

Contents

News – General
Spike2 – Recording multimedia files
Signal – Channel modification using multiple frame commands
Spike2 Script – Simple inter-file arithmetic
Signal Script – Amalgamating Frames
Did you know...? – Text importer improvements
Recent questions – Display of grouped channels

News

First, we must apologise for the late arrival of this newsletter. We had been waiting for the release of Spike2 5.04 to highlight some of the new features available in this version. Also one of our main contributors to the newsletter, Steven (who some of you may know), became a father recently and so, understandably, was otherwise engaged!

I hope that you will find the content of this newsletter worth the wait!

Training Days in New Orleans 6th & 7th November 2003

The pre-Neuroscience training days will be held on Thursday 6th and Friday 7th of November at the Hotel Monteleone, New Orleans. Further details are available from our web site at www.ced.co.uk/nw9y.htm

Version **5.04** of Spike2 is now available and freely downloadable from www.ced.co.uk/upu.shtml for registered v5 users. This revision enables Spike2 to make use of multimedia files for video and audio capture, in parallel with the recording of Spike2 data files, using the associated **s2video.exe** application. Full details of new features are documented in the revision history of Spike2 and the updates section of the website. The next release (5.05) will include interactive curve fitting to time, result and XY views and creation of new time view data channels based on active cursor measurements.

A demonstration version of the software is also available from www.ced.co.uk/pru.shtml

Version **4.18** of Spike2 is freely downloadable for registered v4 users.
Version **2.14** of Signal is also freely downloadable for registered v2 users.

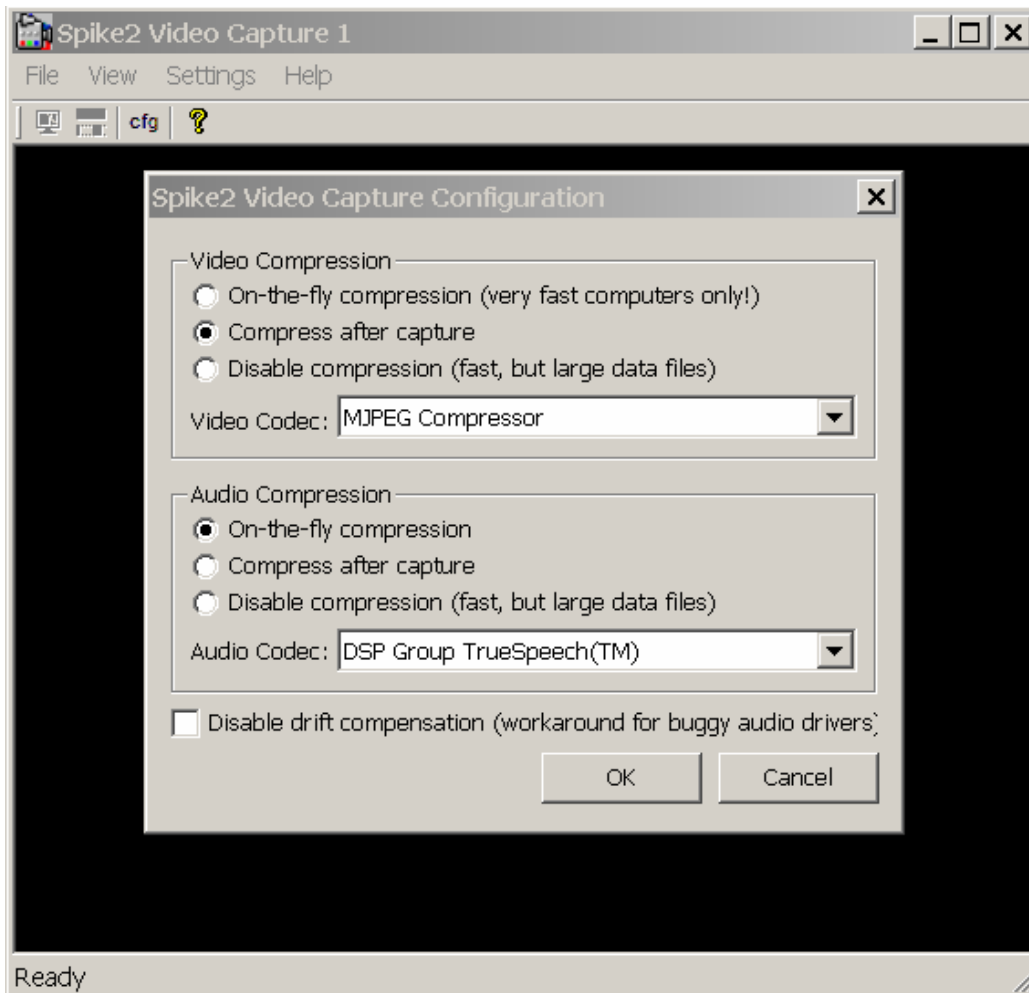
SPIKE2

- Q. I would like to record simple video and audio files alongside my Spike2 data files, to monitor behaviour during an experiment.
- A. The latest Spike2 version 5 now ships with an additional application, **s2video.exe**. The s2video application acts as a slave to Spike2, and responds to Spike2 starting/stopping sampling and saving data to disk, by performing the same actions on the multimedia file. You can run up to 4 copies of this application at a time; each copy of s2video will record one video and/or audio track. The quality of the recording, in terms of video resolution, frame rate and audio sample rate depends on the hardware you are using and the software that you have installed.

The multimedia files are saved alongside the Spike2 data files, and with the same file name. These files can be viewed using the Spike2 View menu Multimedia Files command. When you scroll the time window to view data, the multimedia file window(s) also scroll to the same point. If you rerun the time view, the multimedia file will also rerun.

To run Spike2 data and multimedia capture simultaneously, you will need a suitably powerful computer. A machine with at least a 2GHz processor, 256 MB memory and 40 GB hard disk space is recommended. The s2video data capture and replay software is based on DirectShow (which is part of Microsoft DirectX). It is likely that you already have a version of DirectX installed. The latest DirectX version is 9.0b, which is available for download from the Microsoft website; we strongly recommend that you use the latest version.

To record video you will need a web cam or video camera that is supported by DirectShow. The drivers required to run these devices will normally be supplied with them. With Windows XP or NT2000, your system may automatically recognise the camera when you connect it and locate the drivers for you. You can record audio from the camera (if your camera supports this) or from a sound card.



s2video application with configuration menu

The first time you run the s2video application you must select the devices to record from with the Settings menu, Video Device and Audio Device commands. Once you have selected a video device, you can preview the image in the video display area of the window, and additional "property" menu items appear to control the selected video and audio devices. The properties available depend on the selected devices. They may include controls for brightness, saturation and colour balance for the camera as well as volume and EQ controls

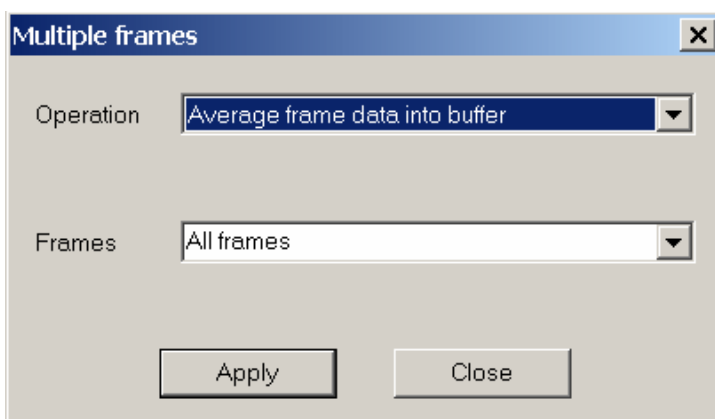
for the audio device. The next step is to set the frame rate for video capture; the lower the frame rate, the smaller the resulting data file. You should use the lowest frame rate that gives you enough information for your task; rates of 10, 5 or even 1 frames per second are sufficient for many applications.

The configuration menu controls the type of data compression to apply to your video and audio data. As a raw video stream can produce several MB of data per second, compression is recommended to give the smallest file size possible. On-the-fly compression is only really useful with a very powerful computer or specialised hardware, unless you are running at low frame rates. Compress after capture means that the data is recorded raw to a temporary file, and when sampling is stopped and the user saves the data file, the temporary file is compressed and saved alongside the data. Compression is performed as a background task, so that Spike2 can sample another file immediately. This is the recommended option for most users.

Compression is achieved through the use of specialist software called codecs. These are made up of two components, a **coder** and a **decoder**. The coder changes the input signal in some way, and the decoder reverses the process. In this case, we are interested in codecs that compress the input signal to save it to disk, and then decompress the signal when we wish to load it from disk. When installed, DirectX adds a standard set of codecs to your system. It is likely that the installation of your video camera or web cam added further codecs to the system. The audio and video codec fields in the configuration dialog let you select codecs on your system that support audio and video. The s2video on-line help gives a comparison of using s2video with some of the codecs that you are likely to find installed.

SIGNAL

- Q. I make a lot of use of the Modify channels command in Signal to apply functions such as rectify, 3-point smoothing etc. but find this very tedious when applying the same functions to a number of frames in the data file. Is there a script I could use to automate this process?
- A. In this case, it would be much easier to use the Multiple frames option from the Analysis pull-down menu. This command produces a dialog that can be used to carry out numerous operations on multiple frames in the document. The dialog contains a selector for the operation to be applied, a field to enter any operation data required plus a standard set of controls to specify frames in the data document to be used.



Signal Multiple frames dialog

The operations available from the dialog include all of the frame buffer operations, all of the channel data modification options and the option to tag and untag frames. For operations that modify the frame buffer, such as accumulating an average or summing frames, the effect is straightforward. For operations that modify file view data, such as rectifying channels or subtracting buffer data from frames, the changed data must be saved to disk if the action is to

'any other Arr...() arithmetic command

```
ArrConst(View(res%).[], arr1[]); 'Copy our calculated data to the result view
View(res%).WindowVisible(1);    'Show result view
View(res%).Optimise(-1);        'Optimise the data
FrontView(res%);                'And bring to the front
Return 1;
end;
```

Scripts. Signal

Q. I am looking for a way to amalgamate data files from a series of experiments, in Signal. Is there a script that does this?

A. The following script amalgamates Signal data files using the MCF command line utility. The user selects two files and these are amalgamated in a new file that is saved in the Signal directory.

```
*****
'$amalgamate cfs files using the MCF utility

Var File1%;           'View handle of first file
Var File1$;           'File name of first file
Var File2%;           'View handle of second file
Var File2$;           'File name of second file
Var output$;          'String to hold the Output file name
Var Result$;          'The command line for the MCF utility
Var Signal$;          'The filepath to the Signal directory
Var err%;
Var NewFile$;         'The file name for the newly created file
Var find$;            'Separator string for filename
Var i%;
Var length%;          'Holds length of filepath

File1%:=Fileopen("",0,1,"First File"); 'Open first file
File1$:=Filename$(0); 'Get filename of first file
Output$:=Filename$(3); 'Get filename minus the filetype (.cfs)
Fileclose(0);         'Close first file
File2%:=Fileopen("",0,1,"Second File"); 'Open file to amalgamate to first file
File2$:=Filename$(0); 'Get filename of second file
Fileclose(0);         'Close second file

Output$:="A" + Output$+".cfs ";        'Give full file name for amalgamated file
Signal$:=FilePath$(2);                 'Get filepath to the Signal folder
Result$:=Print$("MCF " + File1$ + " " + File2$ + " " + Signal$ + Output$ + "-a");
'Produce command line for MCF

FilePathSet(Signal$);                   'Set the filepath to the Signal folder
Progrun(Result$,1);                      'Run the MCF command line
Find$:=Print$("\\");                     'Set separator string

Repeat
i%:=InStr(Signal$,Find$,i%+1);          'Search the Signal filepath for the separator
string
If i%>0 then                             'If present
    length%:=Len(Signal$);                'Get length of the Signal filepath
    Signal$:=Left$(Signal$,i%)+"\\"+Right$(Signal$,Length%-i%);
    'Set the filepath to save new file too
    i%:=i%+1;                             'Increment index to search Signal filepath
endif;
until i%=0;                              'Until search returns 0

NewFile$:=Left$(Output$,8);              'Give new file the generated filename
NewFile$:=NewFile$+".CFS";               'And correct filetype

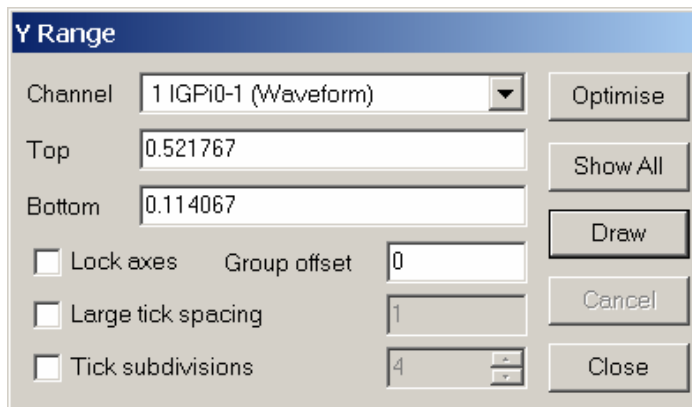
*****
```

Did you know...?

The text importer has been made easier to use and you can now choose to import waveform channels as RealWave or waveform, and event channels as Event-, Event+ or Level.

Recent questions

- Q. I have a number of files that I would like to view with grouped or overdrawn channels in Spike2. I do this by clicking and dragging the channel numbers to overlay the data. However, when I have many waveform channels overdrawn, it is difficult to make a distinction between them and changing the scale only affects the bottom channel in the group. Is there an easier way to view and manipulate overdrawn waveform data?
- A. The Y Axis Range dialog has now been extended to include extra fields that are enabled when the current channel is a member of a group that shares a y-axis. These new fields let you lock the y-axes together, so that the same y-axis controls all of the grouped channels. You can also apply a per-channel vertical display offset to space out channels that would otherwise draw on top of each other. This is a new feature in version 5.



Extended Y Range dialog in Spike2 version 5.04

User group

It will take a short time to set up an archive message board on our web site. As soon as this is done I will use this newsletter to announce that it is available. This seems to be the most sensible route to go rather than direct emailing both for security and for minimising spam.

If you wish to un-subscribe from the mailing list for this newsletter, please notify sales@ced.co.uk.