CED eNewsletter
Cambridge Electronic Design Limited

#85 ~ April 2014

Contents

News

Back to top

European Training Days 2014

Join us in Cambridge on June 5th- 6th for our annual training days and learn how to use Spike2 and Signal more efficiently to customise and streamline your analysis. Register on our website to reserve your place on the appropriate course.

Future meetings and events

Experimental Biology 2014
San Diego Convention Center,
San Diego, CA
USA
April 26th – 30th 2014

Magstim Neuroscience Conference 2014
The Examination Schools,
Oxford
UK
May 10th – 11th 2014

CED European Training Days 2014
St. John's Innovation Centre,
Cambridge
UK
June 5th – 6th 2014

CED Chinese Training Days
Suzhou City,
Shanghai
China
June 22nd – 24th 2014

Symposium for Chinese Neuroscientists Worldwide 2014
Suzhou City,
Shanghai
China
June 24th – 28th 2014

Physiology 2014
Queen Elizabeth II Conference Centre,
London
UK
June 30th – July 2nd 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.

Spike2	Released	Signal	Released
Spike2 version 8.01c	03/14	Signal version 6.01a	03/14
Spike2 version 7.13	03/14	Signal version 5.10	03/14
Spike2 version 6.17	10/12	Signal version 4.11	02/14
Spike2 demo	12/13	Signal demo	03/14

Scripts Spotlight

When it comes to intensive computational analysis, many researchers like to use their own routines, written with MatLab. Both Spike2 and Signal contain script commands that can open a MatLab process, transfer data from a script to MatLab for external processing and then transfer back the resulting values (as long as MatLab is installed on the same PC of course!).

To use MatLab script support in Spike2 and Signal, you need to select the MATLAB interaction script functions option during installation.

`MatLabOpen()` opens an invisible connection to a MatLab command window. The other Matlab script commands will fail if there is no open connection. If you need to make the command window visible you can use `MatLabShow()`. To close an open connection, use `MatLabClose()`.

`MatLabPut()` takes a script variable, or a real or integer array, and passes it into the connected MatLab command window.

`MatLabGet()` copies a variable from the command menu into a script variable.

`MatLabEval()` requests the command window to execute an arbitrary command and gets the result as a string.

Typical use would involve passing data to MatLab with `MatLabPut()`, processing it with `MatLabEval()` and then retrieving the result with `MatLabGet()`.

Show me How!
Click here to watch a video on setting up event channels in Spike2

Q.   Can I log stimulus pulse times in my recordings, based on the TTL output of my current stimulator?

A.   Event channels represent time stamps, such as the pulse output of a stimulator connected to a 1401 digital input, for example. Event channels can record time stamps based on the rising edge of a digital pulse (Event+), the falling edge of a digital pulse (Event-) or both (Level). Level event channels are usually used to indicate the on or off state of a connected device such as a heat lamp or light source, for example.
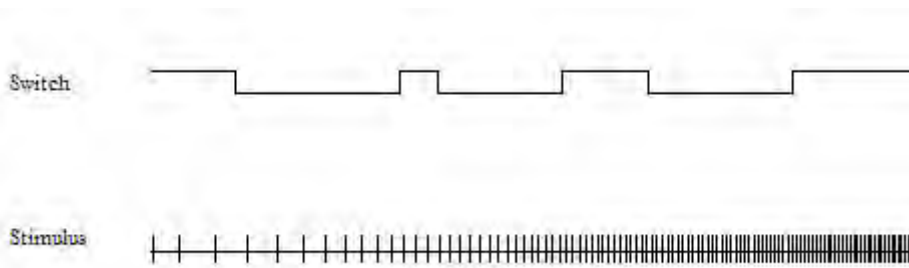
Examples of an Event+ and Level channel

Event channels are created by clicking New in the Sampling configuration and selecting the type of Event channel you want to record in the Type drop-down list. Event channels can use any of the 1401 digital input bits from 8-15, as set by the 1401 Port drop-down. Bits 8 and 9 are usually available on the BNC connectors labelled Event Input 0 and 1 on the front panel of your Micro1401 or Power1401, with bits 10-15 available on the Digital Input connector on the rear.
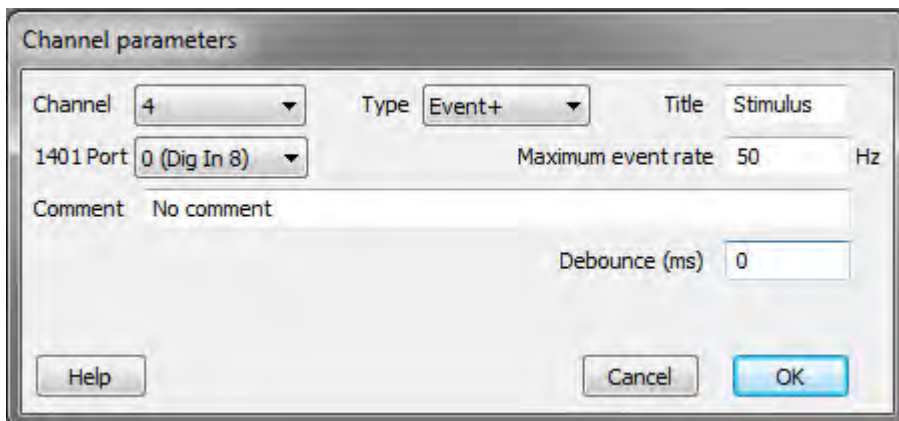

Parameters dialog for an Event+ channel

The Maximum event rate field should be set to your estimate of the maximum mean event rate that will be sustained over a few seconds. Spike2 then uses this value to optimise buffer space allocation. This field does not set the maximum instantaneous rate, which may be much higher.

The Debounce field sets the minimum acceptable interval between consecutive events in milliseconds. Events that fall closer than this value to the previous event are not saved to disk. Typically you would use this if logging events from a mechanical switch. These types of switches commonly have 'bouncy' contacts so that, when the switch is activated, you get multiple switch actions in a very short period, all logged as events. Using the Debounce setting will log the first event and ignore all those that fall within the debounce time range.

The marker channel in Spike2 can also be used to log events from an external source. Markers contain both a time stamp and four 8-bit marker codes, which are set by reading the values of digital input bits 0-7, and are often used to signify certain experimental conditions. An example of this could be codes sent to the 1401 from an experiment design program such as E-prime or Presentation to signify different types of stimulus (See Recent Questions, below, for more information). Channel 32 is always reserved for digital marker data input and is set-up in the sampling configuration by selecting Marker from the Type drop-down list in the channel parameters dialog.
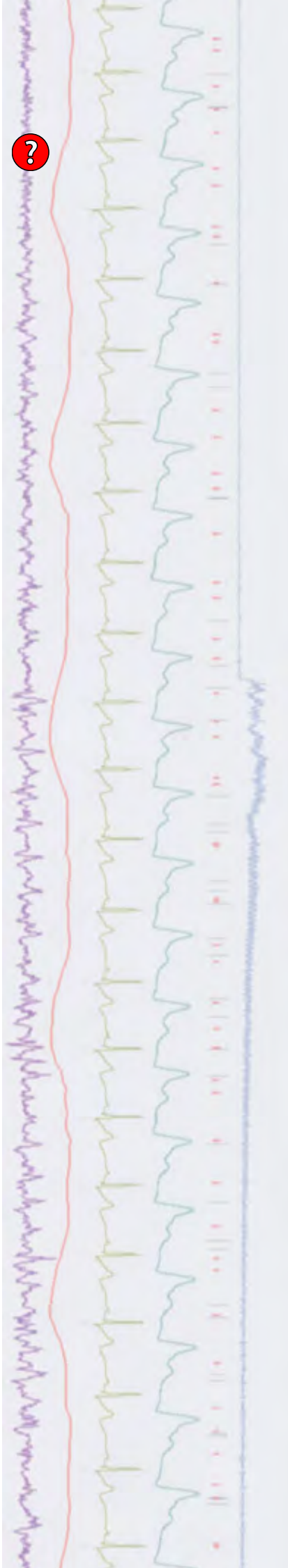
Scripts: Spike2

Q.   I'm recording video files of behavioural activity alongside my data recordings and would really like to find a way of marking the position of the subject in the video.

A.   The script, `PlotOnMMFrames.s2s`, displays multimedia video frames as a background image in an XY view. This allows you to step, by a user-defined frame range, through the video recording and mark points of interest, such as the position of a subject, by clicking with the mouse in the XY view.
This script requires Spike2 version 7 or higher.

Signal

Q.   What are the main differences between the various sweep modes available in Signal?

A.   Signal has a variety of sampling modes making it suitable for a wide range of applications. The `Sweep mode` selector in the `General` tab of the Sampling configuration defines how sweeps of data are taken and triggered and how sampling sweeps relate to the outputs system. The different modes available are:

Basic
The trigger (if any) for sampling data is a TTL pulse at the start of the sweep, and pulse outputs start and finish at the same time as the sampling sweep. It is useful for a wide variety of straightforward sampling requirements.

Peri-trigger
The sampling trigger can be before the start of the sweep, at the start of the sweep or at any point within the sweep. Pulse outputs start at the trigger point and finish at the end of the sampling sweep. This mode allows a wide variety of triggers including threshold crossings on a sampled waveform channel, and allows data to be collected before the sweep trigger, which no other sweep mode can achieve. The trigger point and type of trigger are set in a separate Peri-trigger tab that is enabled in the configuration when you select Peri-trigger mode.

Extended (Called Outputs prior to Signal version 6)
In this mode the pulse outputs are triggered rather than the sweep. The outputs can be longer than the sampling sweep, occur both before the sweep starts and after the sweep is over and the sampling sweep is triggered by the outputs; the sampling sweep trigger time is set along with the outputs in the pulses dialog.

Fixed interval
Similar to Extended mode, but the sweeps are internally timed so that they occur at the specified interval. For a full description of Fixed interval mode, see Newsletter #84.

Fast fixed interval
Allows smaller intervals than fixed interval mode (less than 100ms) but does not support any variation between sweeps or multiple states sampling due to timing constraints

Fast triggers
This mode is very much like Basic mode except that digital markers, multiple
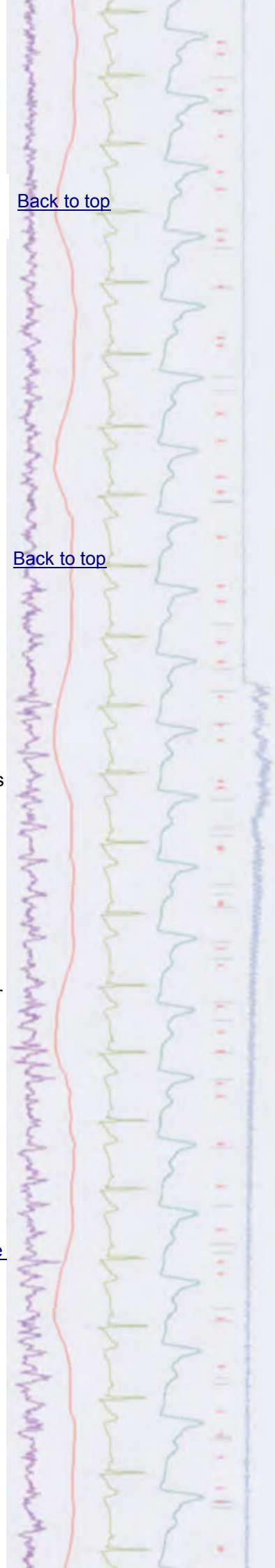
frame states and variations in the pulse outputs are not available, in addition the absolute start time for a sampled frame cannot be measured and is set to zero. This keeps the inter-sweep interval to a minimum - less than 100 microseconds usually.

Gap free
In this mode only the first frame is actually triggered; subsequent frames start immediately after the previous frame finishes with no loss of data or any other variation in sample timing. It is used when you want to sample continuously but are happy to break up the data into frames.

Scripts: **Signal**

Q.   I would like to be able to plot amplitude measurements from a data file with multiple states into a single XY view. I know that this can be achieved by setting up multiple channels for measurements by hand, but I was hoping that there might be a script available to do this automatically.

A.   The attached script, `MeasureFromStates.sgs`, is an example script that measures evoked response amplitude from a file with multiple frame states and plots the measurement from each individual state to a separate channel in an XY-view. This could easily be modified to take a wide range of measurements from multiple states.

Did you know...?

You can mark transitions in a level channel by creating a marker type memory channel and importing the times from the level channel. High-to-low transitions are marked with code 00 and low-to-high transitions with code 01. The `Marker Filter` option, available from the `Analysis` menu, can then be used to show and hide codes indicating either level.

Recent Questions

Q.   I am using E-prime software to generate various stimuli for my psychological experiments and would like to interface this with Spike2 so that the stimulus type is marked in the data file. Is there any way to log markers from E-prime in Spike2?

A.   The 1401 family has 16 digital inputs and 16 digital outputs. Digital input bits 15-8 are the event ports and Digital input bits 7-0 are used for the Digital marker channel as shown by the diagram below.
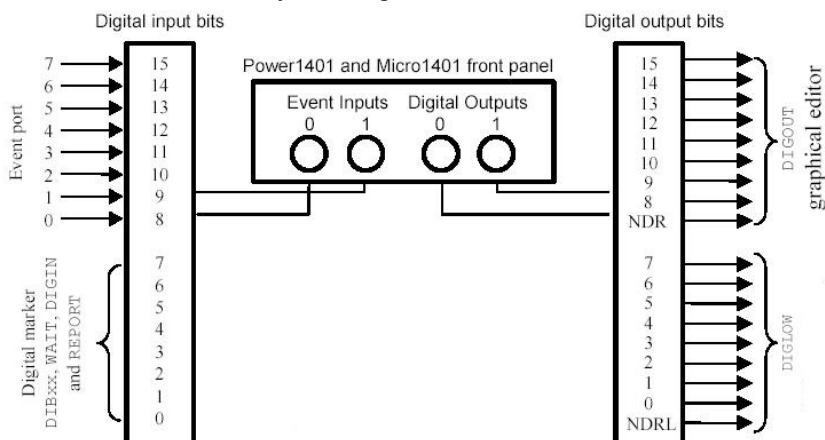

The digital input and output ports are 25-way connectors. The data and ground pins are the same on both. See your *1401 Owners Manual* for full details of all pins.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	GND
Pin	1	14	2	15	3	16	4	17	5	18	6	19	7	20	8	21	13
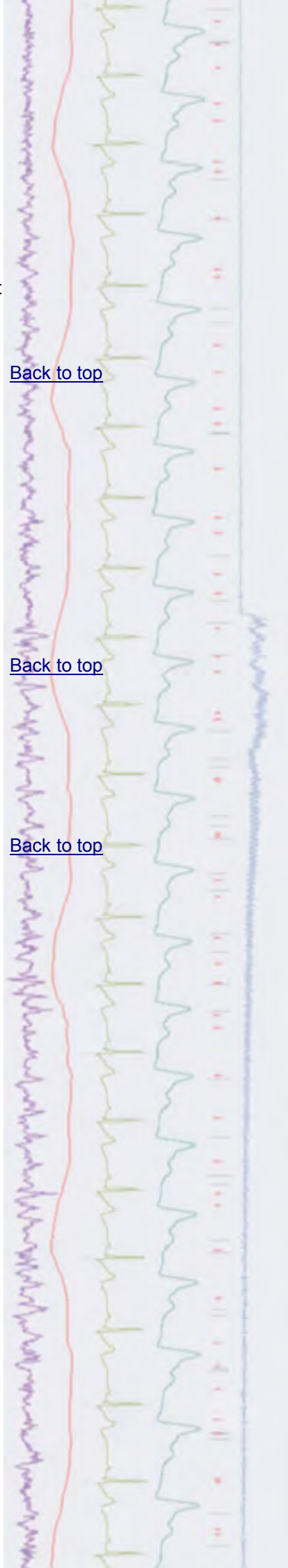
Experiment design programs such as E-prime and Presentation allow the user to generate video and image files as a stimulus and output digital marker information relating to the current stimulus on a 25 way D-type parallel port. This can then be connected to the 25 way digital input connector on the 1401 using an appropriately wired cable.
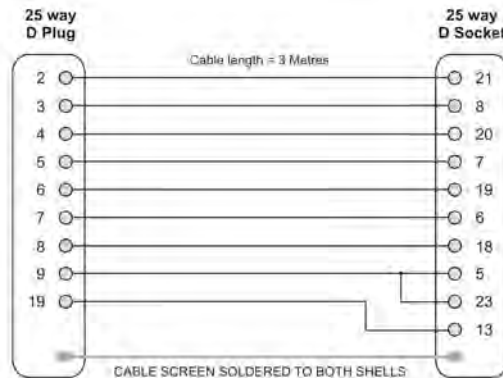
Wiring diagram from parallel port (left) to 1401 digital input (right)

To record the E-prime or Presentation marker data to the Spike2 file, set-up a digital marker channel (channel 32) in the sampling configuration. Codes sent by the program will then be displayed in the marker channel during sampling.

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.