open at the same time. If this is the case you can select *Multiple level* from the checkbox in the bottom left-hand corner of the dialog. The thresholds for multiple levels are calculated by averaging the two thresholds set by the *Open/Close amplitude* settings, and then doubling the difference between this value and the *Base level*. The *Transition points* field can be set to a number greater than zero to provide a minimum interval after a transition before another transition can be detected.

Each time in the idealised trace starts at the first data point after the threshold crossing. The amplitude is the mean value of the data points within the open/closed period. You can select *Linear* from the *Interpolation* field in the dialog to calculate the assumed start of the open/closed times.





Idealised trace generated and overdrawn on raw patch clamp data

Press **New** to open the standard process dialog, and select the frames to be used for the idealised trace generation. The idealised trace for the data is generated in the first available channel above 200, and automatically overdrawn on the raw data trace. You can view and edit details of the events that form the idealised trace by right-clicking on the trace and selecting **View and modify events**. Click and drag the selected level on the trace to adjust amplitude and dwell times. Idealised trace data is stored in the resource file of the Signal data file and you can be backed-up using the **File** menu **Backup...** option. This can be useful if you want to revert to the original idealised trace at any time. Simply re-load the backup resource file from the **File** menu **Resource files > Apply resource file** option.

Once an idealised trace has been generated from the data, you can create **Open/Closed** time histograms and **Amplitude** histograms of the open/closed time durations from the **Open/Closed times** option in the **Analysis** menu. The settings for both **Open/Closed** time and amplitude histograms allow you to include or exclude data based on the associated 'flags'.

¹The SCAN method of fitting idealized traces was originally developed by Professor David Colquhoun at University College London

Scripts: Signal 

- Q. When analysing single channel data, can I duplicate an idealised trace? This would be useful to compare any changes I make with the original trace.
- A. The script `DuplicateIT.sgs` will create a duplicate channel of a selected idealised trace in a data file.

Did you know...?

You can add any vertical cursor to a data or result view in Spike2 with the keyboard shortcut **Ctrl+n**, where **n** is a number from 0-9. Horizontal cursors can be added with **Alt+n**.

Recent Questions

- Q. How do I go about playing two (or more) waveforms simultaneously from multiple 1401 DAC's on-line?
- A. In order to play out simultaneous waveforms from the 1401, the selected waveforms need to be specified in a single 'Play wave' area of the on-line play waveform list. Waveforms in this list are copied to the 1401 memory when you open a new data file for sampling, and the number of play wave areas that can be copied to the 1401 depends on their size and the type of 1401 you are using. Spike2 always reserves 256kB of 1401 memory for sampling and allows up to 10 'play wave areas' of up to 32MB each to store waveforms for replay. These 'play wave' areas can also be updated with new data during



[Back to top](#)

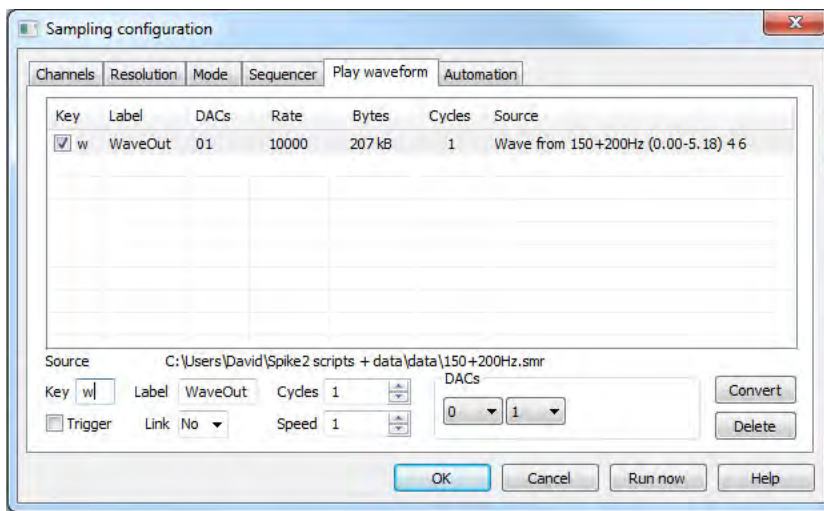
Right-click the script icon and save to disk

[Back to top](#)

[Back to top](#)

sampling using the script language to give greater flexibility.

The simplest way to load waveforms into the Play wave list is by selecting channels in a data file (holding down `Ctrl` and clicking on the channel numbers) and using the `Output Waveform...` command from the `Sample` menu in Spike2. Each play wave area will hold up to 4 data channels for output and the time range of data and the DAC outputs to use are selected from the dialog. Clicking the `Add to online` button will set aside memory to hold the selected channels and the label and key code that are used to define the play wave area are set here. Further options for play wave output such as number of cycles to play, the output speed and whether the play wave area is to be triggered (from the 1401 Trigger or E3 input) or linked to other play wave areas defined in the list, are available from the `Play waveform` tab of the sampling configuration, as below.



When you open a new data file ready to sample, any play wave areas are copied to the 1401 memory ready for output. There are several ways to start output of a wave during sampling; use the associated button in the Play waveform toolbar, or press the associated key code to start the output manually. Waveform output can also be triggered from a script (using the `SampleKey()` command), or from the output sequencer (using the `WAVEGO` command). See the Spike2 help pages or manual for further information about interactive and scripted arbitrary waveform output.

User group

Try the [CED Forums](#) bulletin board for software and hardware support

If you have any comments about the newsletter format and content, or wish to unsubscribe from the mailing list for this newsletter, please notify sales@ced.co.uk.

All Trademarks are acknowledged to be the Trademarks of the registered holders.

