BPA Laboratory manual
Neil Cronin

This manual was written as part of a teaching Development grant that I received at the end of 2015. The goal was to develop a lab manual that served as a set of instructions for how to use several of the devices in our labs. Hopefully this will be useful for your thesis projects, as well as any other testing you may do in our labs or other labs in the future.

Although you will find step by step instructions here, you are encouraged to use your own initiative and expand upon what you learn by applying the techniques to your own research questions. This manual should be sufficient to help you get started with data collection, but you will not become an expert unless you gain a lot of experience and really explore the software and hardware.

You may notice that there is some overlap in the content and that some functions are mentioned more than once. This is intentional so that you don't need to read the entire manual if you don't want to- instead you can just skip to the sections most relevant to you. However, if you are new to data collection, and in particular, if you don't have much experience with Spike2, it is a good idea to start with the 'Basics of Data Collection' section.

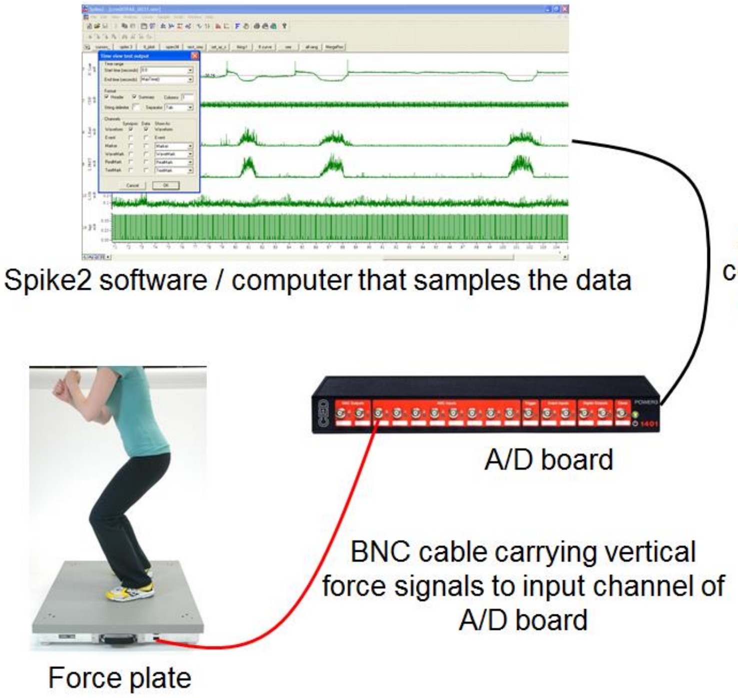
To use this manual you will need to have some data that you have collected or some kind of sample data. You can find some useful examples in the Spike2 data folder, usually located at C/Program Files/Spike6/data.

If you have any feedback on this manual, or there is something missing that you would like to learn that isn't here, please feel free to contact me: neil.j.cronin@jyu.fi.

Contents

Basics of data collection in Spike2

For most applications in our labs, you will use some kind of measurement device combined with an A/D board, Spike2 software (or similar), and a computer. A simple schematic of this setup is shown below using a force plate as an example.


Spike2 software / computer that samples the data
A/D board connected to computer
A/D board
BNC cable carrying vertical force signals to input channel of A/D board
Force plate

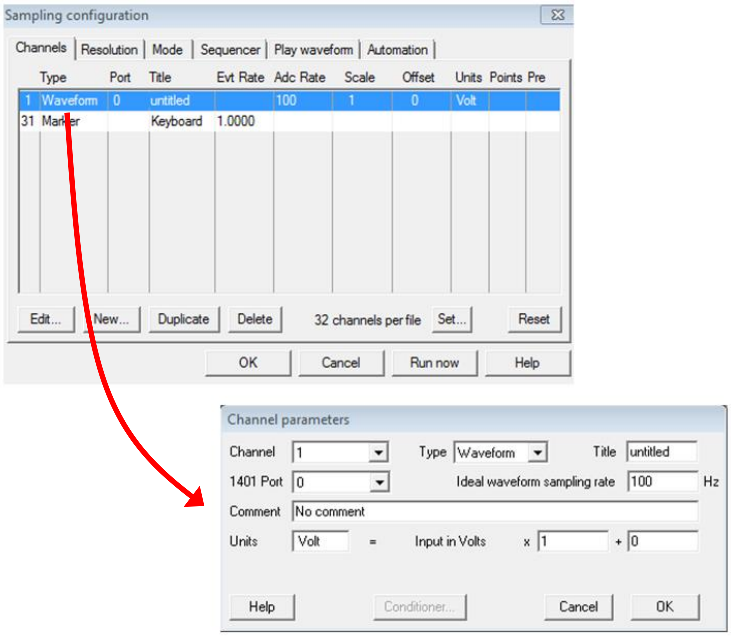
In some cases, it may be necessary for the BNC cable coming from the measurement device to be plugged into an amplifier, and the amplifier will then be connected to the A/D board. Otherwise this basic setup is very similar for most applications in our labs. Data that we sample in a lab are in analog form, usually as an electrical signal. The A/D board takes this data and converts it to digital form using binary logic, which is the main language of computers. It is then much easier to work with the data.

***

Note: In order for your measurement setup to work, the A/D board must be connected to the computer where you are running Spike2 software via a USB cable. Otherwise the signals you are trying to measure will never actually make it to the computer!

***

In order to collect data, you will first need to make a configuration file in Spike2. The aim of this file is to tell the software what kind of signals you want to measure, and how to measure them. You can open the Sampling Configuration by clicking on the shortcut button (📅), and you will then see something like the figure below.



	Type	Port	Title	Evt Rate	Adc Rate	Scale	Offset	Units	Points	Pre
1	Waveform	0	untitled		100	1	0	Volt		
31	Marker		Keyboard	1.0000						

Channel 1 Type Waveform Title untitled

1401 Port 0 Ideal waveform sampling rate 100 Hz

Comment No comment

Units Volt = Input in Volts x 1 + 0

In the Channels tab, you can setup all of the channels that you need for your measurements. In the example shown here, I have a keyboard marker channel and a single waveform channel. By double clicking the waveform channel, I am able to edit various parameters such as the number I would like to assign to this channel, which port I will use for this channel on the A/D board, the title of the channel, and my desired sampling rate. As there are several important things to understand here, we will go through the more important ones.

***
Note: In the sampling configuration, channel number and port are **not** the same thing. The port is the actual number of the port where your signal will be plugged into the A/D board, whereas channel number is just whatever number you want to assign to it. For example, you might have a force signal coming into port 1, but the channel number could be 6.
***

Type. There are several types of signal that you can acquire. The most common is *Waveform*, which is used to sample continuous data such as EMG, force etc. Another option that might be useful is *Event*, which you can use to keep a record of the precise timing of events (see below).

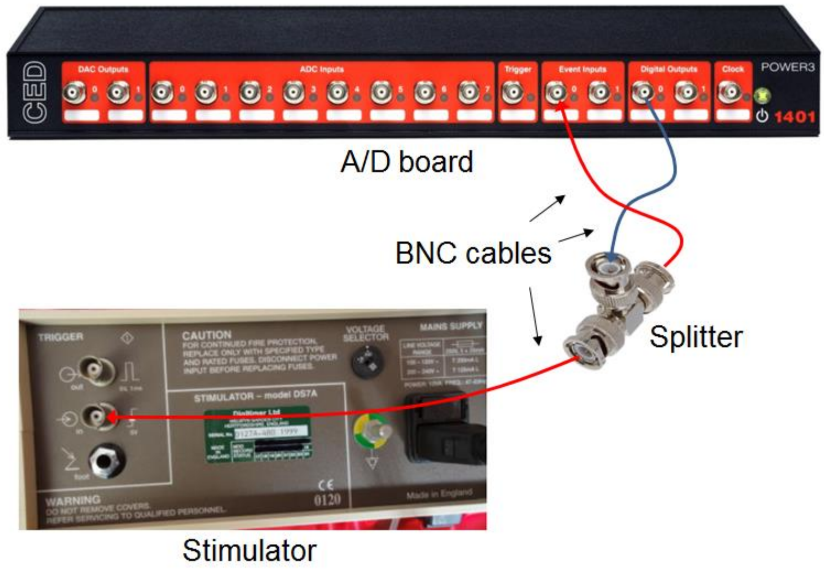
1401 Port. This is the port that you will connect a BNC cable to that carries data from the device you are measuring from. In other words, one end of the cable will be attached to your device (EMG system, force plate etc.), and the other end will be plugged into this port.

Ideal waveform sampling rate. It is important to sample at a sufficient frequency to avoid loss of information from the original signal. However, you should also avoid excessively high sampling frequencies because this uses a lot of computer memory and storage space. Note that what you ask for here is the ideal rate. If you are sampling from many channels and/or if your computer does not have much memory, it is possible that you request a rate of 2000 but the software only samples at, for example, 1700. This is a compromise that must sometimes be made in order to allow all necessary channels to be sampled. If you are only sampling from a few channels, you are unlikely to face this problem.

Input in Volts. There are 2 boxes here where numbers can be edited. In the example shown above, I have set those numbers to ‘x 1’ and ‘+ 0’. This means that when my signal enters the A/D board, I want Spike2 to multiply it by 1 (i.e. do not amplify it or reduce it) and add 0. In other words, I am keeping the signal as it is, which is generally fine for most applications. However, sometimes you may be sampling a signal that you want to amplify, or that has a known offset. These issues can be corrected by changing the x and + digits.

Using event channels

Event channels allow you to keep track of the exact timing of events. For example, if you are using electrical stimulation to evoke H-reflexes, you will want to know exactly when the stimulation occurred so that you can calculate, for example, H-reflex latency. For this example, you might first setup Spike2 to enable you to trigger the stimulator (see ‘The output sequencer’ in the ‘Basics of data analysis’ section below). To do this you would send a digital pulse to the stimulator, which would trigger it to elicit a stimulus. To identify exactly when this digital pulse was sent, you can split the output pulse and send it to both the stimulator and an event channel, as shown in the figure below. This means that whenever a digital pulse is elicited, it will appear in the event channel as a time stamp. You can later use this to set active cursor rules (see below), or in external software such as Matlab to make analysis easier.

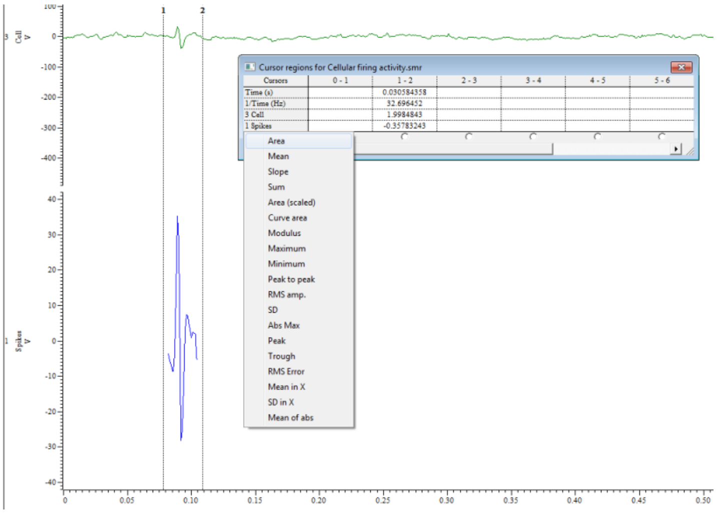


In this example, an alternative method of connecting the cables (without the need for a splitter) would be to plug a cable from the *Digital Output 0* channel into the *Trigger In* port on the stimulator, and then another cable from the *Trigger Out* port into the Event channel on the A/D board. This would work in the same way as the example shown. However, this option may not be available for all devices.

Useful features of the Spike2 interface

During data collection, there are various features that can help you to monitor the quality of your data, and to extract values. All of these are available from the dropdown menu at the top of the screen, which you should explore in detail. Here are some common functions:

Cursors. You can add a cursor by pressing Ctrl and the number of the cursor you would like, e.g. Ctrl and 1 brings cursor 1 into view. Cursors can be moved by clicking and dragging. Right clicking the cursor also allows you to change the cursor label, move its location or delete it. The same things can be achieved with horizontal cursors, which can be inserted from the dropdown menu using *Cursor → New Horizontal*. To make calculations within a region between 2 cursors, first set the cursors to the desired positions, then click the cursor regions button (⇅).

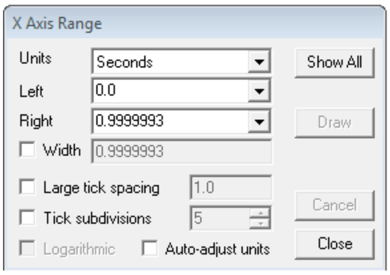


Cursor regions for Cellular firing activity.smr

Cursors	0 - 1	1 - 2	2 - 3	3 - 4	4 - 5	5 - 6
Time (s)		0.030584358				
1/Time (Hz)		32.696452				
3 Cell		1.9984843				
1 Spikes		-0.35783243				

The default calculation is mean, but by clicking the word *mean*, you can access a long list of different calculations. In the example to the left, there are 2 data channels, Cell and Spikes, and 2 cursors. The *Cursor Regions* box shows mean values between cursors 1 and 2 for both channels, as well as the time between cursors and the frequency of the cursor locations. These values will automatically be updated when you move the cursors. As you can see, there is also the possibility to add many more cursors and perform calculations between them.

Zooming axes. During a measurement, you might want to control how much data is visible at one time. By double clicking anywhere along the x axis, you will access the *X Axis Range* dialog box shown to the right. Here you can set a specific time range or a specific width of data to show. After making any changes, click *Draw* and then *Close*. Similar functions can be performed by clicking on the y axis.



X Axis Range

Units: Seconds
Left: 0.0
Right: 0.9999993
□ Width: 0.9999993
□ Large tick spacing: 1.0
□ Tick subdivisions: 5
□ Logarithmic
□ Auto-adjust units

You can also zoom in on a portion of data by holding down the left mouse button and dragging the cursor to cover the region you would like to zoom. To undo this action, press Ctrl + Z.

Autoscale. You will often notice that the scaling on the y axis is not correct, and that perhaps you can't see all of your data. This often happens if you have a force channel where no force is being applied, and then a subject increases the force rapidly. To correct the scaling, use the shortcut Ctrl + Q (or End depending on your keyboard). You can do this as often as is necessary.

Show/Hide Channel. By clicking this option from the dropdown menu (under *View*), you can choose which channel(s) to display. If you are using a saved configuration and would like to always show certain channels whenever you use this configuration, use *Show/Hide Channel* to display the channels that you want to appear, then resave your configuration. In the future, each time you run the configuration it will show the same channels.

***

Note: Channels that are set to be hidden do still exist, and data from these channels are still saved. At any time you can use *Show/Hide Channel* again to show previously hidden channels.

***

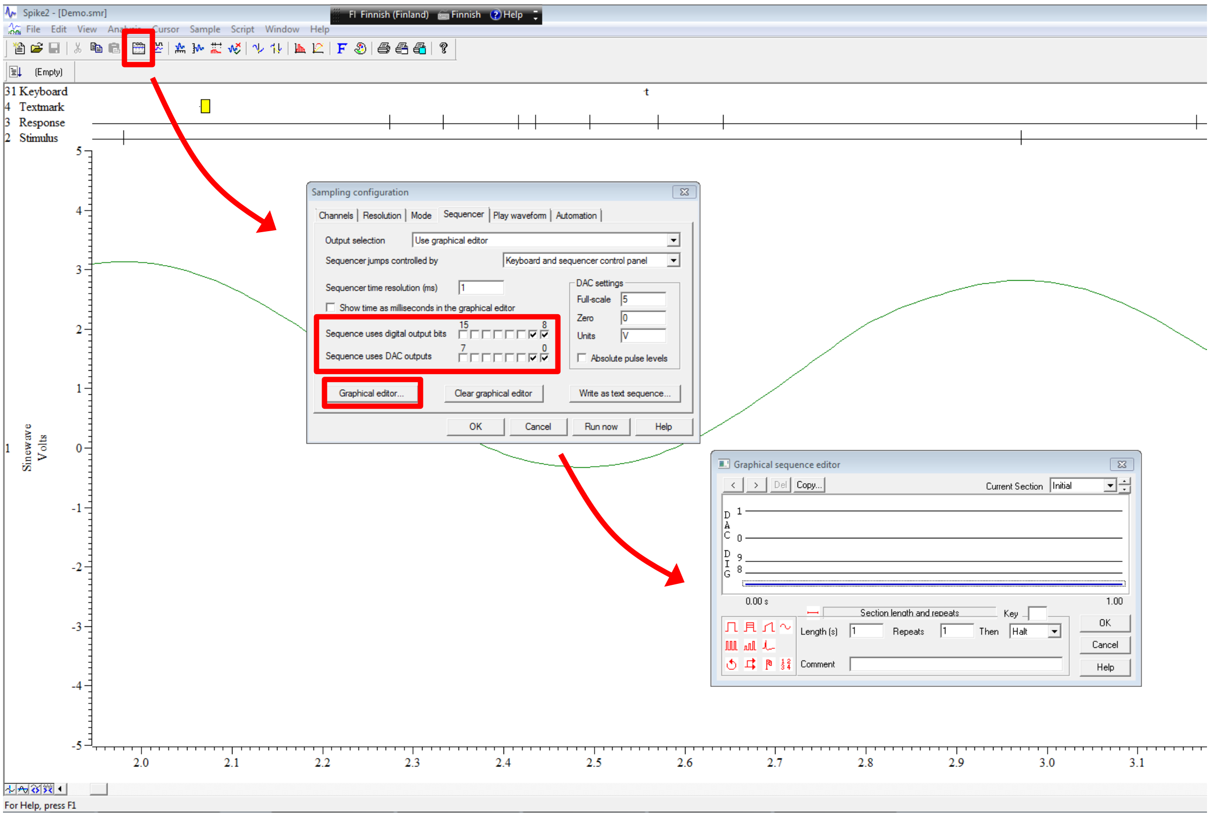
The Write Button. The write button can be found in the toolbar near the top of the screen during sampling (☑ Write). When the box is ticked, data are being sampled **and** saved to file. If the box is not ticked, the data are being sampled but not saved to file. This is a handy function because at the beginning of an experiment, you will often want to check signal quality. This information does not need to be saved, so by leaving the write box unticked during signal checks, the resulting saved file will take up less disk space and will only contain the data you need. Similarly, during an experiment, you might have long breaks in between test conditions. During these breaks you can untick the write box so that only the necessary data are actually saved.

Exporting Data. Data can be exported into various formats using the dropdown menu (*File → Export As*). In the dialog box that appears, click the *Save As Type* dropdown list to access possible file types to export to. The most common are *Spreadsheet text* (for working in Excel) and *Matlab*.

The Output sequencer

There are many uses of the output sequencer, but perhaps the most common is creating some kind of trigger pulse, e.g. to trigger an electrical stimulator to produce a stimulus.

Most electrical devices can be manually triggered. For example, an electrical stimulator or a TMS device both have trigger buttons on them. However, in a measurement setting, it is often more convenient to be able to trigger such devices remotely. This can be done in Spike2. First, open the *Sampling Configuration* (see red box in the figure below), click the *Sequencer* tab and under *Output Selection*, choose *Use Graphical Editor*. Next you should check which digital/DAC outputs are enabled. Here the DAC outputs are labelled 0 to 7 and the DIG outputs 8 to 15. In reality you won't have this many options available. You can think of channels 0 and 1 as the DAC outputs 0 and 1 on the A/D board, and channels 8 and 9 as the digital outputs 0 and 1 on the A/D board. For most triggering purposes, a digital output is most suitable. Make sure that the channel you want to use for your output is ticked. In the example below, I have activated 2 DAC and 2 digital outputs, but it is often enough to just tick the box for channel 8 (digital output 0 on the A/D board). After clicking the *Graphical Editor* button, you will have the chance to add your pulse(s).



Output selection: Use graphical editor

Sequencer jumps controlled by: Keyboard and sequencer control panel

Sequencer time resolution (ms): 1

☐ Show time as milliseconds in the graphical editor

Sequence uses digital output bits 15 ☐ ☐ ☐ ☐ ☐ ☐ ☑ ☑ 8

Sequence uses DAC outputs 7 ☐ ☐ ☐ ☐ ☐ ☐ ☑ ☑ 0

DAC settings: Full-scale 5; Zero 0; Units V; ☐ Absolute pulse levels

Current Section: Initial

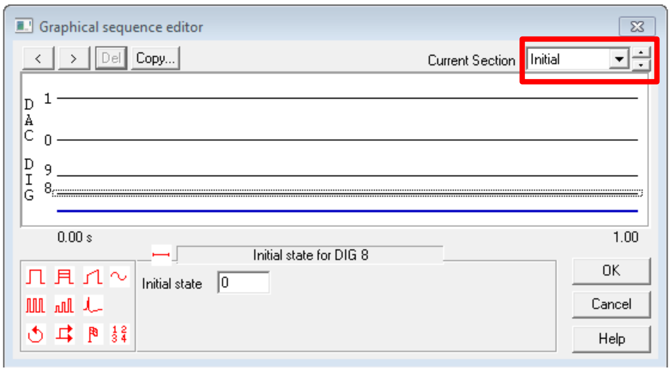
Length (s) 1 Repeats 1 Then Halt Key

Comment

***

Tip: A trigger pulse (i.e. a digital output) can also be useful as a time stamp of exactly when an event occurred. This is very useful if, for example, you want to calculate the latency between an electrical stimulus and an EMG response. For instructions on how to set this up, see ‘Using Event Channels’ above.

***



Current Section: Initial

Initial state for DIG 8

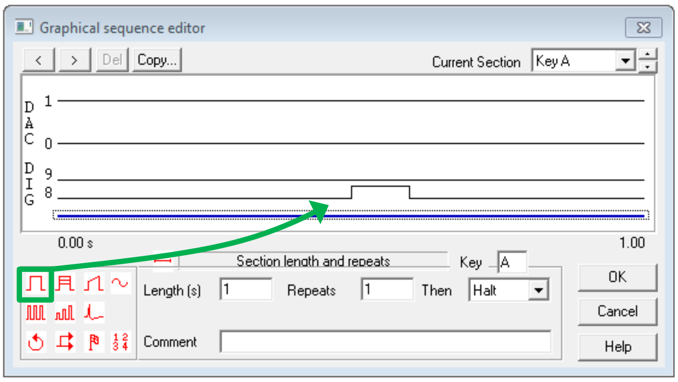
Initial state 0

To add a single digital pulse, first you need to find a section to store it in. You can choose a free section by clicking the dropdown menu next to the *Current Section* (see left). For this example I will use *Key A* (see figure below). Drag and drop a single digital pulse into the appropriate digital channel. In the example below, I used channel 8 (digital output 0 on the A/D board). My pulse is 100ms long and starts 500ms after I initiate it. I then clicked the blue line at the bottom of the sequencer, and in the *Key* box I typed ‘A’.

***

Note: In most cases, the duration of the pulse is not important because it is the rising edge that acts as the trigger. However, it is important to know whether the rising or the falling edge of the pulse serves as the trigger, as this will affect any time-based calculations you do.

***



Current Section: Key A

Section length and repeats

Key: A

Length (s): 1

Repeats: 1

Then: Halt

Comment:

When I run this configuration, an ‘A’ will now appear in the toolbar near the top of the screen. Whenever I want to use this as a trigger, I just press the ‘A’ button on the toolbar. For our example of triggering an electrical stimulator, I would connect a BNC cable to the digital output 0 port at one end, and to the ‘in’ port on my stimulator at the other end. Assuming the stimulator is switched on, pressing the ‘A’ key while Spike2 is running should cause the stimulator to fire a single pulse.

In the above example, I used a single pulse and stored it in *Key A*. Spike2 allows you to store different pulse parameters for Keys A to Z, so you have 26 different possible pulse combinations if you need them. For example, you might use Key A to store a single pulse, Key B to store a double pulse, Key C to store a series of 5 consecutive pulses and so on. You can then assign a shortcut key to each one, as we did with ‘A’ above, and to use each pulse configuration during a measurement you just press the corresponding key. Each time you assign a shortcut key, running the configuration will automatically add that key to the toolbar, so in this case you would now see A, B and C in the toolbar.

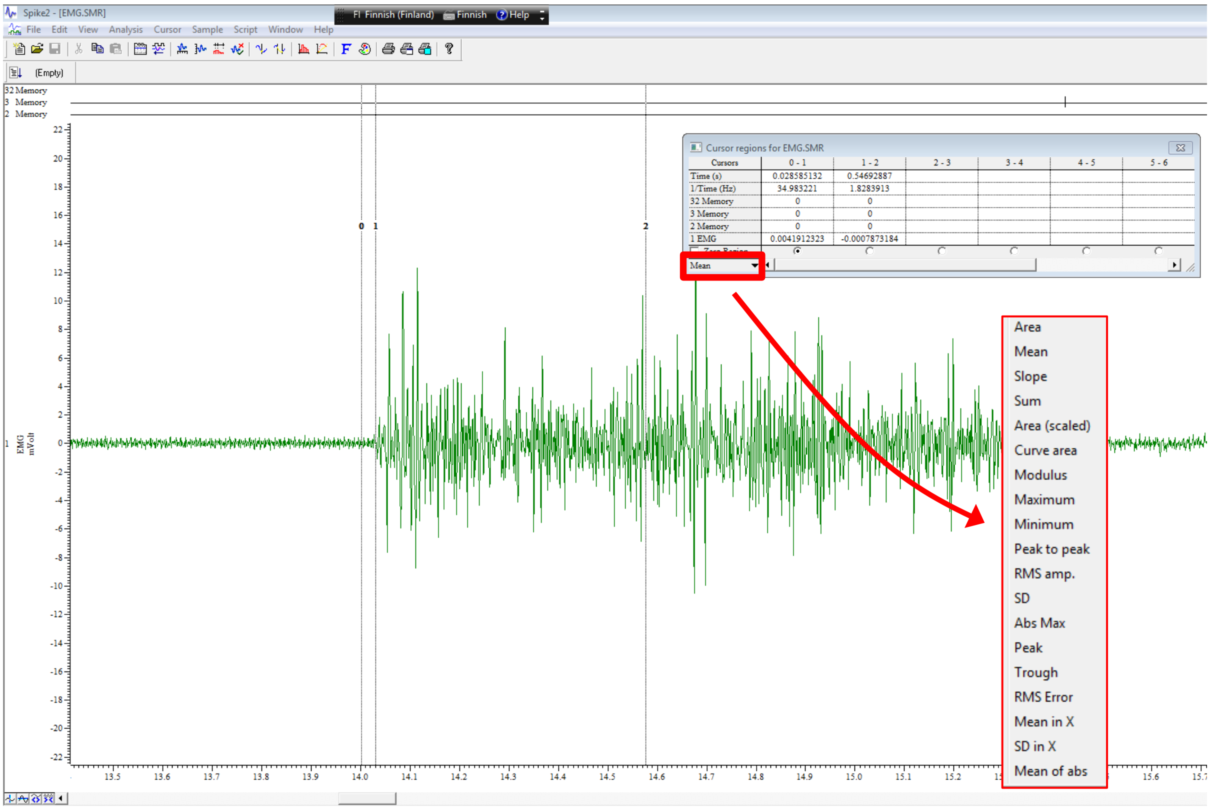
More information. The makers of the Spike2 software, CED, offer excellent instructions for the use of the software, as well as guidance on the use of scripts, data analysis etc. To access the main manual for version 6 of Spike2 (which is the version we use at the time of writing), see the following link: http://ced.co.uk/img/Spike6.pdf.

Practice

1. Using demo data or data that you have collected, explore the Spike2 interface and try out the following basic functions: Insert a vertical cursor; Rename the cursor using a custom label; use cursor regions to calculate area within a window between 2 cursors; export some data to an excel spreadsheet and plot a graph in Excel. Get to know the functionality of the various items on the dropdown menu / shortcut buttons. This can save a lot of time later on.
2. Use the output sequencer to setup the following:
- Key A, section length 5s, 3 digital pulses starting at 2s, pulse duration 0.1s, 1s between pulses, shortcut key ‘A’
- Key B, section length 8s, pulse train with 10 pulses starting at 2s, pulse duration 0.1s, 0.5s between pulses, shortcut key ‘T’

Basics of data analysis in Spike2

Spike2 is useful for a range of analysis tasks, especially when the volume of data you have is not too large. The simplest analysis involves setting cursors and measuring values that are somehow related to those cursors. For example, in the figure below, we set 3 cursors numbered 0, 1 and 2. We then select the *cursor regions* button (⇅) from the toolbar (or *Cursor* → *cursor regions* from the dropdown menu). In the box that appears, we see the mean values between cursors 0-1 and 1-2. By clicking the *Mean* button, a dropdown list appears, and we can choose from a number of common functions (see inset below).



Cursor regions for EMG.SMR

Cursors	0 - 1	1 - 2	2 - 3	3 - 4	4 - 5	5 - 6
Time (s)	0.028585132	0.54692887				
1/Time (Hz)	34.983221	1.8283913				
32 Memory	0	0				
3 Memory	0	0				
2 Memory	0	0				
1 EMG	0.0041912323	-0.0007873184				

This function is handy for individual measurements, but it is not very convenient to have to repeat this process manually hundreds of times. Another very useful method that allows you to somewhat automate processes like this is **Active Cursors**.

***

Tip: To place vertical cursors, use the keyboard shortcut Ctrl + X, where X is the number of the cursor you would like to generate, e.g. Ctrl + 2 to bring Cursor 2 into view. If cursor 2 does not exist, it will be created. If it exists but is currently out of view, it will be 'fetched' into view. You can also create cursors from the dropdown menu using *Cursor* → *New cursor*, and fetch them using *Cursor* → *Fetch*.

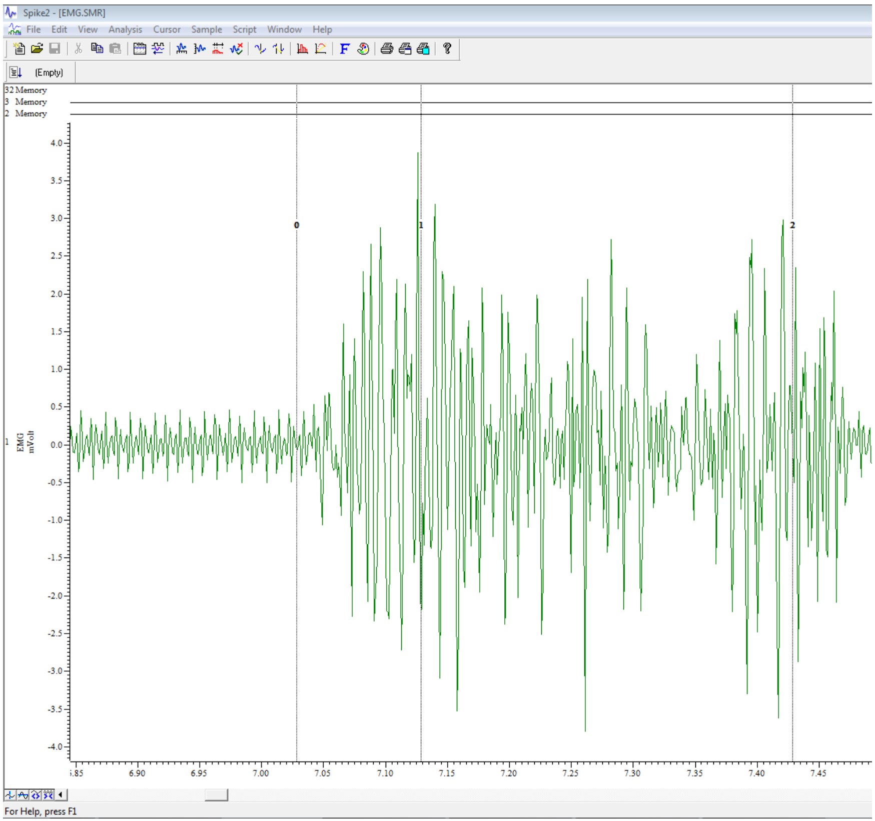
***

Active Cursors

The easiest way to use this function is to have some kind of trigger signal that can be used as a reference. In the example below, I use a digital pulse as my trigger, but you can also use the keyboard channel, textmarks etc. You must first set the 'rules' for the cursor positions. Find or create cursor 0, right click it and choose *active mode*. Use *data points* as the search method and choose your event channel containing the trigger pulse(s) as the search channel (I used *memory 2* but *memory 3* would have also worked).

***
Tip: So far I have only mentioned vertical cursors. To position a horizontal cursor instead, use the dropdown menu (*Cursor → New Horizontal*).
***

You can then set the active mode for other cursors in relation to cursor 0. For example, let's assume that we want to measure mean EMG between 100-400ms after a trigger (see below). I've already set the rules for cursor 0, which will look for the trigger pulse. Then I will set cursor 1 to be placed 100ms after this pulse, and cursor 2 to be placed 400ms after the trigger. To do this, I create/fetch cursor 1, right click it and choose *active mode*. This time, instead of *data points* as a search method, I choose *Expression*. In the expression box that appears below, I type 'Cursor(0) + 0.1'. This tells Spike to first position cursor 0, and then position cursor 1 so that it is 100ms later (i.e. 0.1 seconds). To set the rules for cursor 2, I use the same approach, but this time I type 'Cursor(0) + 0.4', or 'Cursor(1) + 0.3'. Once the rules have been set, I press Ctrl + Alt + right arrow key (or Ctrl + shift + right arrow key depending on your keyboard). This snaps the cursors into position, i.e. finds the next trigger pulse and places cursor 0 there, then places cursors 1 and 2 relative to this. The result looks like this:



In this example, cursor 0 is covering the trigger pulse in the *memory 2* channel, cursor 1 is 100ms after cursor 0, and cursor 2 is another 300ms later. Now that the cursors have been positioned, we could just click the *cursor regions* button and perform any analyses we like, such as calculating the mean value between cursors 1 and 2.

Assuming you have more than one trigger pulse in your data file, you can skip to the next one by using the Ctrl + Alt + right arrow key combination, and the cursors will automatically be repositioned. You can also go in the other direction by pressing the left arrow key instead of the right one.

***
Note: In the example shown above, I only used cursors 0, 1 and 2, but you can use as many as 10 cursors if you need to. Just remember to set the rules for each cursor individually.

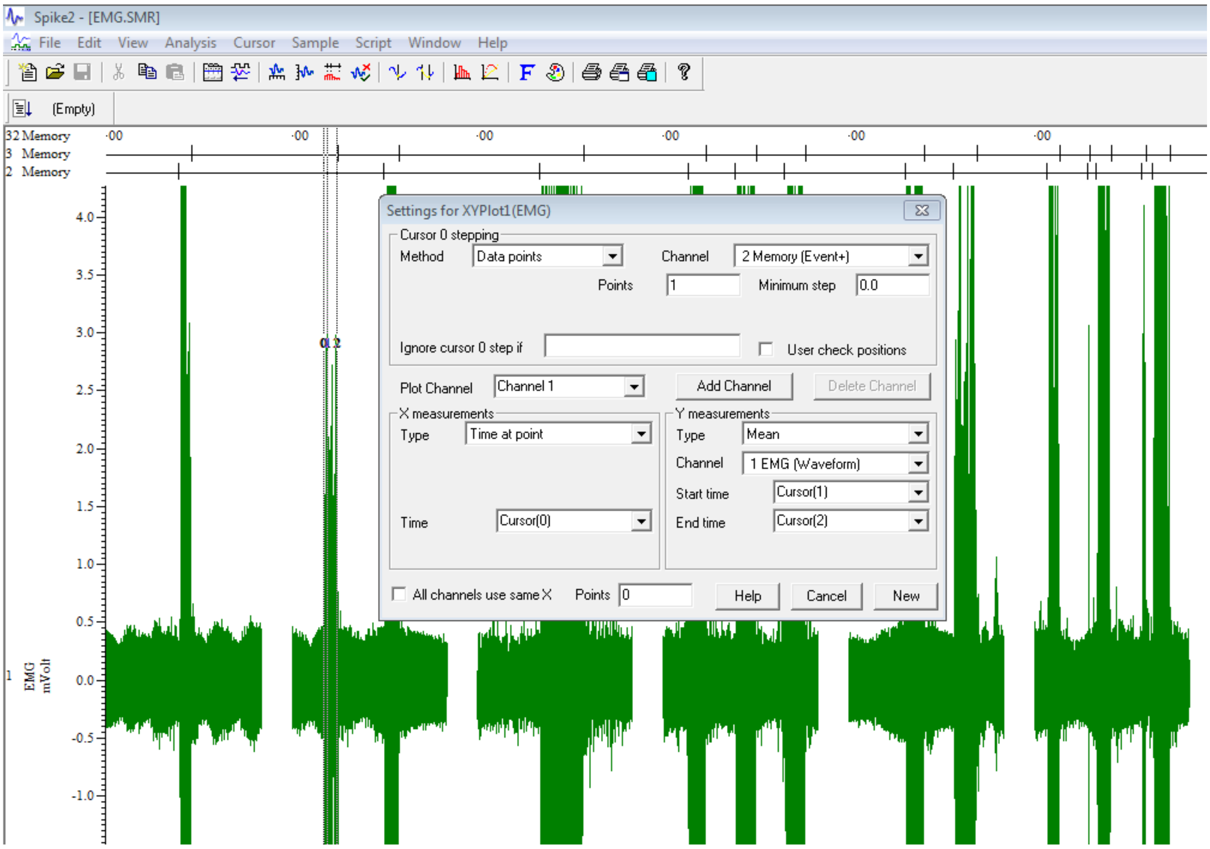
***

As well as the *Data Points* and *Expression* methods outlined above, there are several other ways of using active cursors. After right clicking a cursor and selecting *Active Mode*, clicking the dropdown list next to the *Search method* label will show you all of the possible options. For example, you can set the cursor to look for all values above or below a specific threshold, as well as values within a specified range.

The active cursors method makes it relatively easy to perform a standardized analysis many times throughout a data file. However, as it is, you would need to find each trigger pulse using the keyboard shortcut, click the *cursor regions* button, and then manually write the value(s) down somewhere or copy-paste them to a spreadsheet. This is not very convenient if you have a large number of trigger pulses/trials in your file and you plan to do further analysis. A useful method of easily gathering multiple values together is to create an XY view.

Creating an XY view

Click the ‘measurements’ icon ( ) and choose ‘XY view’ (or *Analysis* → *Measurements* → *XY view* from the dropdown menu). You will then need to setup your plot using the box that appears (see below).



Settings for XYPlot1(EMG)

Cursor 0 stepping

Method: Data points — Channel: 2 Memory (Event+)

Points: 1 — Minimum step: 0.0

Ignore cursor 0 step if: — □ User check positions

Plot Channel: Channel 1 — Add Channel — Delete Channel

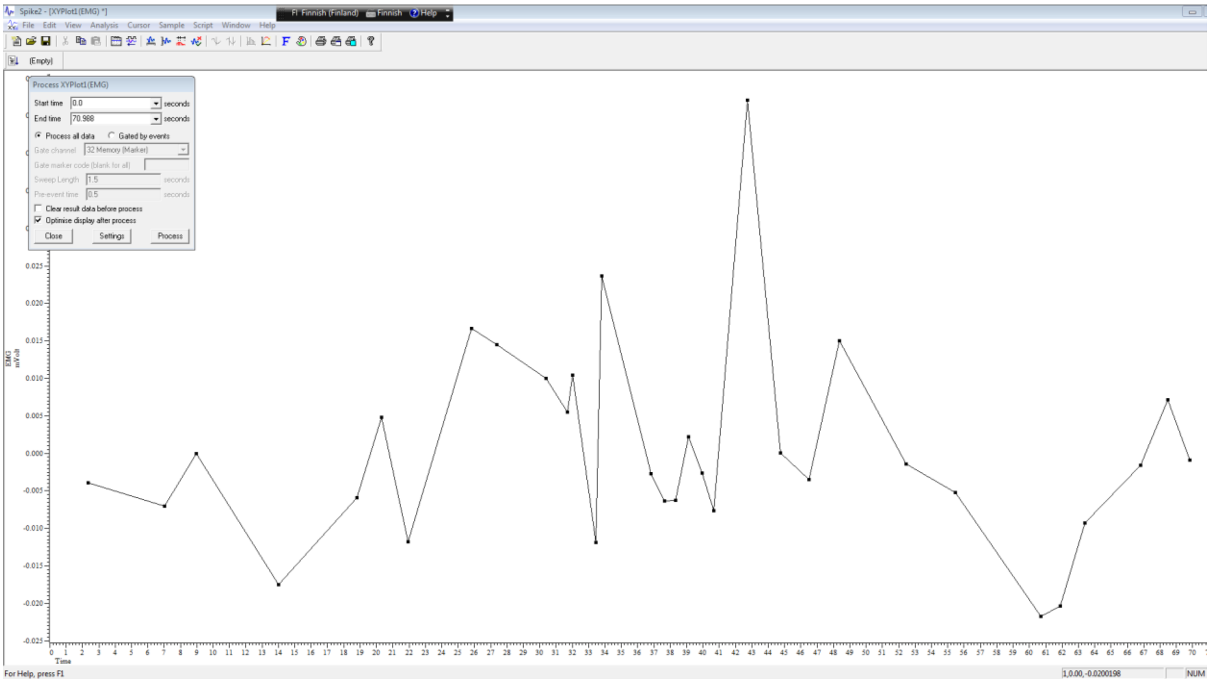
X measurements: Type: Time at point — Time: Cursor(0)

Y measurements: Type: Mean — Channel: 1 EMG (Waveform) — Start time: Cursor(1) — End time: Cursor(2)

□ All channels use same X — Points: 0 — Help — Cancel — New

In this example, the event (trigger) channel is used as the reference. Cursor 0 is moved according to this (and all other cursors move relative to cursor 0). The *X measurements* box looks for time, and the *Y measurements* box is set to calculate the mean of the EMG channel between cursors 1 and 2. This is enough for the purpose of this example, but if I wanted to do more analyses, e.g. also calculating the area between cursors 1 and 2, I could choose ‘add channel’ and repeat what I have just done, but choosing area instead of mean as the *Y measurements* type. Once you have setup all channels that you want, click *new*. In the window that opens, check that the start and end times include the whole portion of your measurement, then click ‘process’.

The result will look something like this (for this example I used the ‘EMG.smr’ file from the Spike2 data folder):



Process XYPlot1(EMG)

Start time 0.0 seconds

End time 70.988 seconds

● Process all data ○ Gated by events

Gate channel 32 Memory (Marker)

Gate marker code (blank for all)

Sweep Length 1.5 seconds

Pre-event time 0.5 seconds

□ Clear result data before process

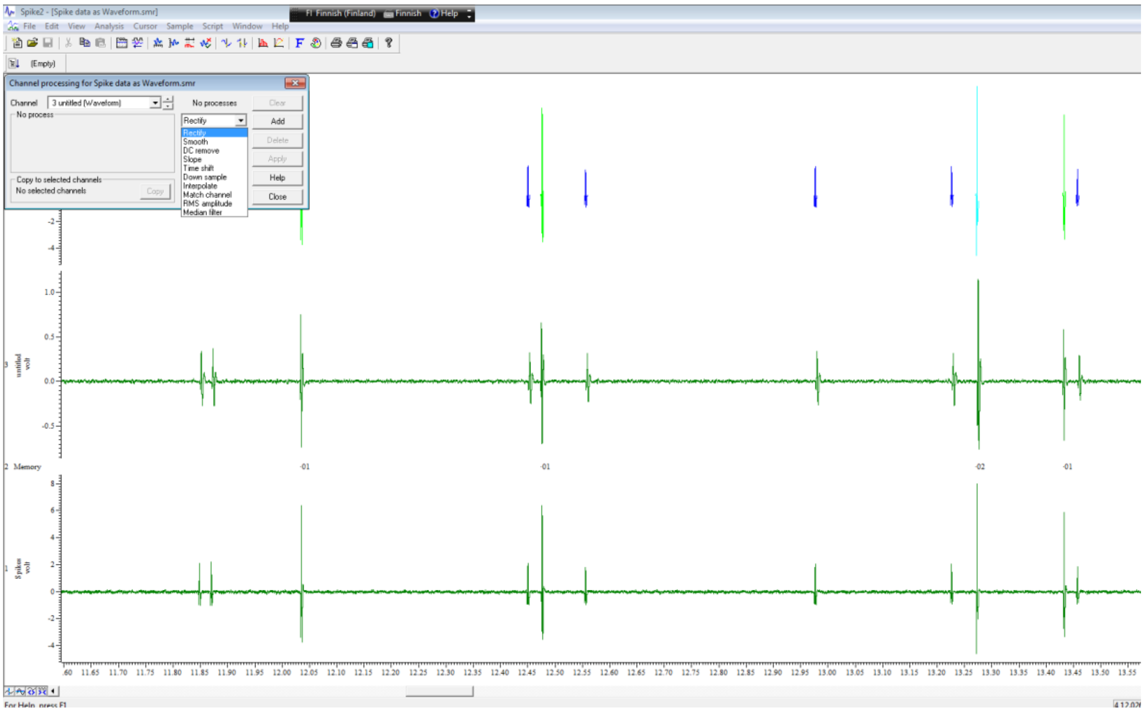
☑ Optimise display after process

Spike2 has gone through the entire file looking for each trigger pulse in my designated trigger channel. It has then positioned cursors 0, 1 and 2 according to the rules that I set earlier, and computed the mean EMG value between cursors 1 and 2. The resulting plot shows these mean EMG values for every trigger pulse in the file (in this case 32). Assuming I want to do further analyses with these numbers, I can easily copy them to a spreadsheet using the dropdown menu (*Edit → Copy as text*) and then paste them to excel.

I could have achieved the same results by just cycling through the pulses one by one (using Ctrl + Alt + right arrow) and copying each mean EMG value into a spreadsheet, but it would obviously have taken much more time. However, it is also important to note that, as with any automated analysis, it is vital to also manually check your data. In the figure above, there is large variation in the size of the different responses, and from this plot it is impossible to say whether this is real variation or whether there are some anomalies in the data.

<u>Other useful features of Spike2</u>

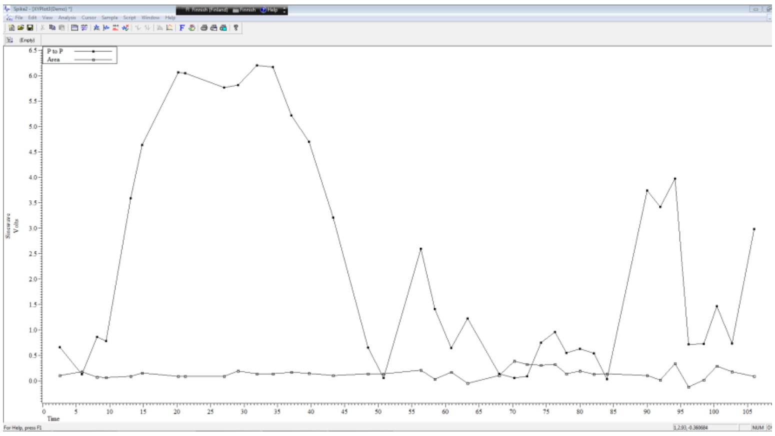
Channel process. There are a number of ways you can modify data that you have recorded. By right clicking anywhere within the channel you’d like to edit, you are shown a dialog box with several options. Choose *Channel Process*, and then select the channel of interest and the process you’d like to apply, followed by *Add*.



This is useful for basic functions such as filtering or rectifying EMG. If you want to keep the original data but also look at a filtered version, then you can first duplicate the channel by right clicking anywhere within the channel of interest and selecting *Channel X → Duplicate Channel*. You can then apply the filter to the duplicated version, allowing you to easily compare the original and filtered data.

<u>Practice</u>

1. Open the data file 'Demo.smr' that can be found in the examples provided by Spike2. Using the keyboard channel as a trigger, set active cursors so that you can measure the *peak-to-peak* of Channel 1 within the time window between the trigger and 100ms after the trigger (i.e. a 100ms window), and calculate the *area* in the time window 100ms after trigger : 200ms after trigger (also a 100ms window). Make an XY plot of both of these results. It should look something like the figure above.



2. Using the data file 'Blood Pressure Waveform' supplied by Spike2, use active cursors to locate all of the peak values in the 'AP Wave' channel. Hint: For this to work, you need to use the correct search method **and** amplitude. Try using different variations until you consistently find the peaks.

Surface Electromyography (EMG)

Surface electromyography (EMG) is a method that allows us to indirectly study electrical signals from muscles that are close to the surface of the skin. The method requires electrodes to be positioned over the skin above the muscle of interest. Although it is associated with several limitations, this method provides a safe, cheap and noninvasive estimate of muscle activity that can be used in a wide variety of test situations. In this section we will cover important issues associated with preparation, data recording and some basic analysis.

Preparation. The materials needed to prepare for an EMG measurement are shown below.

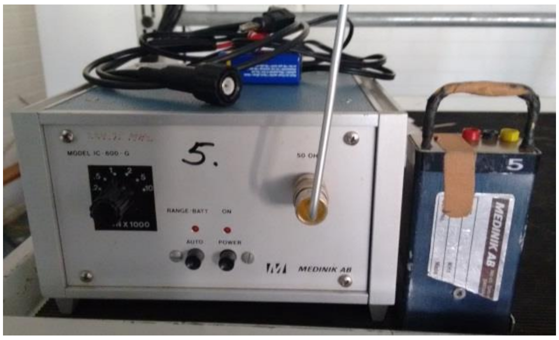
You should first shave the area where the electrodes will be positioned, making sure to remove all hairs. You then sand the area of skin to remove the outermost skin layer, which helps to reduce impedance between the skin and the electrode. It is important not to sand too much skin away. If bleeding occurs, you should not measure from this site. Finally you clean the area using antiseptic. Although this is not essential (and some scientists even suggest that signal quality is better without performing this stage), it is standard practice in our lab to do this. Once the skin is dry, you can attach the electrodes. You can also repeat this process for the ground electrode if the system you are using requires a separate ground. As a general rule, you should follow SENIAM guidelines on where to place electrodes: http://seniam.org/sensor_location.htm

***

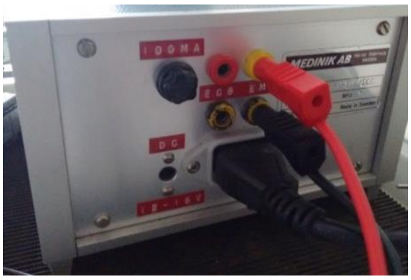
Note: To keep the lab tidy and to maintain good hygiene, do not reuse razors on different people, and make sure that used razors are disposed of in the yellow plastic bins in the lab. Other used items such as sandpaper should be put in a normal bin.

***

Setup. Assuming you are working in the Liikunta demo lab, you will use the Medinik EMG system to collect data. Note that in the figure to the right, both the pre-amplifier and amplifier have a number '5' on them. It is important to use matching numbered units because they are tuned to work at a specific frequency. So for example, pre-amplifier



number 5 will not work with amplifier number 6.



The EMG electrode should be plugged into the pre-amplifier using the red and yellow ports for active electrodes, and the black port for the reference electrode if available. The amplifier should then be connected to the A/D board by plugging cables into the ‘EMG’ ports on the back of the amplifier (see figure to the left). The other end of these plugs is a BNC connector that goes to the A/D board.

Signal checking. Before connecting any cables, you can first check the skin resistance using an Ohmmeter (or resistance meter). Ideally the resistance will be less than 5 kΩ. If you see a high value at first, wait a few minutes and measure again- it will often decrease within a few minutes of the electrodes being applied. If the resistance is still high after several minutes, remove the electrodes and repeat the preparation process. There is no value in measuring an EMG signal unless you have done everything you can to maximize its quality.

Once you are happy with your electrode placement, you can connect the EMG cables and start to check signal quality. Run your configuration file in Spike2. For signal checks, you can leave the *Write* box unticked, since you won’t need to keep these data. First, ask the subject to completely relax the muscle and then check that the EMG signal is around zero. Then ask them to contract the muscle and check that you see a burst of activity in the EMG trace. You could also try a functional test such as asking the subject to walk on the spot for several seconds to check that the signal is somewhat repeatable between steps. Once you are satisfied with your signal quality, you can check the *Write* box and begin testing.

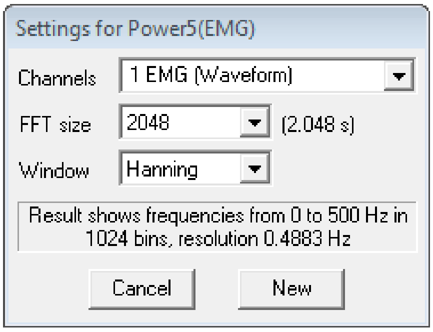
***
Note: Don’t be fooled by changes in the y axis scaling. For example, sometimes when a muscle is relaxed, Spike2 will autoscale the y axis so it looks as though your signal amplitude is huge. To correct this, ask the subject to contract the muscle and then rescale the y axis using Ctrl + Q (or End). Always remember to check the **absolute** values in your EMG channel(s) before concluding that something is wrong.
***

Correcting possible problems. Unfortunately there are many factors that can interfere with EMG signals. Some of the more common ones are outlined here.

- Bad connections. If your signal looks bad, either constantly or just occasionally, check all connections to ensure that all cables are properly connected. Occasionally you will find a faulty cable- if you suspect this is the case, try changing cables one at a time to isolate the problem.
- Movement artefacts. Where possible, tape the cables to the subject’s skin to avoid large cable movements, especially if you are testing during dynamic tasks.
- Batteries. Firstly, as obvious as it sounds, make sure that all systems are switched on! Also check that batteries still work. These basic issues are very often the cause of bad/no signals.
- DC offset. A raw EMG signal should have a mean value of zero. Occasionally this will not be the case because of a shift in the baseline. This is not necessarily a problem since it can be corrected during post-processing, but it is better to address it as early as possible. To do this

in Spike2, right click anywhere in the channel that needs correcting, then choose *Channel Process*. Select the *DC Remove* option and *Add*.
- Lights and other mechanical equipment. Occasionally lights will interfere with signal quality. Try switching off the lights closest to the measurement site. Other possible sources of interference include vibration from other test devices or nearby machinery. Unfortunately, in some cases you will not be able to do anything about the interference, but where possible, try switching off and unplugging anything that is not needed in the measurements but you suspect contributes to the interference.

Basic EMG analysis. One of the first things you might want to do with an EMG signal is check the frequency content. We know that the majority of signal power should be within the range of about 20-300 Hz, so checking the power spectrum gives a good idea of signal validity. To examine the power spectrum, select *Analysis → New Result View → Power Spectrum* from the dropdown menu. In the *Settings* window, choose the channel you want to process and the Fast Fourier Transform (FFT) size. The larger the FFT size window, the higher is the resolution in the resulting power spectrum. In general, 2048 will be sufficient. Click the *New* button and then select the time window during which you want to compute the FFT, followed by *Process*.



Settings for Power5(EMG)

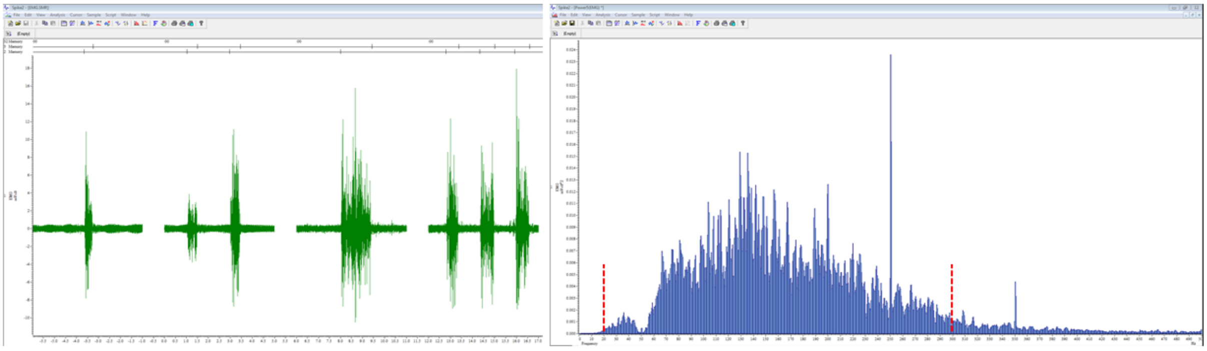
Channels: 1 EMG (Waveform)

FFT size: 2048 (2.048 s)

Window: Hanning

Result shows frequencies from 0 to 500 Hz in 1024 bins, resolution 0.4883 Hz

Below is an example of an EMG signal and an FFT computed from this signal.



As expected, most of the signal power is between 20-300 Hz (indicated by red vertical lines). If there were movement artefacts in the signal or a DC offset, we would expect to see a larger signal at 0-10 Hz. You can save the result of this analysis using *File → Save As*.

Full Wave Rectification. Rectification is the process of removing negative values from a signal. To do this in Spike2, right click anywhere in the channel, select *Channel Process* and then *Rectify* and *Add* (or choose *Analysis → Channel Process* from the dropdown menu).

Smoothing. To produce an EMG envelope, after rectifying the signal you could then add the smoothing option (in the same way as you did to rectify). For the smoothing option, Spike2 simply applies a pre-defined filter to the signal, but you can modify the time constant that is used after you have added the process (larger time values result in a smoother signal). Alternatively, you can use the filtering design tools that are built in to Spike2 to customise your filter. These are accessed using the dropdown menu (*Analysis → Digital Filters*). This allows you to customize the order, type and cutoff frequencies to use for filtering.

For all signal processing, it is a good idea to make a copy of your original channel before applying processing (e.g. by duplicating the channel). This allows you to easily see the effect of any filtering.

Practice

Choose a muscle to measure from and setup Spike2 so that you can record EMG data. Ask the subject to perform a maximal contraction (you may need 2-3 trials), and then contractions at different force levels (e.g. 25, 50 and 75% of MVC). Try performing some basic analyses on your data, such as computing RMS EMG from a certain time window of each contraction. Do your results make sense? (Hint: Remember to check signal quality **before** starting data collection).

Measuring forces

Force data are one of the core components of biomechanics, and give valuable information about the loading of different regions and tissues of the body. Arguably the two most common scenarios where we might want to measure forces in our labs are jumps and locomotion, both of which are presented below.

Measuring forces during jump tests

The most accurate way to measure 3D forces is to use a force platform like the one shown here. Analog data can be sampled in 3 dimensions and recorded using Spike2. This will require 3 BNC cables from the force plate to the A/D board. In some cases, you may only be interested in the vertical force, in which case only 1 cable is needed. When setting your configuration, you will need to know which analog output from the force plate corresponds to which force direction. In general the 3 force components are labelled X, Y and Z, which refer to vertical, medio-lateral and antero-posterior forces. However, different conventions are used to denote which is which. To test this, first stand still on the force plate. The vertical force should be equivalent to your body mass, while the other 2 components should be around 0. To determine the other 2 components, apply a force in the front-back direction, and see which channel shows an increase in signal. Then do the same for the side to side direction. Of course, the channels you refer to as antero-posterior and medio-lateral will depend on which way the force plate is facing relative to the person standing on it.

Another way to do jump tests is to use a jump mat like the one shown here. This is a much more basic version of a force plate. In fact, this one, which is custom-made by our technicians, doesn't quantify force as such. Instead it simply registers contact and take-off times using a force-sensitive resistor. Using the attached stopwatch, you can record contact and flight times separately. This information can then be used to estimate jump height, for example by using the so-called 'Flight time' method, which is very simple. To do this, you will need to solve the following 2 equations (taken from 'Analysis of standing vertical jumps using a force platform' by Nick Linthorne):

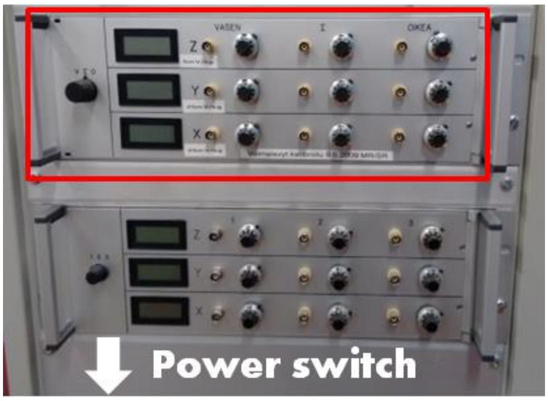
① $v_{\text{to}}=\frac{g\,t_{\text{flight}}}{2},$ ② $y_{\text{flight}}=y_{\text{peak}}-y_{\text{to}}=\boxed{\frac{v_{\text{to}}^{2}}{2g}},$

where v_{to} is estimated vertical takeoff velocity, g is gravity (9.81), t_{flight} is flight time in seconds, y_{flight} is jump height in metres. As a hypothetical example, if my flight time is 0.534s, then v_{to} would be (9.81 * 0.534) / 2 = 2.62, so y_{flight} would be (2.62*2.62) / (2 * 9.81) = 0.35, i.e. 35 cm. Of course this is not as accurate as directly measuring jump height, but it is probably accurate enough for most coaches or for field tests where a force plate may not be available or practical. The interested reader is advised to look at Dr. Nick Linthorne's excellent article on this topic which is available for free here: https://www.brunel.ac.uk/~spstnpl/Publications/VerticalJump(Linthorne).pdf

Measuring forces during locomotion

In Jyväskylä we have the unique ability to measure forces from each limb individually over a 10m instrumented walkway (it is actually slightly shorter due to the configuration of the force plates). This walkway is effectively 2 rows, each consisting of several force plates laid end to end. It is possible to record force signals from any of the individual plates, but for locomotion purposes, it is more practical to measure the net force from each row separately (or the net force of all plates for running where only 1 limb touches the ground at any given time).

To sample the force data, your BNC cables need to be connected to the control box on the front of the big grey unit shown below. Note that it is possible to sample data from either side separately (‘Vasen’ and ‘Oikea’, left and right) or from both sides combined (‘Σ’). So if you want to sample forces in 3 directions and from both legs separately, you will need 6 BNC cables.



Remember to check which directions the X, Y and Z correspond to. Note also that left and right depend on which direction the subject will be moving in! So if you intend to collect data in both directions, then for trials in one direction, the side corresponding to left will correspond to the right side when the subject moves in the other direction. For this reason, it is generally easier to only sample data in one direction.

***

Tip: Although not essential, it is strongly recommended that if you intend to use the walkway, you switch it on at least one hour before your data collection will begin. Otherwise you will experience a lot of drift in the baseline level of the force signals, which makes analysis slightly more complicated later on.

***

Practice

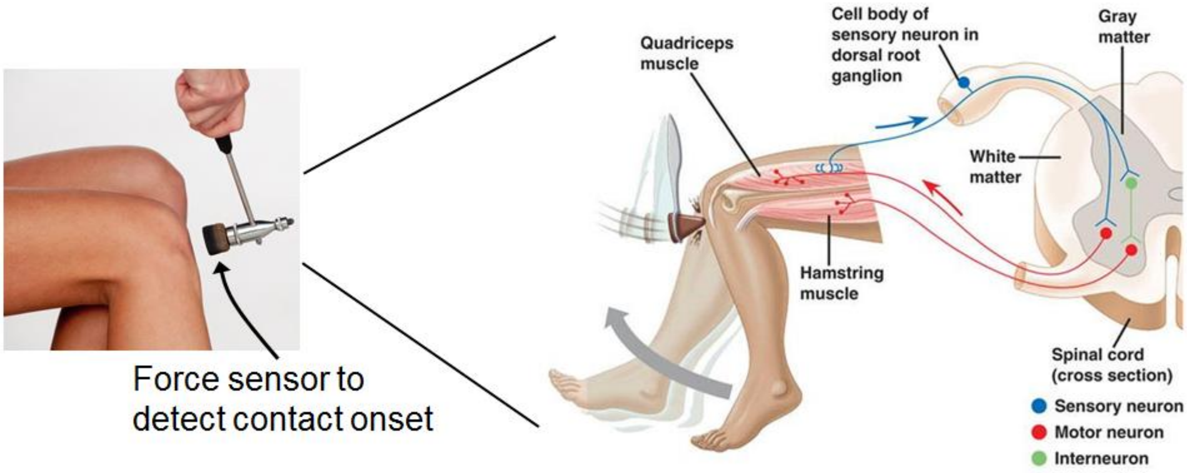
1. Go to this link: https://walkingwithrichard.files.wordpress.com/2016/02/modifying-the-ground-reaction-2016.pdf (or ask me for the PDF if the link doesn’t work). Try to complete the different force challenges, and save your data in Spike2 to show whether or not you succeed. Try to think about what you needed to do in order to match each force profile, as this could help you to understand the different clinical conditions that they are associated with.
2. Using a jump mat, a metronome (Hint: download a free app) and the equations shown above, ask a subject to jump at different frequencies (controlled with the metronome) and examine how jump height changes with frequency. Are your results logical? Why/why not?

Measuring reflexes

There are 2 broad ways of testing reflex function: stretch reflexes, which are elicited using a mechanical stimulus such as a fast joint rotation, and H-reflexes, which are evoked using electrical stimulation. In the literature you will often hear the H-reflex referred to as the 'electrical equivalent' of the stretch reflex. It is important to realise that this is not quite true, because the nature of the stimulus is very different, and the way that the stimulus is processed by the CNS is also different (see for example: http://jn.physiology.org/content/80/2/610). However, it is true that both tests assess similar (but not identical) pathways, and are often referred to as monosynaptic reflexes because an afferent impulse travels to the CNS, crosses a single synapse and the efferent response then travels to the muscle. In these tests we use surface EMG to measure the reflex responses. Below is a description of how to perform these tests for the soleus muscle, but they can also be adapted to study other muscles such as the quadriceps.

Measuring stretch reflexes

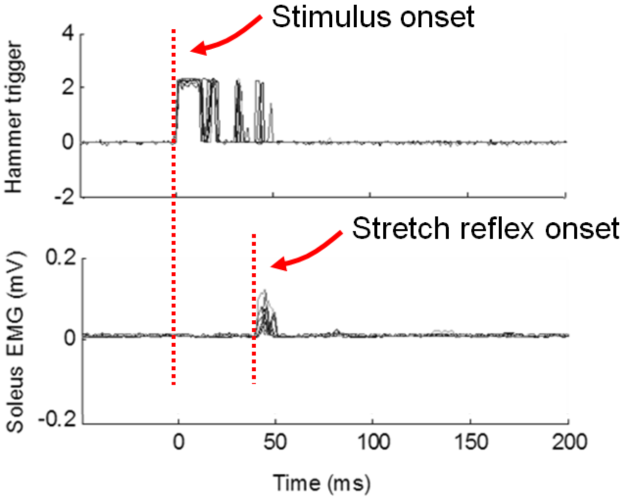
Stretch reflexes can be evoked in at least 3 ways. One is through a natural 'perturbation' such as when you accidentally trip whilst walking, but this is obviously very difficult to measure. A second is to use a dynamometer to impose a fast joint stretch. Although we have this possibility in our labs, for demonstration purposes, the easiest way to perform this test is to use a 3rd method- a tendon tap- to evoke the response.


Force sensor to detect contact onset
Quadriceps muscle
Cell body of sensory neuron in dorsal root ganglion
Gray matter
White matter
Hamstring muscle
Spinal cord (cross section)
● Sensory neuron
● Motor neuron
● Interneuron

The tendon 'hammer' shown in the picture contains a force-sensitive resistor so that when it contacts the skin, a trigger pulse is emitted, helping us to identify the precise onset of the stimulus (the tendon hammer from our lab is shown to the right and this also has a force-sensitive resistor in it). By also measuring EMG, we can then look at the latency and amplitude of the resulting reflex response. To set up the experiment, you will need to add an EMG channel and the analog force signal from the tendon hammer to your configuration.

Reflex latency is remarkably consistent between trials (variation around 1-2ms between repetitions) but reflex amplitude can vary hugely. It is therefore important to perform several trials rather than just 1 or 2 if you want to get a more reliable estimate of reflex amplitude.

The data shown in the figure to the right are taken from a passive muscle. Stretch reflexes can be measured in active muscle, but because of the background EMG activity, it is much more difficult to identify the latency of the response.



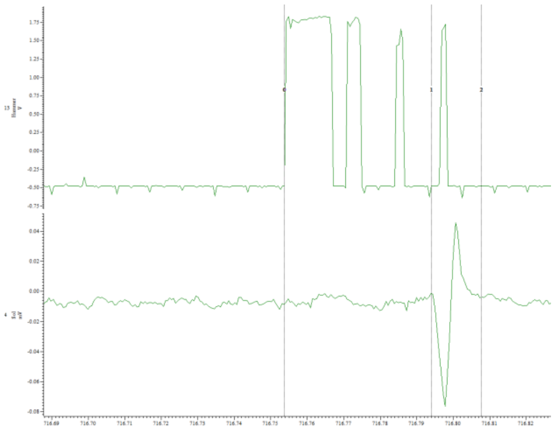
It is very important that the hammer is lined up with the tendon correctly. If not, you might not evoke a stretch reflex response. To check this, first palpate for the tendon, and try to get the hammer to hit approximately in this region. For the Achilles tendon, approximately 3-4cm above the calcaneus usually works well, and for the patella tendon, approximately 1cm below the lowest edge of the patella. To check that it works, try evoking 2-3 responses (with an interval of ~10s in between), and examine the EMG trace to see if there is a clear burst like response.

***

Note: Your repetitions should be as consistent as possible, so you should try to ensure that the subject's position is consistent, and that you drop the tendon hammer from the same height and in the same way each time.

***

To analyse stretch reflex amplitude and latency in Spike2 (see figure to the right), you can use cursors to manually identify the onsets of the stimulus and response (to get latency time), and then set a 3rd cursor approximately 20ms after the reflex onset, and measure peak-to-peak or area within that region. You could also use the tendon hammer trigger to produce an event pulse, which could then be used to set active cursors (for instructions, see 'Basics of Data Analysis in Spike2'). This would be useful if you are measuring latency/amplitude from a large number of trials.

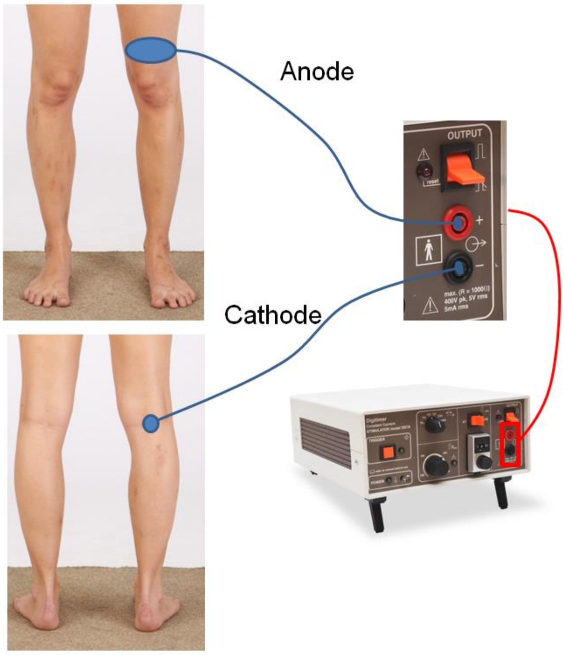


Measuring H-reflexes

This test requires a more complex setup than the stretch reflex test, so let's review this first. Here I describe the more commonly used Digitimer stimulator (shown in the photo). We also have an excellent Biopac STMISOL, but this must be triggered via Spike2 and operates in a different way, so I won't cover that here.

Although you can trigger the Digitimer stimulator manually, I would advise you to trigger it via Spike2, partly because it is more convenient for you, and partly because it avoids the accidental double pulses that tend to occur when leaving your finger on the stimulator button a fraction to long! A schematic of the configuration setup for this test is shown in the ‘Basics of Data Analysis in Spike2’ section.

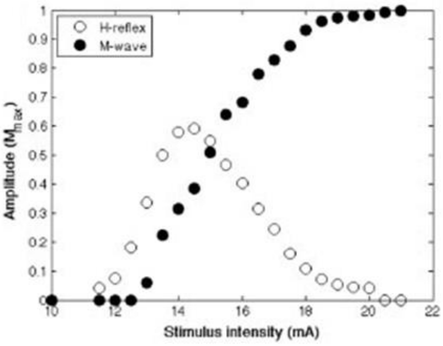
After finalizing your configuration and placing EMG electrodes over the muscle of interest, you are ready to start setting up the stimulating electrodes. Firstly, a quick recap of electricity: it is the flow of electrons around a circuit. Therefore, you need to have a complete circuit for your stimulation to work. To achieve this, you apply a cathode to the region you want to stimulate. Electrons will flow from this point to your anode, which will be positioned just above the patella in this case. In this setup, a complete circuit is formed from one electrode to the other via the body. Therefore, applying an electrical stimulus results in current flowing from the cathode through the leg towards the anode.



The precise position of the anode is less crucial, but the cathode position has to be exactly right or the stimulus will not activate the target nerve properly. To determine the correct cathode position, we usually use a movable electrode such as the one shown here. This allows us to easily try different electrode positions and check the effect this has on the EMG response. This can then be replaced with a permanent electrode once we are satisfied that we are stimulating the target nerve.

***
Tip: When looking for the optimal electrode position, it is a good idea to use a low, constant current and try different positions. This allows you to compare the effect of a change in position only on the EMG response. If you change electrode position and current intensity, then you will not know which of these factors was responsible for the change in EMG response.
***

Once the electrodes are attached and the cables are connected, you can begin measuring H-reflex / M-wave responses. Aim to increase the current intensity in small, regular intervals. If the intervals between pulses are too large, you may miss valuable information about the H-reflex / M-wave recruitment curve (see figure to the right). You should use a minimum of 10-12s between consecutive pulses to avoid the effects of post-activation depression that can reduce H-reflex amplitude.



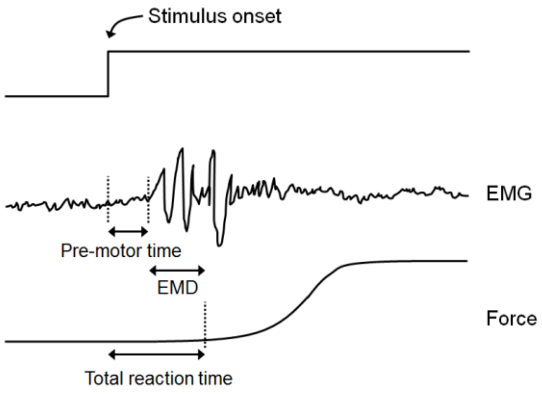
It is very important that when recording the recruitment curve, the subject maintains a constant position. Even small movements of the head or limbs can have a big effect on reflex responses, as can attention, open versus closed eyes etc. It is therefore important to explain to your subject before testing that staying still is vital.

Practice

Using either the H-reflex or stretch reflex method, record a series of reflex responses (at least 5), then perform some kind of intervention and repeat the reflex measurements. How does the intervention affect reflex latency? And amplitude? How might you explain your results? (Hint: examples of interventions could be a fatigue test, a warmup etc.).

Reactions

In the previous demo we looked at reflexes, which are automatic responses that we do not voluntarily control. Reflexes occur in a very short time frame, usually tens of milliseconds, and are often useful for rapid corrections, e.g. in response to an unexpected trip. Conversely, reactions are under voluntary control, and therefore occur over a longer time frame. For example, when the starter's gun goes off, a sprinter will take approximately 100ms to react to the sound of the gun and push off from the starting blocks.



In this demo you will learn how to test the reactions to sensory stimuli that test different pathways: touch, vision and auditory stimuli. A schematic of what will be measured is shown to the right. The stimulus could be a sound, light or touch signal, which the subject will react to. As you can see, the total reaction time is made up of two components, the pre-motor time (sometimes referred to as the EMG onset) and the electro-mechanical delay. The easy way to remember the second one is that it is the delay between the first electrical activity (i.e. EMG onset) and the mechanical response (i.e. force onset).

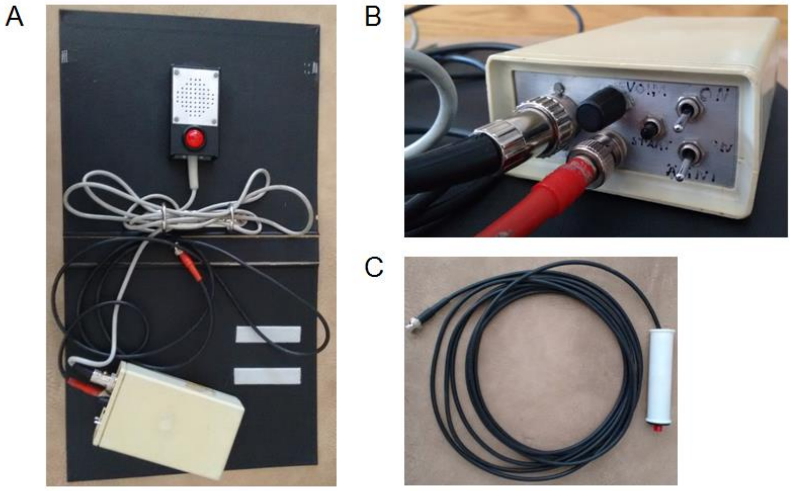
As you may have guessed from this figure, as well as a device to evoke the different stimuli, you will also need to sample EMG and force from the muscle(s) that will be used in the reaction tests. Suggested target muscles are the biceps and the quadriceps, since it is easy to measure force and EMG from these muscles in our demo lab.

***

Note: As you can see in the example above, it is not always easy to determine the precise onset of a force or EMG signal. If you analyse the data manually, the same person should analyse all data to avoid errors due to interpretation between different people. Another method is to use a threshold algorithm to detect onsets. This method can be more objective, but may not work in all cases due to the variability of biological signals.

***

To evoke the stimuli for these tests, you will use the custom-made kit shown below, which allows us to test all three of the target pathways using simple reaction tests. In parts A and B of the figure, you can see the equipment used for the sound and vision tests. The BNC cable should be connected to the A/D board so that you can sample the stimulus on/off signal and thus know exactly when the stimulus occurred. The white box in B is used to produce the stimulus by pressing the *START* button. If the *ÄÄNI* switch is down, the



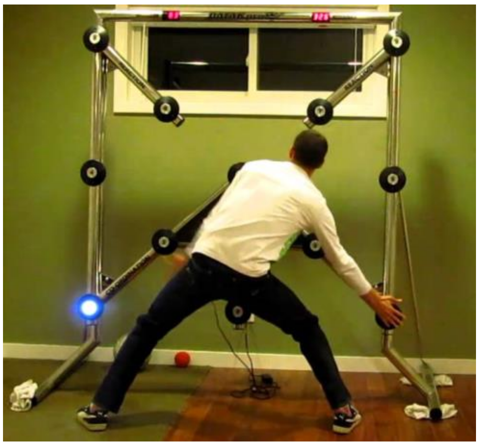
stimulus produced is light only. If it is up, both light and sound will be produced. Turning the *VOIM* knob will alter the tone and volume of the sound. Note that when you are testing vision, the *ÄÄNI* switch should be turned off so that the subject only reacts to the light stimulus (and the light obviously needs to be within the subject's field of view). Similarly, when testing the reaction to sound, the light box should be kept out of the subject's view to ensure that they only react to the sound stimulus.

To test the reaction to touch, remove the BNC cable from the unit shown in A of the figure above from the A/D board, and instead plug in the BNC cable shown in C. On the other end of this cable is a red button, which is essentially just a force-sensitive resistor. Every time this button is pressed, an on/off trigger pulse is sent to the A/D board.

For all 3 tests, it is important to give clear instructions to the subject. Since you are testing their reaction times, they must react to the stimulus as quickly as possible. You should also record several trials because many subjects show learning effects with this task, whereby reaction time is slow at first but gradually speeds up.

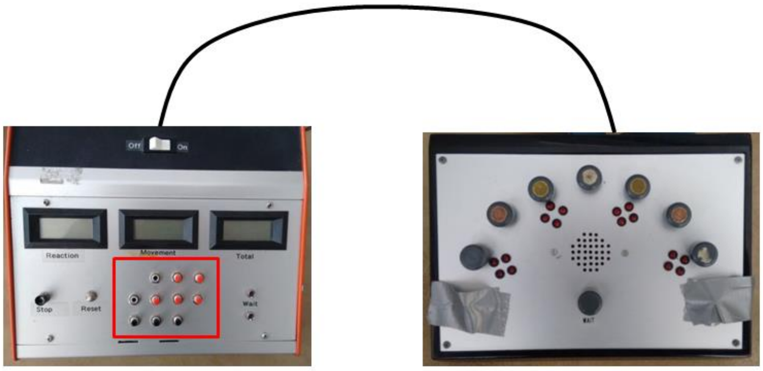
<u>Complex versus simple reactions</u>

In the examples presented above, we only considered what are known as simple reactions. These involve one stimulus to which a subject has to react. Therefore, the reaction time is usually quite short because the subject already knows which stimulus they need to react to, and the only unknown factor is precisely when it will start. However, in real life, we are often faced with more complex stimuli, as shown in the picture below.



In this example, a light will come on and the subject will need to press it, but the subject does not know in advance which light it will be. Thus, the reaction process becomes more complicated because the subject will first need to determine which stimulus to react to, and then actually press the appropriate button. As a result, reaction time to complex stimuli is usually longer than when using simple stimuli.

In our lab we have our own device than can be used to examine complex reactions (see below). This device does not allow you to output the results, but they are displayed on the screen. The tester can press any of the red or black buttons and after a random delay, the corresponding light(s) will light up. The subject will then press the corresponding button to switch off that light, and the device will record the reaction time.



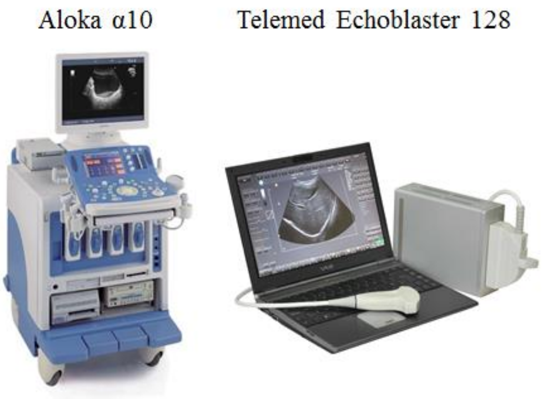
To make sure it is always a fair test, there is a *wait* light shown on the tester's panel and a *wait* button in front of the subject. The subject should keep their finger pressed on the *wait* button to avoid them being able to anticipate which light(s) might come on and moving their hand closer to the target button. The *wait* button on the tester's panel will light up when the subject is pressing the *wait* button.

Practice

1. Try examining simple reaction times using an upper limb muscle and a lower limb, and then compare the results. What do you hypothesise will happen? Which muscle will show the faster reaction time? And why? Does it matter which pathway you test (sight, sound, touch)?
2. Design and carry out your own complex reaction time test. Examples of what you might test include a gender comparison, left hand versus right hand, effects of different light conditions etc.

Ultrasound

Ultrasound is a noninvasive method that is used in a wide range of applications, including sonar, fetal scanning, and more recently, examinations of muscle and tendon tissues. In Jyväskylä we have several ultrasound systems, but the most commonly used are probably the Aloka α10 and the Telemed device. Since these devices are quite different, they are each dealt with separately.



Aloka α10

The Aloka is the bigger of the two devices, and in terms of image quality, it is far superior to the Telemed system. However, because of its size and weight, it is not suitable for dynamic measurements. Below is a summary of the main applications that you might use with this device, and how to perform the measurements on screen.

***

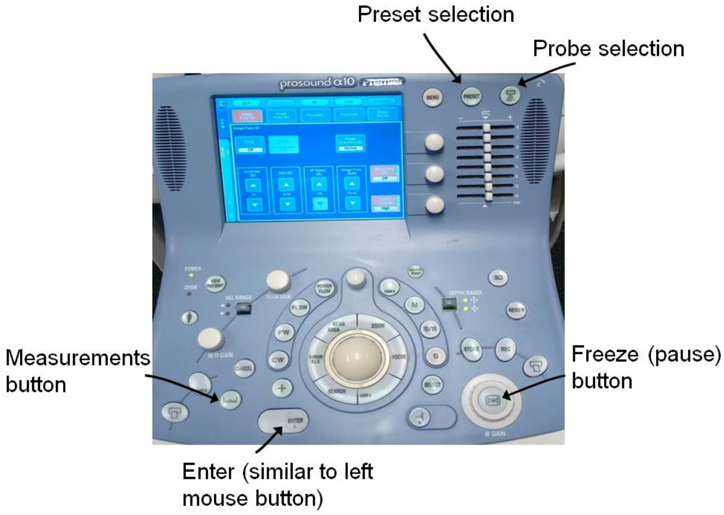
Note: When scanning with ultrasound, you will almost always need to use gel to help transmit the sound waves from the probe to the skin. It is important that you clean the gel from the probe properly after use. The probes are extremely expensive and the scanning surface is very delicate, so only use paper towels for this, but ensure that all gel is removed.

***

The power switch for the device is on the left as you look at it, just below and to the left of the blue control panel. After switching on, wait a few minutes until the system is ready (once the touchscreen controls are visible, it is ready).

Measuring thickness / pennation angle / muscle length

The first thing to do is to select the probe you want to use. Press the probe select button (see picture to the right) and a list of the probes that are attached will be displayed. If you are unsure which probe is selected, press on the probe surface with your thumb and look to see whether this is evident in the image.



You will then need to scan the structure of interest. If you plan to take static measures on screen (e.g. thickness), the easiest way is to find a good image, then press the freeze button. This means that you don't need to keep the probe in the same position for a long time.

Once the image is frozen, press the measurements button followed by the + button. A crosshair will appear on the screen allowing you to position your first measurement point. The default measurement type is length, so you will notice that after placing the first point, when you move the cursor, the length of the 'line' will constantly be updated. You can also easily measure angles, area etc, by using the touchscreen options to change the measurement type.

***

Note: In order to save images or videos, you will first need to add a new patient by pressing the *New Patient* button. You do not need to fill in all of the fields but it is good practice to at least give a patient name that you will remember to make finding your data easier later on. You will find the keyboard hidden under the blue interface at the front of the machine; push it in and it will then spring out.

***

Saving single images

In order to analyse images offline or use the images somehow, you will first need to save them and then export them. To save a single image, once you have pressed the freeze button, press *Store* (just above the freeze button), and the image will be saved. Data are always saved with the associated date of the measurement, so as long as you have the date recorded, you will always be able to find old data.

Exporting single images

To export data from this machine, you need to insert a memory stick into it **before** switching it on. If you forget to insert a stick before data collection, this is not a problem, but it means that after you have collected your data, you will need to switch off the machine, insert the stick (and wait a few minutes until the device is ready to reboot), then switch the machine back on.

To export single images, press the *freeze* button and then the *review* button. In the window that appears, you can see all of the files that have been saved in this session. select the image(s) that you want to export by clicking on them, then select *export* followed by *MO*. As mentioned above, this will only work if the device recognises your memory stick. On the memory stick, the data will be saved in a folder titled 'DICOM' or 'DICOMIMG'.

Saving videos

To save a video, you first need to check that the save settings are correct. On the digital display that is part of the blue control panel, find and press the *Store setup* button (on tab 'B 1/3'). Make sure that the *store media* option is set to *HDD* so that files are saved to the Aloka's hard disk. The *time cycle* option determines how many seconds of data are saved each time you record. The options are 1, 2, 4, 8 and 16 seconds (obviously the longer the trial, the larger the file size will be). To change the recording time, turn the white dial that is to the right of the *time cycle* option.

***

Note: Any settings that you change concerning video recording will not be saved if you select a different preset or switch the device off. You should therefore always check these settings before you start recording trials.
***

The final setting to check is the *acquire mode*. You will most likely want to use *pre time* or *post time*. For *pre time*, when the *store* button is pressed, the device will save data from X seconds before you press the *store* button until when the *store* button is pressed (where X is the number of seconds you have chosen to record for). On the contrary, with *post time*, the recording will start when you press *store*, and continue for X seconds after that. This means that you can record an event by pressing *store* shortly after the event you're recording has actually happened (using *pre time*) or you can start recording just before the event happens (using *post time*).

Once you have checked all of the necessary settings, to record a video you simply make sure that the image is not paused and then press *store*. You can review any saved videos by pressing *freeze* and then *review* on the blue panel.

Exporting videos

This process is essentially the same as exporting single images, except that it produces larger files and may take longer to export.

Triggering and synchronisation

To automatically trigger the ultrasound to record, you will need to create a digital output pulse (square pulse, 0.1s is fine). This will act as the 'record' command to the ultrasound machine (see 'the output sequencer' section above for instructions to set this up). The digital output should be connected to the *footswitch* port on the Aloka. Before data collection, remember to set the video record settings as described above.

***
Note: The *footswitch* and *pulse* ports do not use standard BNC cables, but we have 2-3 of the required cable type in the lab. These should be located somewhere on the Aloka device or in the new biomechanics lab.
***

To make your analysis much easier, you will need to add a visible pulse to the ultrasound screen. This signal will show within the ultrasound image the time point when data recording started, which allows you to synchronise ultrasound with other data sources during your analysis. To set this up, use a DAC output that is connected to the *pulse* slot on the Aloka. The DAC output can be a simple square wave pulse or whatever you want to use to indicate the start/end of a recording.

Another step that you could take to make data synchronisation as easy as possible is to split the DAC output signal so that one cable goes to the Aloka as stated in the previous paragraph, and another goes to the A/D board as an *Event* channel (remember to define this in your configuration).

Telemed device

As mentioned earlier, the Telemed device offers poorer image quality than the Aloka, but it is a good choice if you require a more portable option, and/or if sampling rate is not an important factor; the Telemed offers a maximum video recording frequency of 80Hz (i.e. it records 80 pictures per second), whereas the Aloka can easily sample at over 100Hz with good image quality.

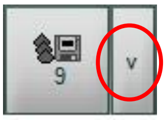
To set up the Telemed device for data collection, follow this set of steps:

1. If using the laptop that is mainly reserved for the Telemed device, login to the laptop using the username and password shown on the post-it note (**username:** .\localadmin **password:** Fq_1vzUvye6b).
2. Make sure that the power switch at the back of the silver ultrasound unit is on, and that the usb cable from the Telemed device is connected to the laptop. If you are recording video and need to synchronise it with other data sources (e.g. force, EMG in Spike), plug the output pulse cable from the silver ultrasound unit into an input channel in the A/D board, and remember to add this channel to your configuration so that Spike knows what it is.
3. Run the Echo Wave II software (shortcut on desktop, or downloadable from here for free if using a different laptop: http://www.telemedultrasound.com/download/software-downloads/?lang=en)
4. Before recording anything, set the save location by choosing Menu → Tools → Options → Click the 'Saving and printing' tab → In the 'Save files to folder' area (top left) click 'Browse' and choose the folder to save to (use C:\Echo Images and then add your own folder) and click OK.

Now you are ready to start imaging something.

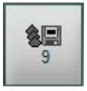
<u>Recording videos</u>

Before recording, click the down arrow just to the right of the save button at the bottom of the screen:



Make sure the 'Cine Quick Save' option is selected. If you have defined a saving folder as shown in step 4 above, all videos will be saved to this folder.

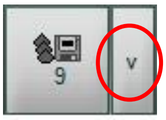
- To record a video, first pause the image (space bar). This is necessary because this device always starts recording once you unpause it.
- Where relevant, start Signal/Spike/whatever software you are using, then press space bar on the ultrasound laptop again and this will start the ultrasound recording and send an 'on' pulse to the A/D board so you know exactly when ultrasound started recording and can use this pulse to later synchronise ultrasound data with other data sources.
- When the movement/contraction that you are imaging has finished, press space bar again to pause the image and stop recording with ultrasound.
- To save the video, press number 9 or click on the 'Cine quick save' button at the bottom of the screen:



- Video filenames include the date and time that they were recorded, so make sure you have a list of the order in which your trials were recorded.

Recording single images

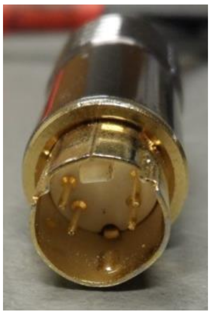
Before recording, click the down arrow just to the right of the save button at the bottom of the screen:

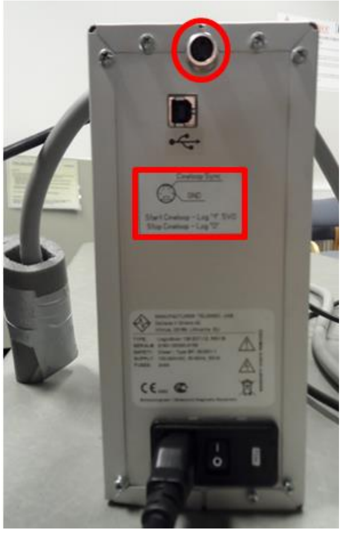


Make sure the ‘Image Quick Save’ option is selected. If you have defined a saving folder as shown in step 4 above, all images will be saved to this folder.

- To record an image, simply find the region of interest with the probe and press the quick save button (number 9). Or, if it is too difficult to keep the probe still and record at the same time, first pause the image (space bar) and then press the quick save button.
- Images are saved as Jpegs in the target folder
- Image filenames include the date and time that they were recorded, so make sure you have a list of the order in which your trials were recorded.

Synchronisation





The Telemed uses a different form of synchronisation to the Aloka. In this case, the Telemed simply sends a signal out to notify when it starts and stops recording. To get this to work, connect the cable shown to the left to the port on the back of the Telemed device (shown in a red circle in the figure to the right). This cable has a specialized connection at one end, but the other end is a BNC.

The BNC end of this cable should be connected to the A/D board as a waveform input channel. When you want to record ultrasound data, first pause the image. This sets the trigger signal to 0V. When you want to start recording, press pause again (i.e. unpause the image) and in the input channel that you just created, you should notice that the signal jumps from 0 to 5V, indicating that the Telemed is recording. To stop recording, simply pause the image again and you should see the synchronization signal go from 5V back to 0. Remember to save the ultrasound video. Otherwise, the next time you unpause the device, your video will be lost.

Convert video (.tvd) data to avi files

At present the system is set to record video files in the .tvd format, which allows the files to be saved faster and has various other benefits compared to saving directly to .avi. You can change the settings so that videos are recorded directly as .avis, but assuming you have recorded .tvd files, here’s how to convert them:

- In Windows, click *start* and in the ‘Search programs and files’ box, type ‘cmd’ and then press enter.
- In the cmd window that appears, type the following commands in order (must be spelt correctly):

```
cd c:\      (then enter)
cd program files (x86)      (enter)
cd telemed      (enter)
cd Echo Wave II      (enter)
```

(alternatively, navigate to the Echo Wave II folder manually, then press *Control* and *Shift* at the same time and right click the mouse, then choose *Open Command Window Here*).

The next command starts the conversion. This must be exactly as typed below:

```
Echowave.exe -convert_directory “C:\Echo Images\Your folder name” tvd avi_comp
```

(where you replace the ‘Your folder name’ part with the exact name of the folder where your data are).

For each .tvd file in the folder you specified, the saved file will be opened in the Telemed software and then converted to a .avi file with the same name as the original.

Analysis of ultrasound data

Nowadays there are some excellent free options for ultrasound analysis. If you are working mainly with single images, the best option is probably ImageJ, which can be downloaded for free from here: https://imagej.nih.gov/ij/download.html

You can also find good instructions on how to use it from here: https://imagej.nih.gov/ij/docs/index.html

To analyse video data, the best software option will depend on the parameter(s) you are interested in. For measures of fascicle length, you can use the Ultratrack programme that is free to download: https://www.ncbi.nlm.nih.gov/pubmed/27040836 (links to the software can be found from this paper). Although various other algorithms have been proposed, unfortunately very few are available for free. I am happy to discuss with you about your needs if you still have questions about analysis.

More topics coming soon…