Spike2 and Drosophila song analysis

TOC

Figures

Song file preparation.

The file needs to be in .wav format.

Using audio manipulation software, insert silence in the song in the place of any noise not to be analysed (female rejection song, box lid opening etc).

Ideal song has a high signal to noise ratio, and clear peaks of similar maximum amplitude across its length. A particular worry is double peaks, which are produced by a very limited number of flies, perhaps because of wing problems.

I completely avoided digital enhancement of the song as it caused complications. For example, digitally reducing the noise altered the song peaks which could confuse Spike2 during the analysis.

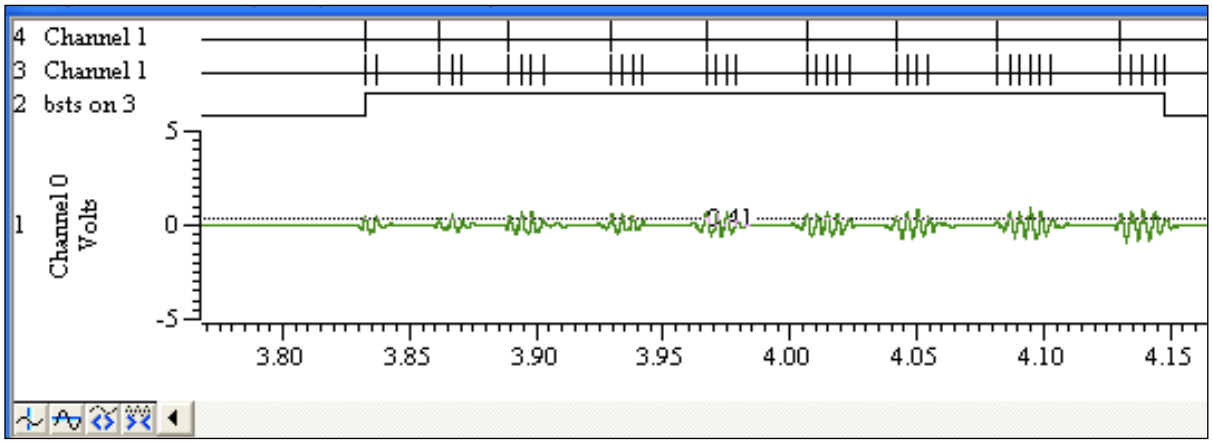
File import

Start Spike2 and import a .wav file (file->import).

Overview

3 channels need to be generated for each file. 2 are 'event' channels and 1 is a 'burst' channel (Figure 1). The event channels are using the peak detection ability of Spike2, helped by a minimum time below which no new peaks are detected. Burst channels are generated by the burst script, which was written by the developer and is available in the standard Spike2 installation.

Figure 1: Illustration of the channels to be generated for each song file.'4 Channel 1' and '3 Channel 1' are event channels (pulse event and spike event, respectively), '2 bsts on 3' is the burst channel. 'Chan 01' is the imported song file. After importing into Spike2, only the bottom channel exists. A horizontal cursor has also been drawn.



 can be used to (in order) create a vertical cursor, create a horizontal cursor, zoom in and zoom out.

The results are analysed as soon as they are generated, and the output is recorded in excel next to the rest of the information available for the individuals (pedigree

status, CHC information etc). Excel can be programmed calculate some song parameters from the raw output of the song analysis, described in this document.

The actions in the guide are written in the chronological order they are performed in.

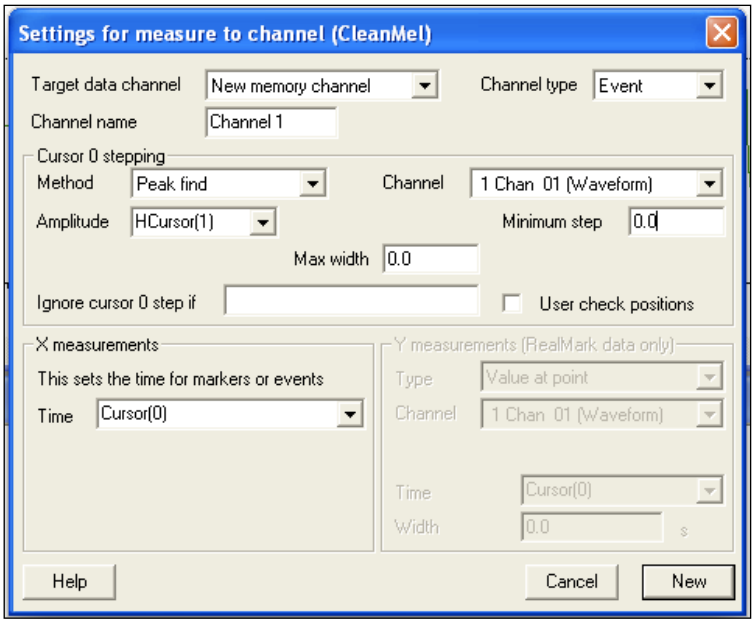
Event channels

Events are single points in time when something happens. We are interested in peaks in the song amplitude. We want one channel showing all peaks above a threshold (henceforth 'spike event') and one showing only the first peak of a song pulse ('pulse event').

Spike2 can auto-detect peaks, which allows some flexibility on the relative amplitude of each song pulse. Alternatively it is possible to manually set a particular amplitude threshold which will signify a peak every time it is crossed. I have found that a strategy between these two extremes works best.

Define event channels

- Create a new horizontal cursor (it will be called 'HCursor(1)' by default).
- Zoom in so that each pulse is clearly visible (as in Figure 1).
- Set the horizontal cursor position so that most peaks are above it (it is not necessary to be extremely accurate, Spike2 will detect peaks around the cursor).
- Generate a new data channel (click 'Measurements'  choose 'Data channel...'). A window as in Figure 2 appears.
- **Figure 2**: Channel measure settings



Spike event

- Set 'Target data channel' to the first available channel number[1]
- Set 'Channel type' to 'Event'
- Set 'Method' to 'Peak find'
- Set 'Amplitude' to 'HCursor(1)'
- Set 'Minimum step to '0.002'[2]
- Tick 'User check positions'.
- Click 'New', 'Process' and 'Yes to all' in the new windows that appear.
- Visually check that events were detected in bursts of any amplitude across the song.

Burst event

We will make two channels. The first is fully automatic, like above, and will be used to indicate the mistakes Spike2 makes, making it easier to visually avoid them in the second channel in which each event is manually confirmed. The manual confirmation is the longest step in song analysis.

- Generate a new data channel
- Set 'Target data channel' to a memory channel
- Set 'Minimum step' to 0.022
- Repeat as above, until a new channel is automatically generated
- Generate yet another data channel
- Set 'Target data channel' to the next available channel number
- Leave 'Minimum step' to 0.022
- Repeat as above. This time, instead of 'Yes to All', manually click 'Yes' or 'No' depending on whether the detected peak belongs to a unique pulse[3]. The automatic pulse event channel combined with the spike event channel will resemble a 'space invader' when additional/wrong events have been detected in the automatic channel, making it easy to eliminate them (Figure 3). Rapid clicking results in a space invader animation, adding some taste to the otherwise tedious repetitive clicking.
- Once done, delete the memory channel so that it is not used by mistake in subsequent analysis (right click on channel name and choose delete).

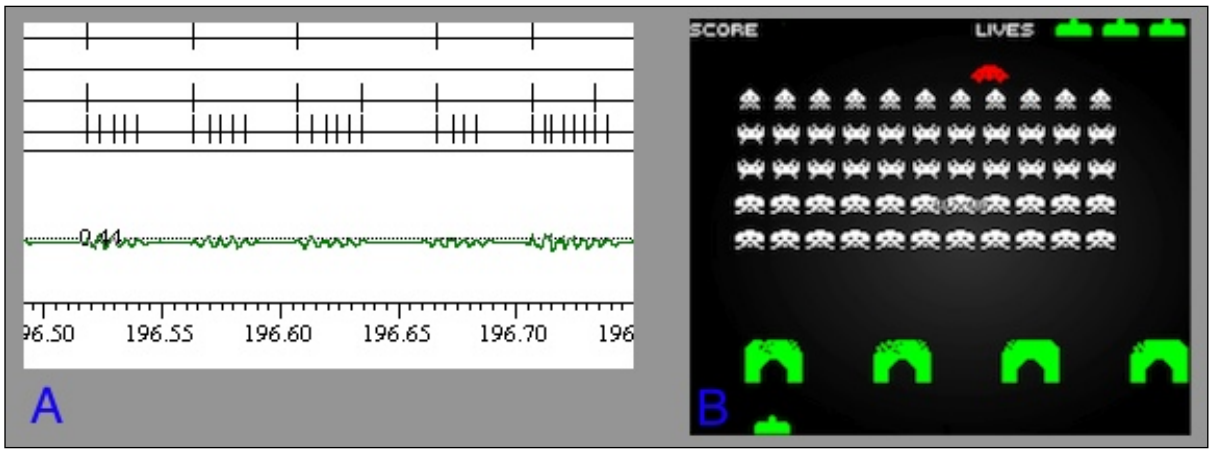
---

[1] Memory channels can be edited, but are not saved when saving the file. They are also prone to be randomly lost by bugs while running the burst script. Avoid using them.

[2] This length works well with *D. montana* song and can sometimes correctly interpret possible double peaks in a recording. Other songs will require a different value.

[3] I have found that pulse length tends to increase as the song progresses. Consequently, there are more mistakes made by Spike2 towards the end of the song. a) some mistakes mess up the following event detection, because of the minimum step use. b) the resulting event channel cannot be corrected, be careful when clicking.

Figure 3: False positive events. It is easy to visually detect false-positive events for the pulse event channels. Panel A shows from the top one manually confirmed pulse channel, the automatic pulse channel and the event channel. The first two pulses are identified correctly, while the third pulse is not, it resembles a space invader, shown in panel B, and it needs to be eliminated by removing the event in the end.



Pulse channels

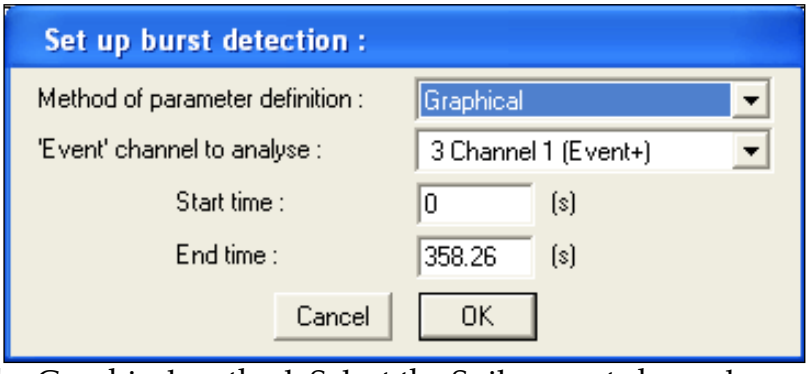
Pulse channels are based on event channels. They are defined by grouping events into groups termed 'bursts'. We will use the pulse event channel to generate a burst channel, which could be termed a 'pulse train burst'.

Define burst channels

All scripts are run from a special bar. If it is not visible select Script->Script Bar.

- Click on 'Burst' on the Script Bar[4]. The Script Bar will give place to a list of buttons.
- Click 'Mark Bursts'. A window like Figure 4 appears.

Figure 4: Burst detection first window

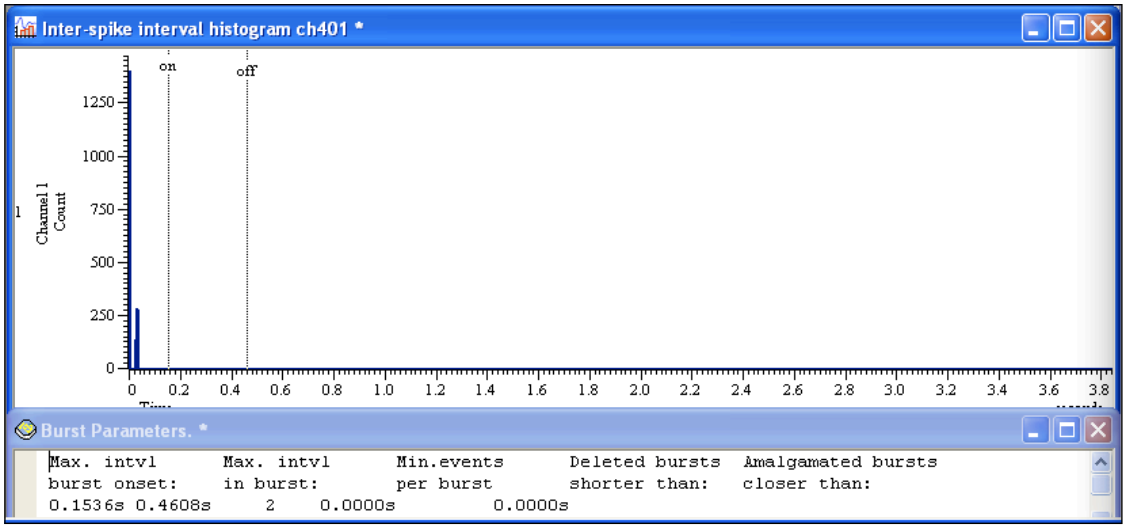


- Choose the Graphical method. Select the Spike event channel.
- Two new windows appear. One with summary statistics and one histogram of the time interval between bursts. The histogram has an 'on' and 'off' horizontal

[4] You will need to install the burst script, you will only need to do this once. It is found in the 'scripts' folder in the Spike2 installation folder (default path: `c:\Spike6\scripts\bursts.s2s`). To load it, go to Script->Scripts Bar List... and click 'Add'.

bar. See Figure 5 These can be moved to define when a burst should start and end. Move these as close as possible to 0 and click 'Update'. Once the burst channel appears to indicate pulse trains, click 'Accept' in the 'Script Bar'.

Figure 5: Inter spike interval histogram for the peak event channel. The on and off vertical cursors need to be moved as close to 0 as possible, without touching a bar.



- The next graphical window shows 'Distribution of spikes per burst'. Similarly to before, false positive results can be eliminated by indicating the minimum number of spikes an acceptable burst can have. When ready click 'Accept' in the Script Bar. Typically no intervention is needed.
- Similarly, false positive bursts can be eliminated using the newly appearing histograms. The song file can be inspected throughout the process to make sure that only real song events are detected as bursts (may need zooming out). Once satisfied click 'Ok' on the Script Bar.
- Click 'Accept' in the Script Bar.
- Click 'Save Bursts' in the Script Bar.

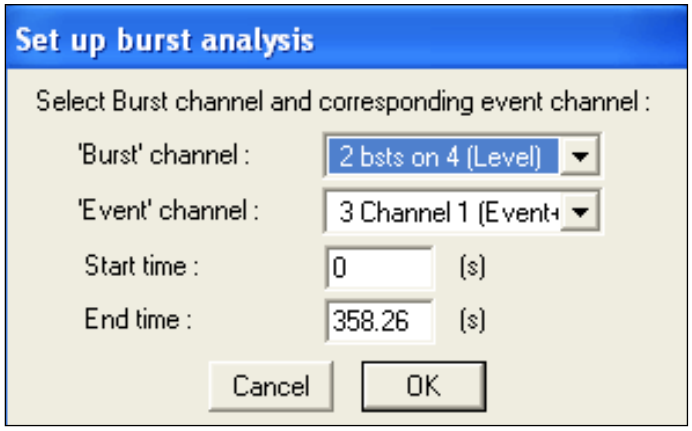
Song component calculations

Pulse Train Length (PTL)

While running the burst script, click 'Analyse bursts'.

Click 'OK'. A 'Set up burst analysis' window appears (Figure 6). Select the peak channel in 'Event channel' and click 'OK'.

Figure 6: The 'Set up burst analysis' window. It can be used to calculate statistics comparing burst and event channels, but we are not using this functionality. The peak event channel should be chosen for 'Event' channel in the dialogue.



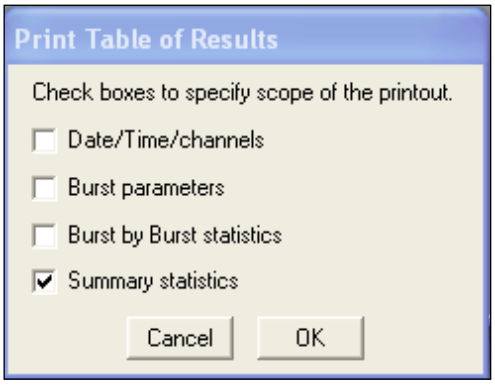
Click 'Show stats' in the script bar.

Uncheck the boxes as in Figure 7 and click 'OK'.

Record 'Mean burst length (ms)' (PTL) and 'SD burst durations' (PTLSD) in the Excel file.

Click 'Hide Stats', 'Back', 'Quit', 'Yes' in the Script Bar.

Figure 7: Options to click on the 'Print Table of Results' dialogue



Pulse number (PN), Pulse Train Number (PTN), Cycle Number (CN)

In the song file with the Spike event, generate one vertical cursor in the very beginning and one in the very end of the song.

Click 'Cursor regions'  and choose 'Sum'.

The largest number (in peak event channel) is total song cycles ('spikes').

The second largest number (in pulse event channel) is total pulses ('pulses').

1/2 of the smallest number (in burst channel) is the number of pulse trains ('PulseTrains').

Copy these to an excel spreadsheet which can be programmed to automatically calculate:

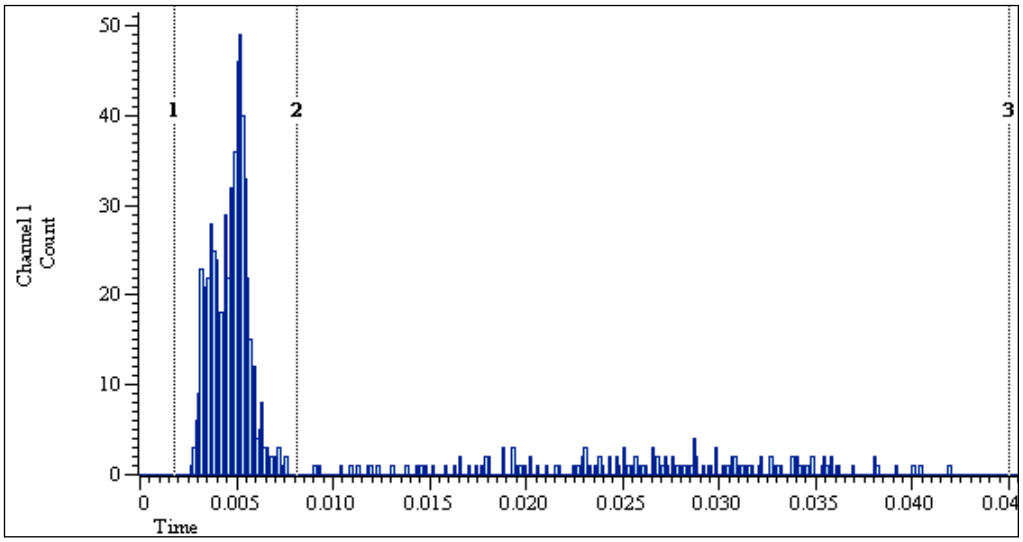
PN=spikes / pulses

CN= pulses / pulse trains

Cycle Length (CL = 1/FRE) and Pause Length (PAU)

- Click 'New Result View'  .
- Choose 'Interval Histogram'.
- Choose the ***peak event channel***. Set 'Bin Size' to '0.00125'.
- Click 'New'' and then 'Process' in the new window.
- A New window appears with a histogram of the time between the peak events. A typical output is shown in Figure 8. The normally-distributed distribution to the left is the CL, and sometimes it may be bimodal.
- Generate 3 vertical cursors and position them as in Figure 8.
- Click 'Cursor regions'  and choose 'Mean in X'. CL (1/FRE) is the number obtained between cursors 1-2, PAU is the number obtained between cursors 2-3. Also record the output of 'SD in X' as CLSD and PAUSD in the Excel file.

Figure 8: Interval histogram for the peak event channel.

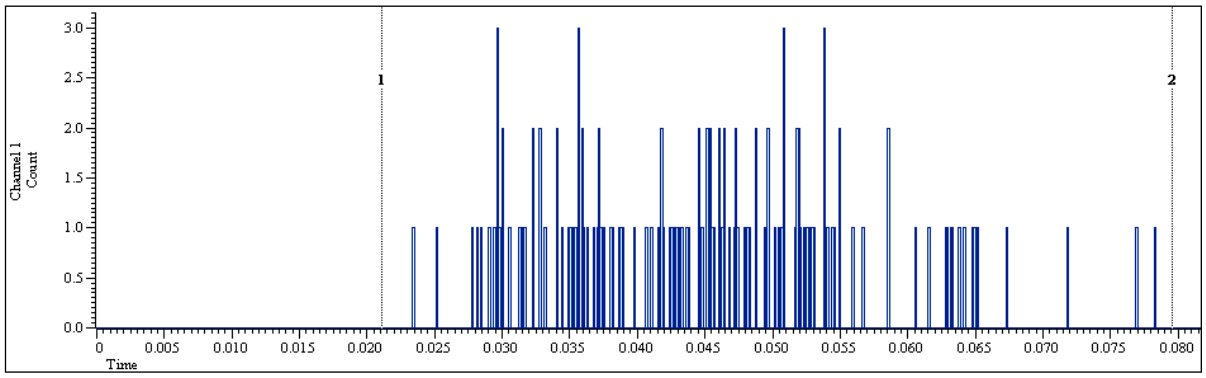


InterPulse Interval (IPI)

- Click 'New Result View'  .
- Choose 'Interval Histogram'.
- Choose the ***pulse event channel***. Set 'Bin Size' to '0.00125'.

- Click 'New" and then 'Process' in the new window.
- A New window appears with a histogram of the time between the peak events. A typical output is shown in Figure 9.
- Enclose all bars in two vertical cursors and position them as in Figure 9.
- Click 'Cursor regions'  and choose 'Mean in X'. This is the IPI. Select 'SD in X' and record it as IPISD in the excel file.

Figure 9: Interval histogram and vertical bars, for a pulse event channel.



Pulse Length (PL)

IPI = PL + PAU => PL = IPI - PAU

IPI and PAU were calculated above, so PL is easily calculated by inference.

Song character relationships

I no longer think all song components need to be calculated, because some correlate with each other. All information on pulse song is provided by FRE, IPI, PN and CN, as shown in Figure 10.

Figure 10: Illustration song components measured in the literature. Only those that do not correlate (in circles) need to be measured.