

### Contents

- [News](#) – Updates & Meetings
- [Scripts Spotlight](#) – Checking for a valid data file
- [Spike2](#) – Sampling data
- [Spike2 Script](#) – Re-creating n-trodes
- [Signal](#) – Version 6
- [Signal Script](#) – Save processed channels to a new file
- [Did you know...?](#) – File extensions for 64-bit versions
- [Recent Questions](#) – Spike2 keyboard shortcuts for sampling

### News

#### **Spike2 version 8 and Signal Version 6 released**

We are pleased to announce the release of both Spike2 version 8 and Signal version 6. These are both major upgrades and include a range of new and improved features, with continuous development planned through the course of the versions.

#### **Get regular updates**

Registered Spike2 version 8 and Signal version 6 users will be able to download new updates as they become available to take advantage of the latest developments from CED.

#### **Take advantage of new features to analyse existing data**

The new versions are backward-compatible and will read data files and sampling configurations from older versions. Scripts from older versions should run without modification.

For more information, see the [Spike2 version 8](#) and [Signal version 6](#) data sheets.

We will be writing to all registered users of Spike2 and Signal with upgrade details in due course.

#### **Future meetings and events**

##### **CED Chinese Training Days 2014**

Shanghai, CA  
China  
January 22<sup>nd</sup> - 14<sup>th</sup> 2014

##### **Australian Neuroscience Society Annual Meeting 2014**

Adelaide Convention Centre  
Adelaide, SA  
Australia  
January 28<sup>th</sup> – 31<sup>st</sup> 2014

##### **Biophysical Society Annual Meeting 2014**

Moscone Center  
San Francisco, CA  
USA  
February 15<sup>th</sup> – 19<sup>th</sup> 2014

##### **Society for The Neural Control of Movement Annual Meeting 2014**

NH Grand Hotel Krasnapolsky  
Amsterdam  
The Netherlands  
April 21<sup>st</sup> – 25<sup>th</sup> 2014

[Back to top](#)



## Experimental Biology 2014

San Diego Convention Center

San Diego, CA

USA

April 26<sup>th</sup> – 30<sup>th</sup> 2014

### 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.

<b>Spike2</b>	<b>Released</b>	<b>Signal</b>	<b>Released</b>
<a href="#">Spike2 version 8.00a</a>	12/13	<a href="#">Signal version 6</a>	12/13
<a href="#">Spike2 version 7.12c</a>	09/13	<a href="#">Signal version 5.09</a>	05/13
<a href="#">Spike2 version 6.17</a>	10/12	<a href="#">Signal version 4.10</a>	10/12
<a href="#">Spike2 demo</a>	03/13	<a href="#">Signal demo</a>	06/12

### Scripts Spotlight

[Back to top](#)

Many of the scripts you write will concentrate on analysing data and, to this end, you will normally have to give the user an opportunity to select some data to analyse when the script runs. Sometimes a user might have a data file already open for analysis when they run the script. Other users might wish to be given the option of selecting a file from a folder once the script is running. Using our knowledge of the `View()` family of commands, we can add some code to check for an already open data file and use it if found, or prompt the user to select a file for themselves.

```
var dataFile%;  
dataFile%:= FrontView();  
if ViewKind(dataFile%)<>0 then  
    Message("Please select a data file for analysis");  
    dataFile%:= FileOpen("",0,1);  
    if dataFile% < 0 then  
        Message("Unable to open a data file!");  
        Halt;  
    endif;  
else  
    Message(Print$("We found %s", Filename$(3)+Filename$(4)));  
endif;
```

First we bring the current view, whatever it may be, to the front using `FrontView()`. The `dataFile%` global variable will then be assigned the view handle.

Next we use `ViewKind()` to see what type of view `dataFile%` is. This returns a value corresponding to the type of file; a return value of 0 indicates a data file. Here I use the `<>` expression which means 'is not equal to'. Therefore, if the return value of `ViewKind()` *does not* equal 0, we step into the `if...endif` statement. If 0 was returned we step to the `else` statement and display a message with the name of the file found. This is useful if the user cannot see their data file because it has been iconised!

If the current view is not a data file we can give the user a message asking them to select a file. For this we use `FileOpen()` with an empty string argument, which opens the standard file open dialog. The second argument specifies the type of file to open, again, 0 is used to indicate a data file. The third argument sets what to do with the file



when it is first opened. An argument of 1 here will make the new window visible immediately. `FileOpen()` returns the view handle of the newly opened file so we can assign this value to our `dataFile%` variable to get the view handle of our data file.

So far, so good; we've checked to see if the current file is a data file and given the user the option of selecting a file if not. The final stage is to check that the file opened correctly. For this we can use the return value from `FileOpen()`, which is now stored in our `dataFile%` variable. If the file opened successfully then the view handle of the newly opened file is returned but, if the file could not be opened, then a negative error code is returned. Here we can check that the `dataFile%` variable holds a value greater than 0 and, if not, give the user another message indicating that the file could not be opened, and `Halt;` the script.



**Show me How!**

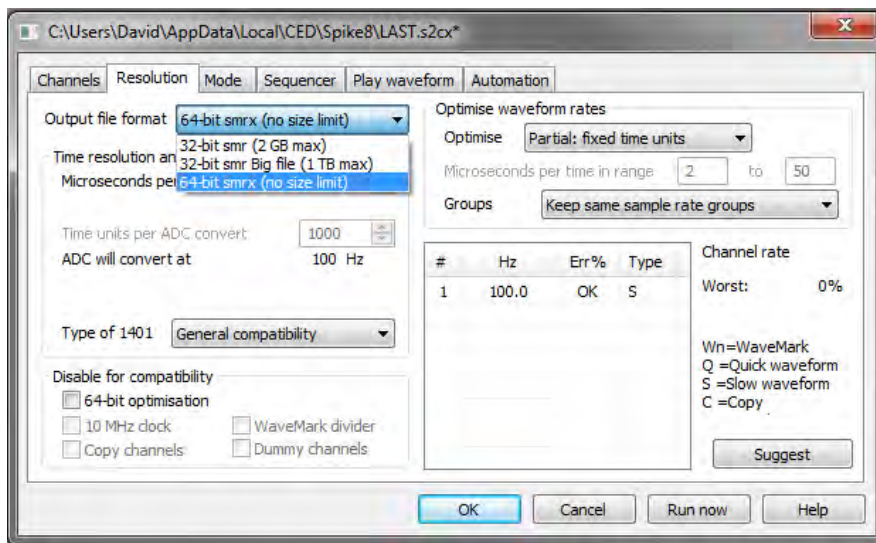
Click here to watch a video on setting up sampling configurations in Spike2

Q. I see that Spike2 version 8 can produce 64-bit data files, which will be very useful for my long-term recordings, but for some of my other studies I export my Spike2 files to MATLAB to make use of some custom routines. I only have a 32-bit copy of MATLAB so is there any way to save sampled data from Spike2 version 8 to a 32-bit file?

[Back to top](#)

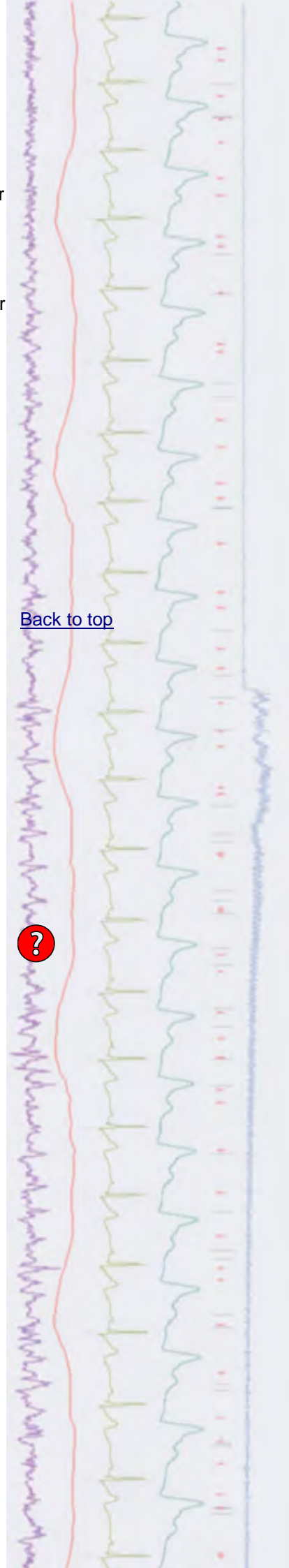
A. Spike2 version 8 is optimised to create and work with 64-bit data files, which have no time limitations and are limited in size only by the capabilities of your operating system, allowing both high timing resolution and long run times. The 32-bit file system is still supported, so older files can be opened and analysed using version 8.

As in this example, you may use third- party programs that cannot read 64-bit files to analyse your data, or your collaborators may be using an older version of Spike2 that cannot read the 64-bit file format. In this case, the new 64-bit files can either be exported to 32-bit from the File menu Export as... option, or you can sample data directly to 32-bit files.



*Spike2 version 8 sampling configuration showing Output file format settings*

To sample data to the old file formats for compatibility, select either the 32-bit





big option or the 32-bit option from the Output file format drop-down list in the Resolution tab of the sampling configuration. The 32-bit big format is compatible with Spike2 version 6 and version 7. The standard 32-bit format is the most compatible format for old versions of Spike2.

#### Scripts: Spike2

- Q. I made a new WaveMark channel (n-trode) from a continuous recording and then sorted the spikes into units. Although I was happy with the area of the waveform considered for templating, I did not store enough of the waveform data in the WaveMarks. Is it possible for me to go back and generate the same n-trode WaveMarks again, but with more pre and post trigger data? Ideally I'd like to use my existing WaveMarks as the trigger for each new section.
- A. The script, [Evt2NTrode.s2s](#) allows you to extract WaveMarks from 1, 2 or 4 pre-recorded waveform channels. A settings dialog in the script is used to set the source waveforms to use, the number of pre and post trigger sample points to include and alignment methods for n-trode WaveMark data. The new WaveMark channel is then added to the data file as a permanent disk-based channel. This script requires Spike2 version 7 or higher.

# Signal

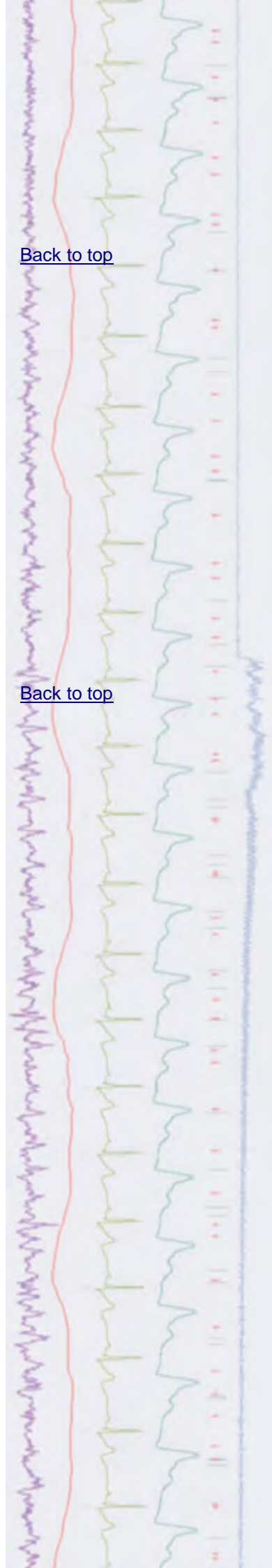
Version 6 is the latest release of Signal and includes many new and improved features. Version 6 is completely compatible with earlier versions; it will read data files and sampling configurations created by earlier versions without problems. Scripts that ran in earlier versions should also work without modification.

New features include:

- Dynamic clamp, patch and voltage clamp functions have been extended with an improved membrane analysis dialog, a duplicate option in the dynamic clamp models dialog and the addition of an optional rectification mode in the noise model.
- Both standard voltage based telegraphy and the auxiliary telegraphing system can now operate simultaneously.
- A new type of data called real markers has been added. Real marker data consists of a marker (a time and four code bytes) plus one or more floating point values.
- The memory channel mechanisms used to hold idealised trace data have been extended to allow marker and real-marker data channels held in memory. These could be used to display peak times and values from waveform channels, for example.
- Both 64- and 32-bit versions of Signal version 6 will be shipped. The 64-bit version requires a 64-bit version of Windows, is some 10 percent faster than the 32-bit code and will interface with 64-bit versions of MATLAB.
- The virtual channel system has been extended to provide mechanisms to generate a waveform from data in a real marker channel.
- New memory channel dialogs have been added that allow you to create,

[Back to top](#)

[Back to top](#)



import data into and modify memory channels holding marker, real marker and idealised trace data.

- New built-in processing mechanisms have been provided to create marker and real marker data both offline and during sampling, plus a new script language function to do the same.

There are many other improvements and more are planned. You can find a full list of new features, bug fixes and changes in the [technical history](#) page on the website.

### Scripts: Signal

Q. I would like to be able to create a duplicate view from my data file holding rectified and smoothed data, for comparison alongside the raw data.

A. The script, [ProcessedDataView.sgs](#) creates a duplicate window holding rectified and smoothed data, and displays it alongside the raw data file for comparison. This could be modified to perform any number of the Modify Channel commands if required.

[Back to top](#)

### Did you know...?

Spike2 version 8 and Signal version 6 are available as both 64-bit and 32-bit installations. The new 64-bit versions create data files and sampling configurations with an *x* suffix added to the file extension to distinguish them from the standard 32-bit file formats. Spike2 data files are saved as *.smrx* and sampling configuration files as *.s2cx*. Signal sampling configurations will have the extension *.sgcx*.

[Back to top](#)

### Recent Questions

Q. Can I control the writing to disk of data in Spike2 from a remote keyboard?

A. The following keyboard shortcuts are available for sampling commands, allowing the use of a remote keyboard to control sampling:

- Sample Start/Stop – Ctrl+Alt+S
- Sample Reset – Ctrl+Alt+R
- Sample Abort – Ctrl+Alt+A
- Toggle Write to disk – Shift+Ctrl+W

[Back to top](#)

There are many more keyboard shortcuts available in Spike2, allowing you keyboard control of scrolling and zooming of data, creation of cursors and selecting channels, to name a few. See the on-line help for further details.

### CED User forums

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](mailto:sales@ced.co.uk).

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

