



opensignals
(r)evolution

biosignal acquisition tool-kit for high-level research applications

user manual



ATTENTION

Please read this manual before
using your PLUX product(s) and
this software

The information contained in this manual has been carefully checked and were made every effort to ensure its quality. PLUX reserves the right to make changes and improvements to this manual and products referenced at any time without notice.

The word Bluetooth and its logo are trademarks of Bluetooth SIG Inc. and any use of such marks is under license. Other trademarks are the property of their respective owner.

PLUX Wireless Biosignals S.A.

email: plux@pluxbiosignals.com

web: <https://www.pluxbiosignals.com/>

Headquarters

Zona Industrial das Corredouras, Lt. 14 – 1º

2630-369 Arruda dos Vinhos

Portugal

tel.: +351 263 978 572

fax: +351 263 978 902

Lisbon Office

Av. 5 de Outubro, nº 70 – 2º

1050-059 Lisboa

Portugal

tel.: +351 211 956 542

fax: +351 211 956 546

DISCLAIMER

OpenSignals (r)evolution, biosignalsplux & BITalino products are intended for use in life science education and research applications only; **they are not medical devices, nor medical software solutions, nor are they intended for medical diagnosis, cure, mitigation, treatment or prevention of disease and is provided to you “as is”.**

We expressly disclaim any liability whatsoever for any direct, indirect, consequential, incidental or special damages, including, without limitation, lost revenues, lost profits, losses resulting from business interruption or loss of data, regardless of the form of action or legal theory under which the liability may be asserted, even if advised of the possibility of such damages.

TABLE OF CONTENTS

DISCLAIMER	4
Introduction.....	9
1 Getting Started.....	10
1.1 Download & Install OpenSignals (r)evolution	10
1.1.1 Windows Installation	10
1.1.2 Mac OS X Installation.....	10
1.1.3 Linux Installation.....	11
1.2 OpenSignals (r)evolution Updates.....	12
1.3 Add-ons Installation (Activation)	12
1.4 Connecting Bluetooth Devices	15
<i>Note about BLE Connection – BLD112 (Bluegiga).....</i>	<i>16</i>
1.4.1 Windows 7	19
1.4.2 Windows 8	22
1.4.3 Windows 10	24
1.4.4 Linux.....	26
1.4.5 Mac OS X.....	33
1.5 OpenSignals Help	36
1.5.1 User Manual	36
1.5.2 Support	36
1.5.3 Access License Information	36
1.5.4 Activating & Deactivating Add-ons	36
2 OpenSignals (r)evolution	37
2.1 OpenSignals Main Screen	37
2.2 Device Manager	39
2.2.1 Connecting Devices.....	40
2.2.2 Connecting Devices Manually.....	41
2.2.3 Device States	42
2.2.4 Device Configuration	43
2.3 General Acquisition Settings	59
2.3.1 Continuous Mode	59
2.3.2 File Prefix	60
2.3.3 Devices.....	60
2.3.4 File Formats & Specifications.....	61
2.3.5 Converted Values.....	61
2.3.6 Save Locations	62
2.3.7 Auto-save Options	66
2.3.8 Online Scheduling	67
2.3.9 Replicate Device Configuration	68
2.3.10 Integration & Data Streaming – TCP/IP & Lab Streaming Layer.....	69
2.4 Save Last Recording.....	71
3 Recording & Visualizing Signals	73
3.1 Signal Acquisition Mode	73
3.1.1 Signal Acquisition.....	75
3.1.2 Information Bar	78
3.1.3 Control Bar.....	81

3.2	Signal Visualization Mode	85
3.2.1	Open OpenSignals Files	86
3.2.2	Signal Visualization	88
3.2.3	Information Bar	89
3.2.4	Control Bar.....	89
3.2.5	Visualization of Digital Input and Output Signals	91
3.2.6	Changing the Sensor Type of a Channel in Post-Acquisition	92
3.3	Data Logging	94
3.3.1	Scheduling Offline Acquisitions	95
3.3.2	Downloading Offline Acquired Data	99
3.3.3	Deleting Offline Acquired Data	101
3.4	Synchronization	102
3.4.1	Connecting Devices.....	102
3.4.2	Device Configurations in OpenSignals (r)evolution	103
3.5	Accessories	105
3.5.1	Trigger	105
3.5.2	Light-Emitting Diode (LED).....	108
3.5.3	Digital Synchronization Cable	110
3.5.4	Multi Sync Splitter	111
4	Add-Ons	112
4.1	Add-on Purchase & Activation	113
4.2	Listing & Using Add-ons	114
4.3	Add-on Reports	115
4.4	Add-On Processable Devices and Channels	117
4.5	Event Emulation	118
4.5.1	Hardware Configuration	118
4.5.2	Add-on Configuration	118
4.6	Event Annotation	122
4.6.1	Hardware Configuration	122
4.6.2	Add-on Configuration	122
4.7	Goniometer	126
4.7.1	Hardware Configuration	126
4.7.2	Add-on Configuration	127
4.8	Signal Statistics	129
4.8.1	Hardware Configuration	129
4.8.2	Add-on Configuration	129
4.9	Oximeter	131
4.9.1	Hardware Configuration	131
4.9.2	Add-on Configuration	132
4.10	Muscle Load	133
4.10.1	Hardware Configuration	133
4.10.2	Add-on Configuration	134
4.11	Electromyography (EMG) Analysis	137
4.11.1	Hardware Configuration	137
4.11.2	Add-on Configuration	138
4.12	Electrodermal Activity (EDA) Events	146
4.12.1	Hardware Configuration	146
4.12.2	Add-on Configuration	147
4.13	Heart Rate Variability (HRV) Analysis	149
4.13.1	Hardware Configuration	150
4.13.2	Add-on Configuration	151

4.14	Respiration Analysis (PZT & RIP)	156
4.14.1	Hardware Configuration	156
4.15	Add-on Configuration	157
4.16	Load Cell Force Analysis	159
4.16.1	Hardware Configuration	159
4.16.2	Add-on Configuration	161
4.17	Video Synchronization	163
4.17.1	Hardware Configuration	163
4.17.2	Add-on Configuration	164
4.18	Force Platform	166
4.18.1	Hardware Configuration	166
4.18.2	Add-on Configuration	168
4.19	fNIRS Analysis	171
4.19.1	Hardware Configuration	172
4.19.2	Add-on Configuration	172
4.20	Manual Measurement Input	174
4.20.1	Hardware Configuration	174
4.20.2	Add-on Configuration	175
5	Advanced Device Configuration	178
5.1	BITalino	179
5.2	BITalino (r)evolution BT (Bluetooth)	182
5.3	BITalino (r)evolution BLE (Bluetooth Low Energy)	185
5.4	biosignalsplux	187
5.5	biosignalsplux Hybrid-8	189
5.6	biosignalsplux Force	191
5.7	biosignalsplux Solo <small>[new designation for OpenBAN]</small>	193
5.8	MuscleBAN	195
5.9	MotionBAN	197
5.10	CardioBAN	199
5.11	RespiBAN Explorer	201
5.12	RespiBAN Researcher	203
5.13	RespiBAN Professional	205
5.14	Compatible Sensors & Channel Types	207
5.14.1	biosignalsplux	207
5.14.2	BITalino	211
5.15	3rd-party Sensors	214
5.15.1	I/O Breakout Cable	214
5.15.2	Sensor Breakout PCB	215
6	Software & APIs	217
6.1	OpenSignals (r)evolution	217
6.2	OpenSignals Mobile	217
6.3	APIs	217
7	Troubleshooting	218
7.1	Windows Bluetooth Stack Troubleshooting	218
7.1.1	Windows 7 & Windows 8	219
7.1.2	Windows 10	223
7.2	Device Firmware & Update	228
7.2.1	Checking Firmware Version (external tool)	228
7.2.2	Updating or Upgrading Firmware	229
7.3	Launching Errors in new Linux Distributions	231
7.4	Error List	232

- 8 Safety Information 234**
- 9 Regulatory & Legal Information 236**
 - 9.1 Disclaimer 236**
 - 9.2 Certification 236**
 - 9.3 Contact & Support..... 236**

Introduction

PLUX – Wireless Biosignals, S.A. creates innovative products for Physiotherapists and Researchers, by developing advanced biosignal monitoring platforms that integrate wearable body sensors, such as electromyography (EMG) and electrocardiography (ECG), combined with wireless connectivity and software applications to deliver real value recognized by its target markets.

OpenSignals (r)evolution is the software for data acquisition, visualization and processing while using PLUX's biosignal hardware platforms. The purpose of this manual is to introduce the reader on how to fully work with *OpenSignals (r)evolution*, exploring all its features and options. This version of the manual covers the *OpenSignals (r)evolution* version release for all *biosignalsplux* and *BITalino (r)evolution* products.

OpenSignals (r)evolution allows the user to acquire data from one or multiple devices, setup the desired acquisition parameters (such as sampling frequency and resolution), save standard ASCII text format or in the more efficient Hierarchical Data Format (HDF) for post-processing in third-party tools such as Python or MATLAB and visualize/zoom into the acquired data over a large number of channels.

It is extendable with add-ons that are either provided with the devices or can be purchased separately. It is even possible to create your own add-ons with specific functions and instruments (if you are interested, contact us and learn how to access the required tools) and share them and your projects with the *BITalino* and *biosignalsplux* communities.



<https://www.pluxbiosignals.com/>

1 Getting Started

OpenSignals (r)evolution is a Python-powered web-based application.

The *OpenSignals (r)evolution* software can be downloaded for free from the *biosignalsplux* and *BITalino* websites. New versions are released on a regular base, reason for which you are invited to check for updates at:

<https://www.pluxbiosignals.com/pages/opensignals>

After a successful download, follow the steps described in the next sections to install *OpenSignals (r)evolution* and to correctly set up the Bluetooth connection with your device.

1.1 Download & Install OpenSignals (r)evolution

1.1.1 Windows Installation



Double click the *OpenSignals (r)evolution* .EXE file that you've downloaded before and follow the steps of the installation wizard to install the software.

OpenSignals (r)evolution will appear on your desktop after a successful installation.

1.1.2 Mac OS X Installation



Double click the *OpenSignals (r)evolution* .PKG file that you've downloaded before and follow the steps of the installation program to install the software.

OpenSignals (r)evolution will appear in your applications folder after a successful installation.

1.1.3 Linux Installation



Open the *Terminal* and enter the directory of the downloaded *.deb* file using the *cd* command.

If you are not sure how to determine the correct path of the *downloaded .deb* file, enter the *sudo apt install* command into your terminal and drag'n'drop the downloaded *OpenSignals_revolution_ubuntu_amd64.deb* file into the terminal window. If done correctly, the current command, in your terminal, should be similar to this:

```
plux@ubuntu: ~$ sudo apt install  
./OpenSignals_revolution_ubuntu_amd64.deb
```

Press enter to continue.

Due to the *sudo* command you will be requested to enter your ubuntu account password. With this step it will be ensured that the *OpenSignals (r)evolution* installation is executed with administrator privileges.

You should now be able to find *OpenSignals (r)evolution* in the applications folder or run it through *Terminal* by simply typing *opensignals*.

1.2 OpenSignals (r)evolution Updates

Recent versions of the software will automatically notify you when newer versions are available for download. If you want to install the newest updates, click on *INSTALL* or click on *NOT NOW* to download the software at another time. Uncheck the *Remind me later* box if you do not wish to be notified anymore.

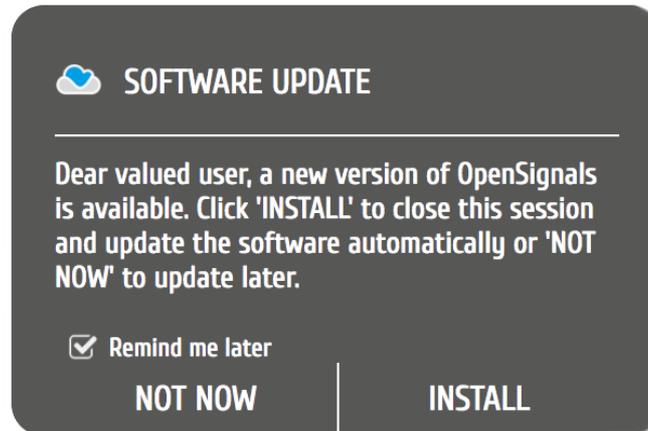


Figure 1: Software update notification.

1.3 Add-ons Installation (Activation)

OpenSignals (r)evolution has a range of optional software add-ons for real time and post-processing, which can be used to expand the capabilities of your *OpenSignals (r)evolution* installation. These add-ons are either included in your *biosignalsplux* kit or can be purchased separately to work with any of your PLUX device(s).

In both cases these add-ons are activated by PLUX and linked to the MAC-address of your device. However, a verification within your *OpenSignals (r)evolution* software must be done in order to use the add-ons.

The most recommended approach requires an internet connection for the activation process, as it is done remotely during the start of *OpenSignals (r)evolution*. You can use the software add-ons without a stable Internet connection after the activation.

For more information about *OpenSignals (r)evolution* add-ons check also Section 4 *Add-Ons*.

Automatic Add-On Activation

For this, simply ensure that your computer has an activate internet connection and connect the device to your computer and in *OpenSignals (r)evolution*. *OpenSignals (r)evolution* will automatically connect with the PLUX servers and activate all the add-ons linked to your device's MAC-address.

Manual Add-On Activation

In case the *Automatic Add-On Activation* process cannot be achieved for any reason, open the *OpenSignals (r)evolution* help interface (click the “?” icon) and refresh the add-ons list by clicking on one of the available refresh icons as highlighted in Figure 2.

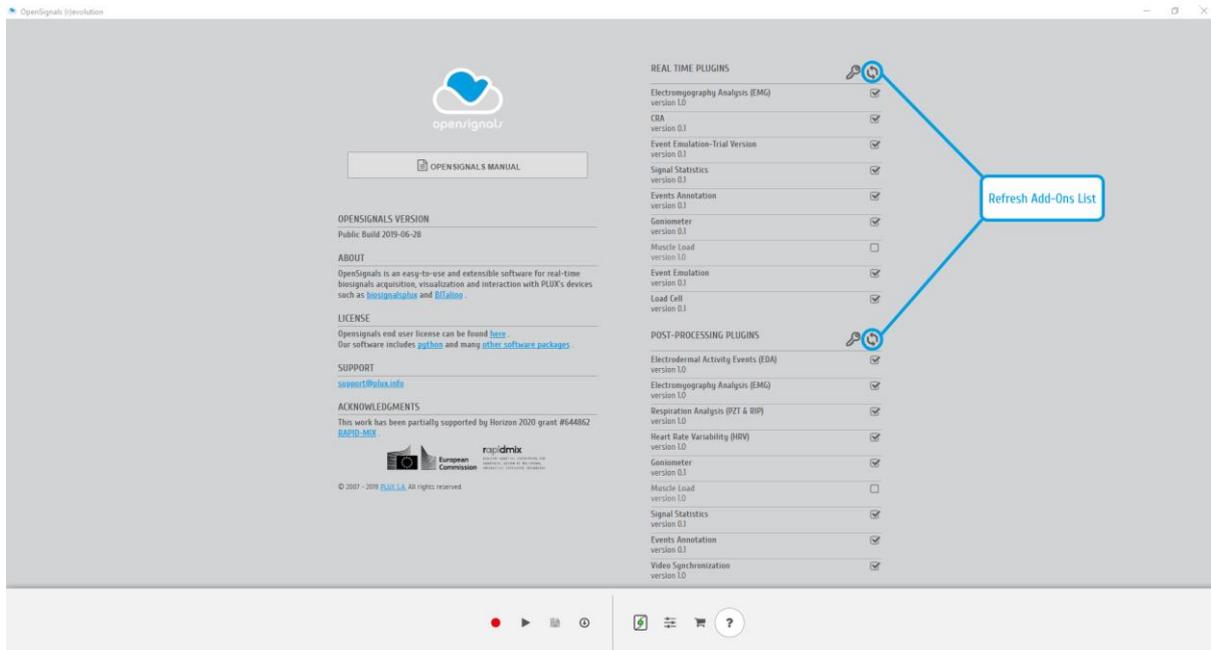


Figure 2: Manually refresh the add-ons list.

Offline Add-On Activation

However, if the computer where *OpenSignals (r)evolution* is installed does not have access to the internet, an offline Add-On activation approach is also available.

In these special cases, you should request an offline activation code for the purchased Add-On to PLUX Support Team (contact info at <https://support.pluxbiosignals.com/>), specifying on the e-mail the MAC-address of your device.

After receiving the offline activation code, it will be necessary to insert it on *OpenSignals (r)evolution*, which can be achieved through the previously mentioned help interface (Figure 3).

Once the help interface is opened, instead of clicking on the **refresh** icons you should click on one of the **key** icons (accordingly to the type of plugin you purchased | real-time or post-processing), as demonstrated on Figure 3.

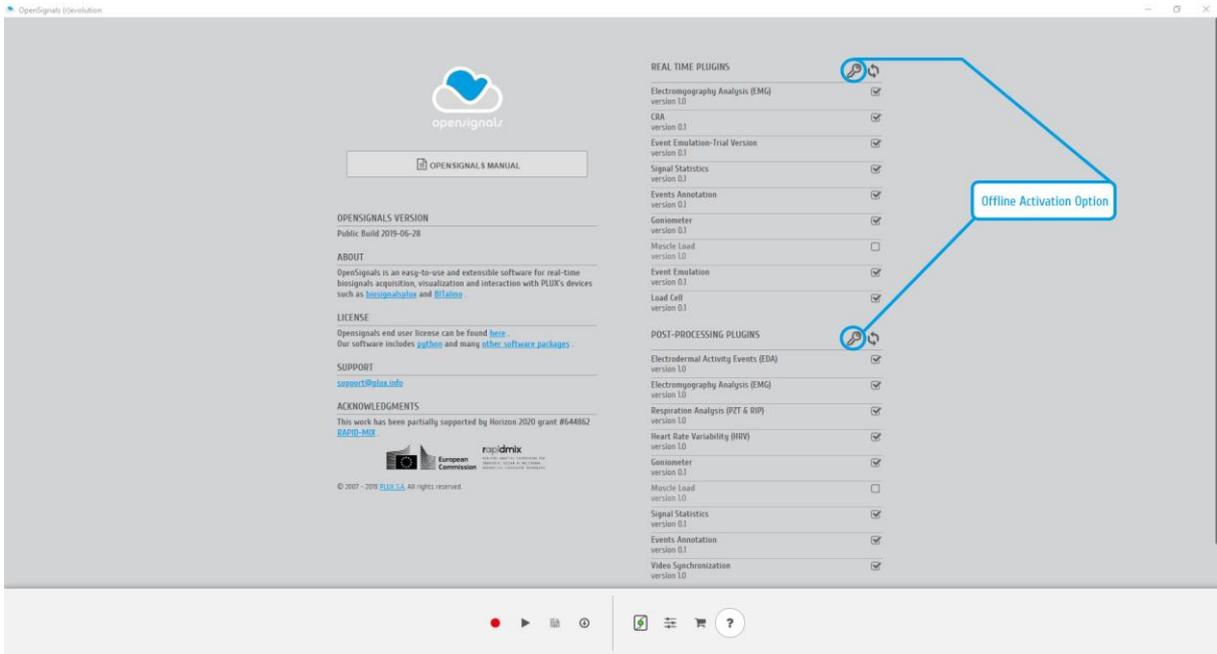


Figure 3: Buttons for accessing the offline activation screen

A new window will be prompted and there you can **a)** paste the activation code and **b)** click on “ACTIVATE” button (Figure 4).

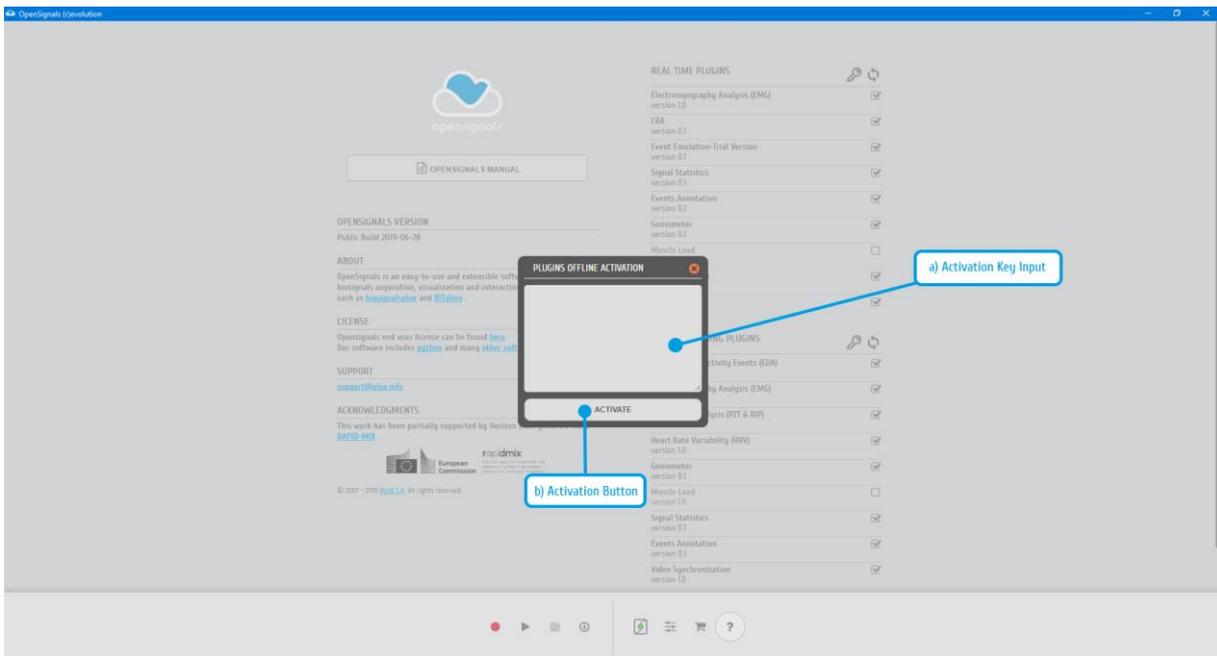


Figure 4: Offline activation inputs

The described steps should be enough to ensure an offline activation of the purchased Add-Ons. In order to finish the procedure, you only need to refresh the Add-Ons list, as described on the *Manual Add-On Activation* section available at Page 13.

1.4 Connecting Bluetooth Devices

NOTE

OpenSignals (r)evolution is able to establish a Bluetooth connection to your device whenever you add a PLUX device in the *OpenSignals (r)evolution* device manager. In this case, it is not needed to manually establish a Bluetooth connection with your device using your operating system's tools.

However, if you are experiencing any kind of connection issues, we recommend connecting your device with your operating system first (before adding your device in *OpenSignals (r)evolution's* device manager).

When connecting your device, you first need to enable Bluetooth on your computer or need to plug in the recommended Bluetooth dongle to an available USB port, if Bluetooth is not built in your computer or not reliable for high data rate transmissions.

biosignalsplux users may receive this dongle as part of their kits (not included in single purchases of single biosignalsplux hub).

BITalino users can purchase a trialled & tested Bluetooth dongle known to work with *BITalino*, at the [BITalino web store](#).

The following links contain recommended options accordingly with the device and Bluetooth technology:

biosignalsplux Bluetooth Dongle

<https://www.pluxbiosignals.com/products/bluetooth-dongle?variant=41293950648511>

BITalino Bluetooth Low Energy (BLE) Dongle (for BITalino BLE devices)

<https://www.pluxbiosignals.com/products/bluetooth-dongle?variant=41293957169343>

We recommend to follow the instructions presented on the following pages, accordingly to the operating system under use, ensuring a connection between PLUX's devices and your computer.

Note about BLE Connection – BLD112 (Bluegiga)

In order to ensure an adequate data transmission between Windows computers and PLUX BLE based devices a high-performance Bluetooth Enumerator is necessary, taking into consideration the communication rates supported by PLUX devices (available sampling rates up to 4000 Hz).

The internal Bluetooth of some computers may not support these sampling rates, so, in these specific situations we recommend the usage of a properly tested dongle from Bluegiga¹.

In normal circumstances the connection established between the computer and the dongle is enough to trigger the installation of the respective drivers, however, when this automatic procedure does not take place, the following steps should be followed:

1. Download BLD112 drivers for Windows at:
<https://www.silabs.com/documents/login/software/BLED112-Signed-Win-Drv.zip>
2. Extract the files inside the .zip (extracted files and the parent .zip file can be deleted after the installation procedure);
3. Access to the Windows “Device Manager”. This tool can be reached by clicking on the “Start Menu” icon on the bottom-left corner of the screen, followed by typing “Device Manager” (or the respective translation accordingly to the native language);
4. After attaching the BLD112 dongle to an USB port, it should appear in the “Other Devices” section with a warning icon flagging a non-functional device;

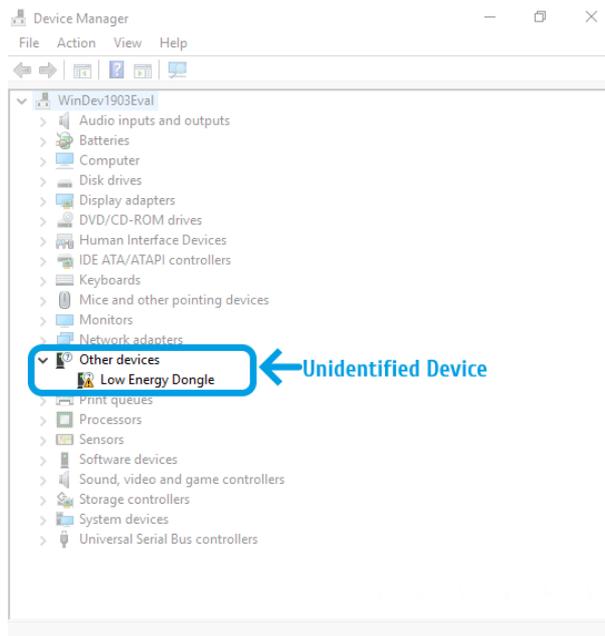


Figure 5: Unidentified BLD112 dongle.

¹ <https://www.pluxbiosignals.com/products/bluetooth-dongle?variant=41293957169343>

5. Right-mouse click on the “Low Energy Dongle” item. The “Update driver” option should be selected;

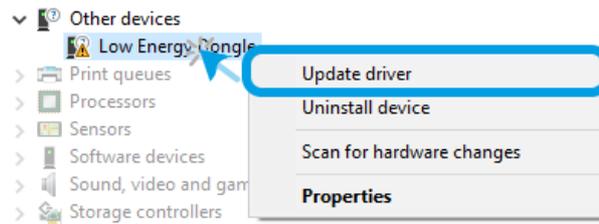


Figure 6: Update driver option which gives access to the manual installation of Bluegiga drivers.

6. Click on “Browse my computer for driver software”;

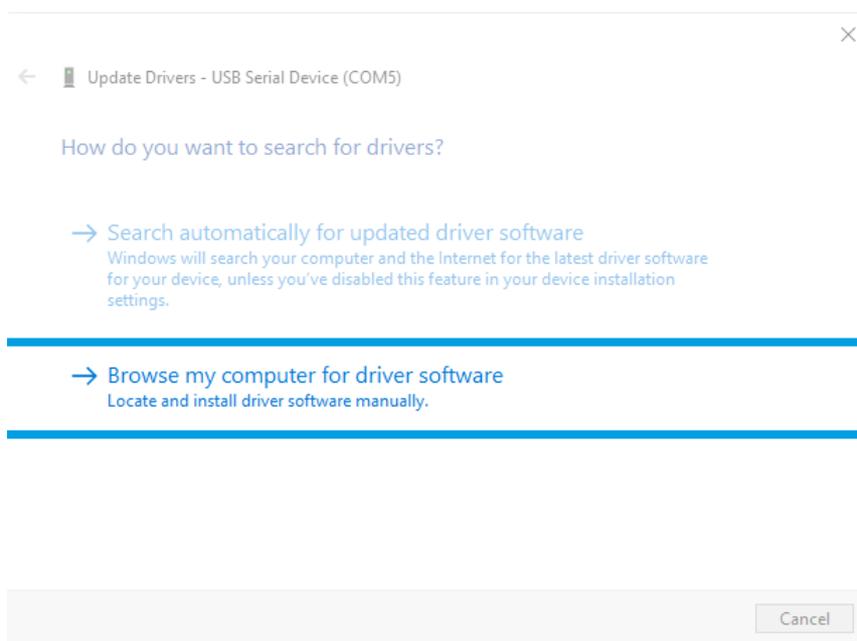


Figure 7: Definition of the type of searching methodology.

7. Search for the folder extracted on step 2, select “windrv”, click “OK” and “Next”;

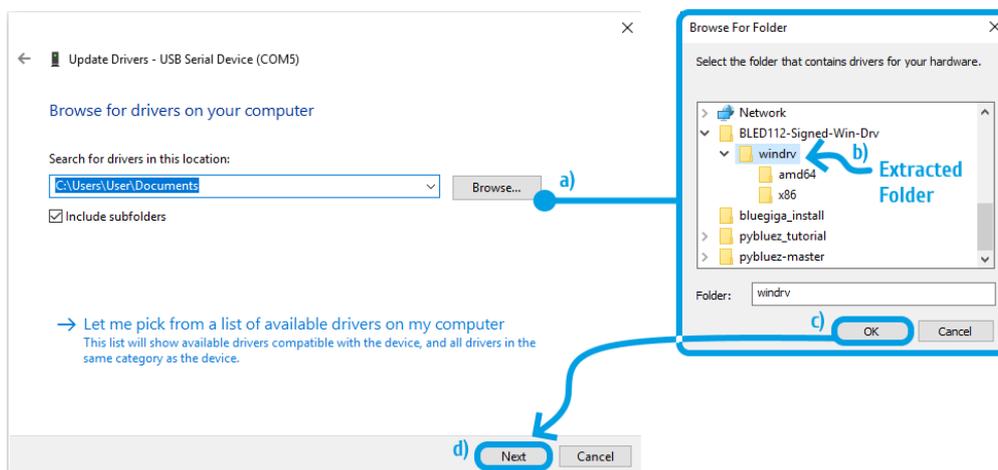


Figure 8: Navigate through directories and select the extracted folder containing BLD112 drivers.

- The previous action triggers the start of the installation. A confirmation dialog will appear, requesting the permission of the user to install the drivers. Click on “Install” button;

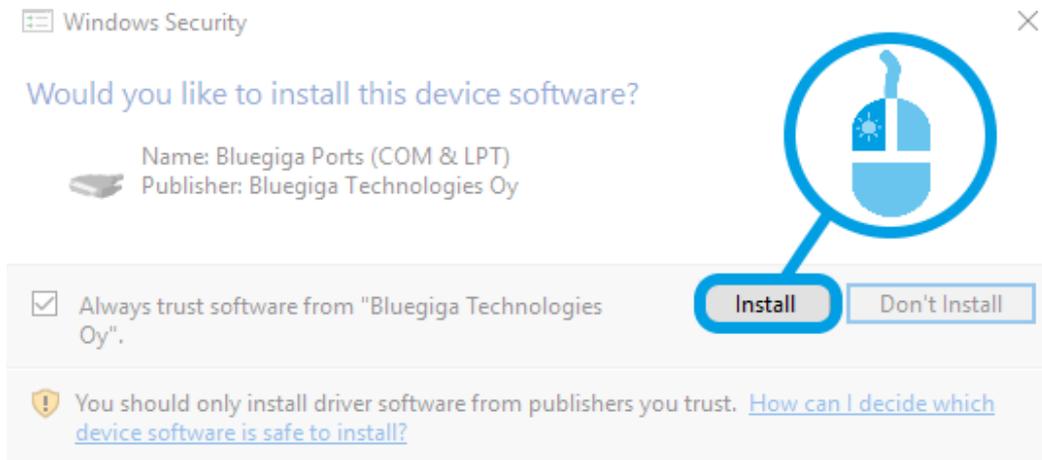


Figure 9: Confirm the installation intention.

- After a few seconds the installation procedure should be completed with success and the BLD112 dongle is now recognized as a COM port.

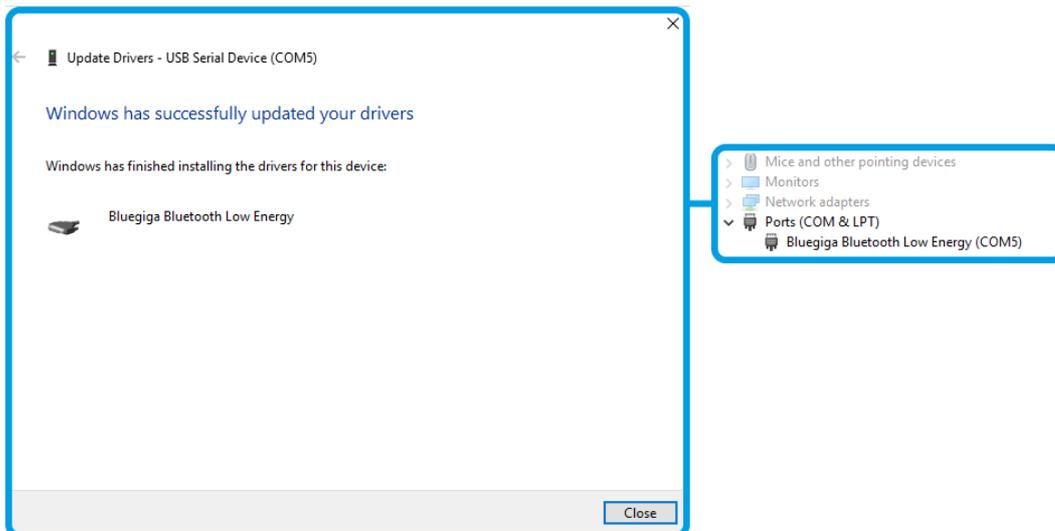


Figure 10: Check the state change of BLD112 dongle.

1.4.1 Windows 7

To connect a PLUX's device to your computer via Bluetooth, click on the start button at the right lower part of the Windows taskbar and click on *Control Panel* to access the control panel of your computer.

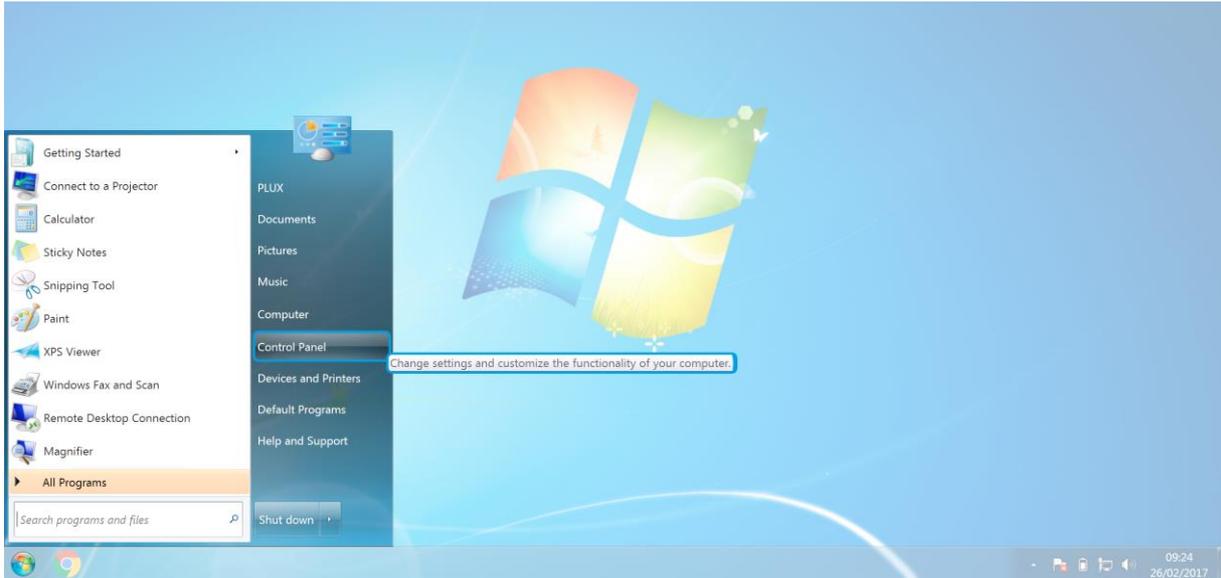


Figure 11: Access the Windows 7 control panel.

In the control panel, select the *Add a device* option in the *Hardware and Sound* section.

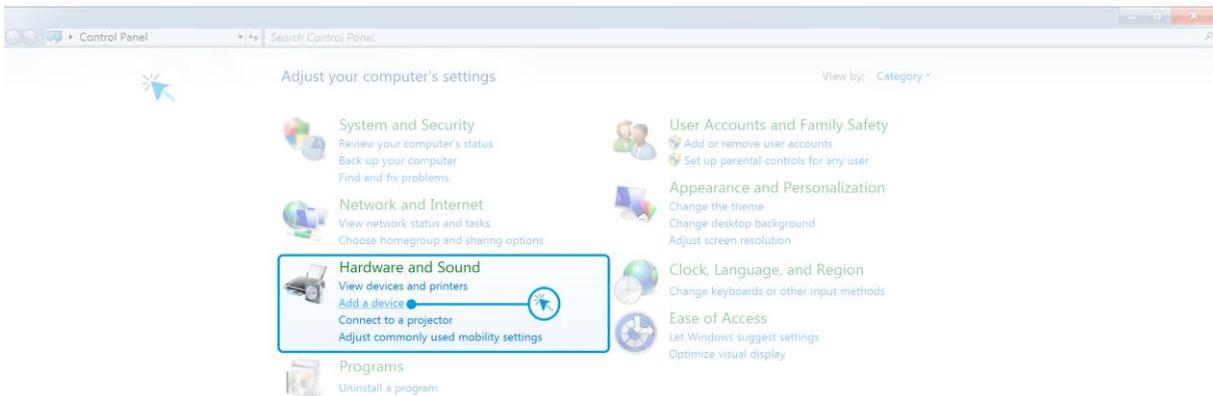


Figure 12: Select Add a device in the Hardware and Sound section.

At this point, turn on your *biosignalsplux* or *BITalino* (if not done before) and wait until it appears in the list of available devices. When your device has been found (here: *biosignalsplux*), click on the device and press *Next* at the lower right corner of the *Add a device* window.

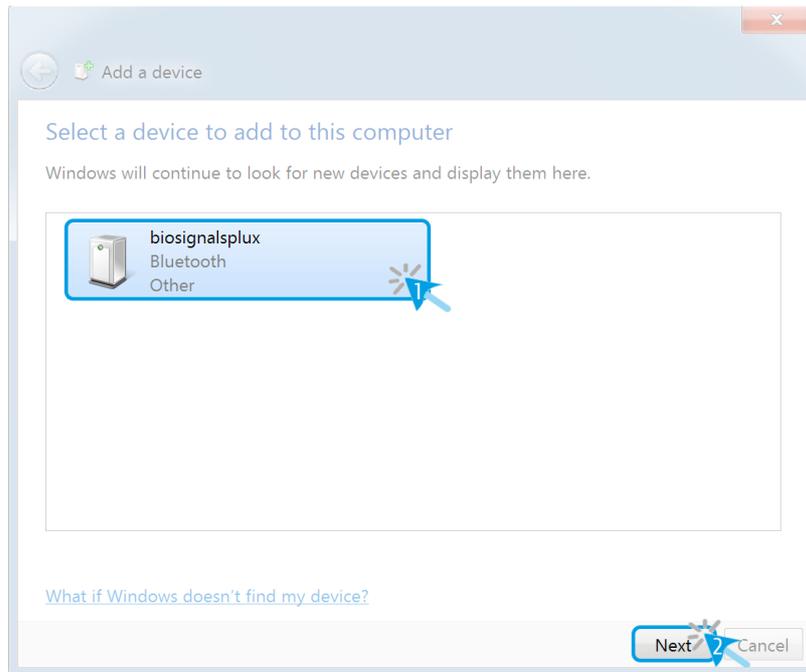


Figure 13: Select your device to add it to your computer.

Click on *Enter the device's pairing code* and click on *Next* at the lower right corner of the window to proceed.

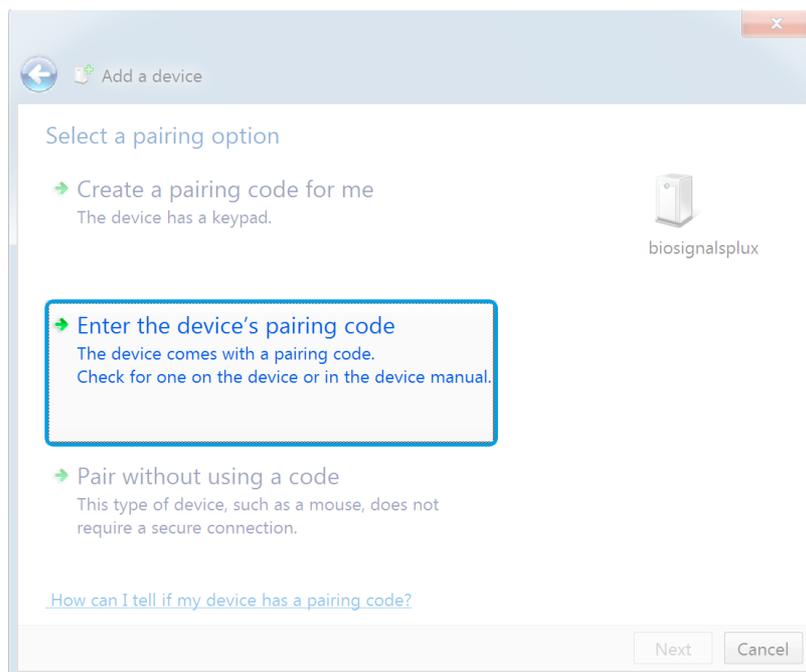


Figure 14: Select *Enter the device's pairing code*.

For *biosignalsplux* devices enter **123** and click on *Next* to connect to your device. For *BITalino* devices enter **1234** and click on *Next* to connect to your device.

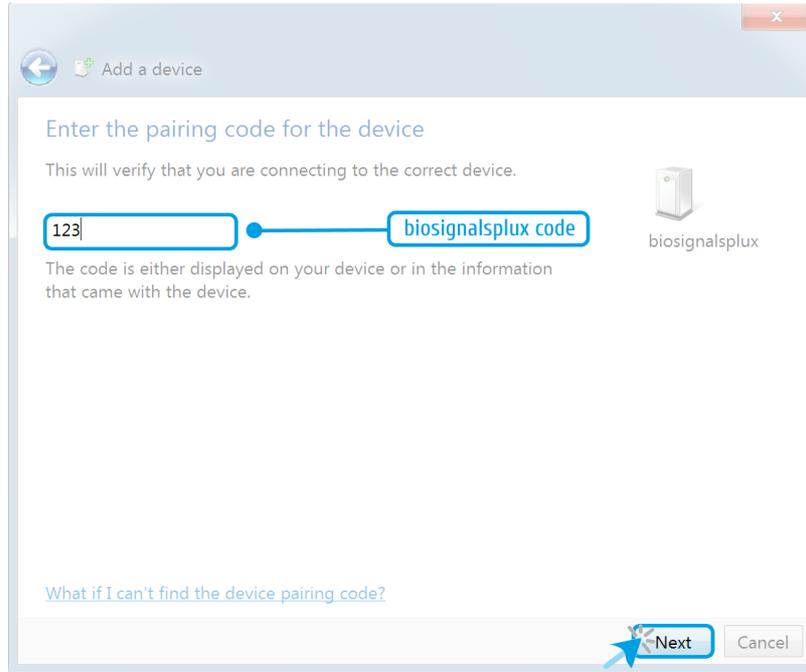


Figure 15: Enter the pairing code to connect to your device.

If Windows confirms that your device has been successfully added to your computer, the connection has been successfully established.

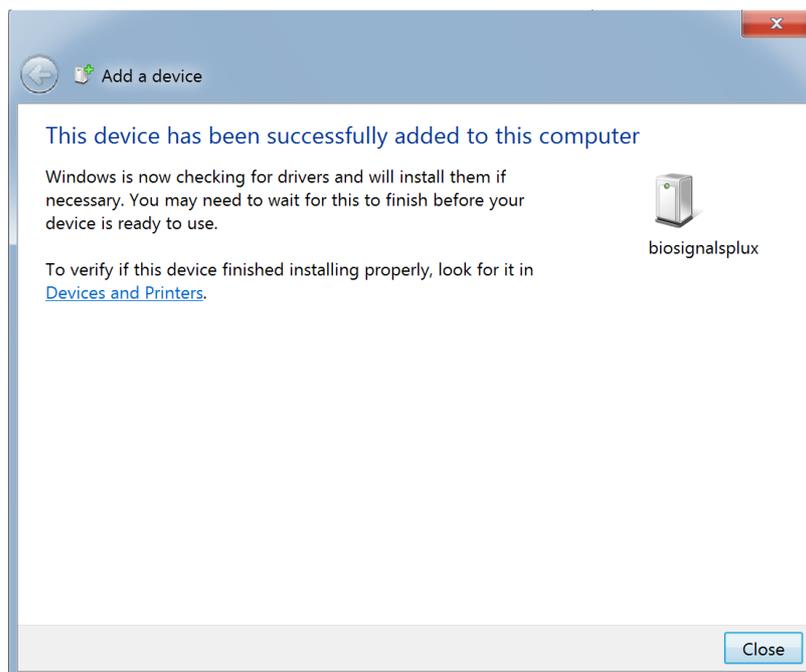


Figure 16: Windows confirmation if the device has been successfully added.

1.4.2 Windows 8

To connect your *biosignalsplex* device to your computer via Bluetooth, **a)** press the triangle symbol at the right lower part of the Windows taskbar and **b)** click on the Bluetooth icon to open the Bluetooth settings panel.



Figure 17: Access to the Bluetooth settings panel in Windows 8.

At this point, turn on your *biosignalsplex* (if not done before) and wait until it appears in the list of Bluetooth devices. When your *biosignalsplex* device has been found, click on the device and click on *Pair* to connect to your device.

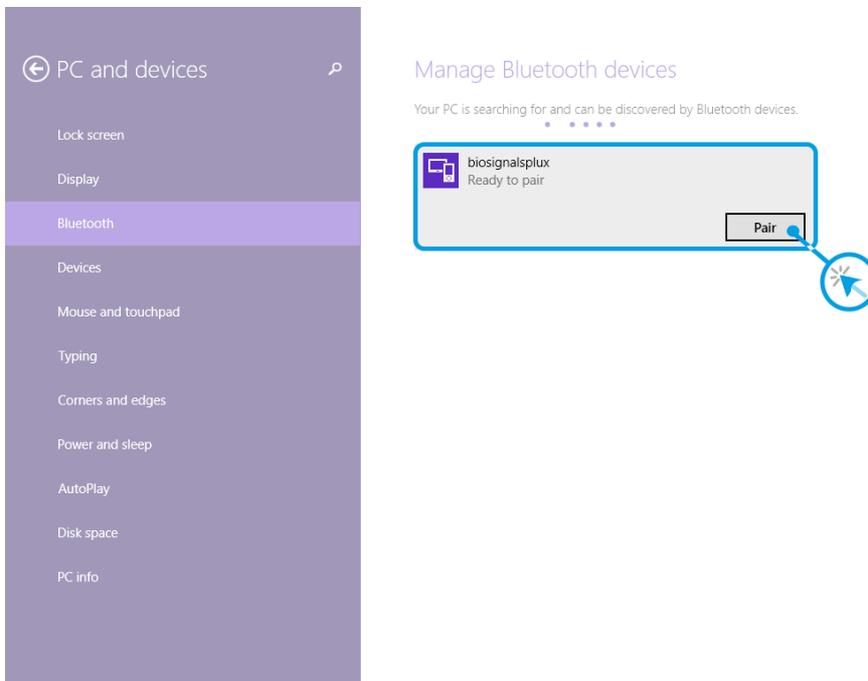


Figure 18: Bluetooth settings panel.

Windows will now try to connect to your device and pop up a window requesting you to enter the passcode (pin) of your device. Enter **123** (to connect to *biosignalsplex*) or **1234** (to connect to *BITalino*) and click on *Next*.

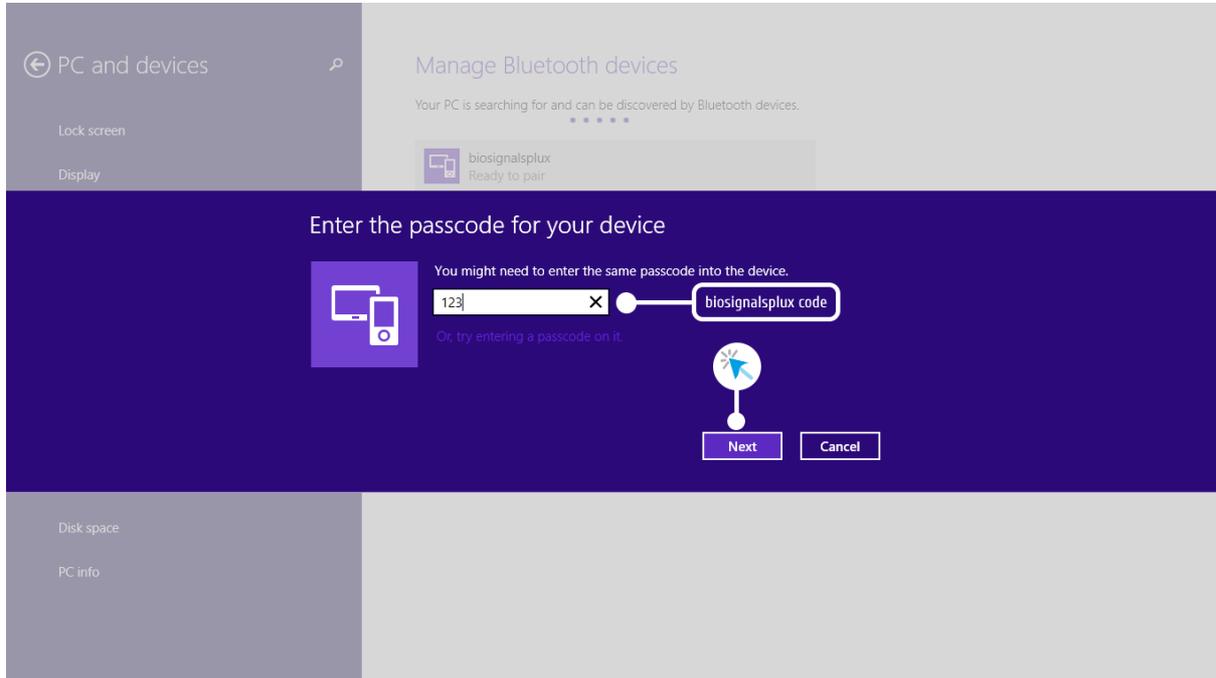


Figure 19: Requesting biosignalsplux' passcode (123).

If *Connected* is displayed under the name of your *biosignalsplux* device in the Bluetooth settings panel, the connection has been successfully established.

1.4.3 Windows 10

To connect a *biosignalsplux* or *BITalino* device to your computer via Bluetooth, click on the start button at the left lower corner in Windows 10 to open the start menu. Click on the gear symbol to open the settings panel of your operating system as can be seen below.

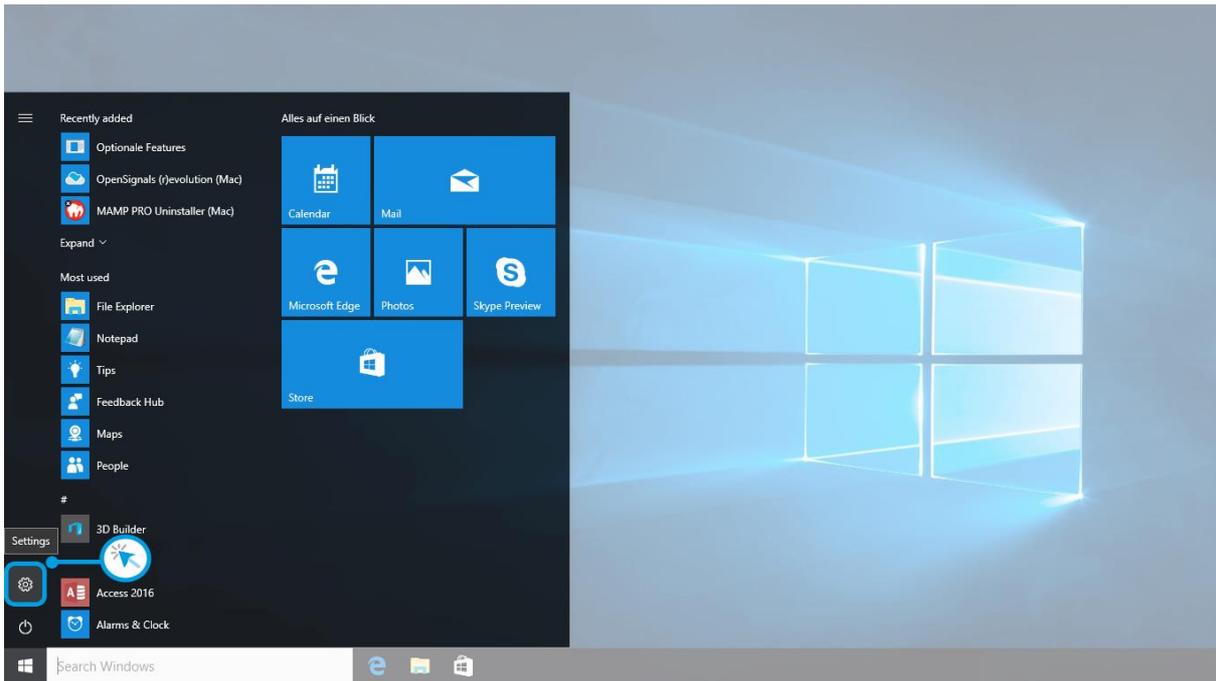


Figure 20: Access to the settings panel in Windows 10.

In the settings panel, click on the *Devices Bluetooth, printers, mouse* field to open the device configurations of your system.

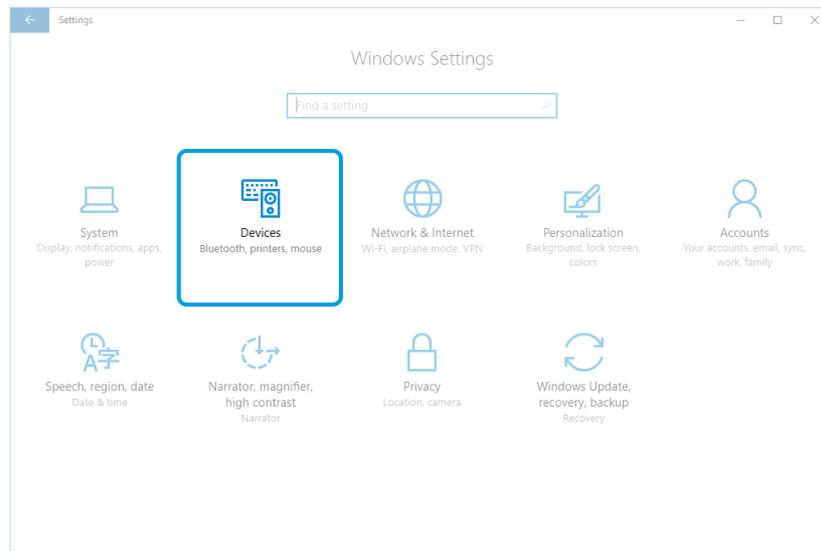


Figure 21: Windows 10 settings panel.

Select *Bluetooth* from the list, which is displayed on the left side of your settings window. Note, that in some versions the Bluetooth option might not be available and that Bluetooth devices might be listed under *Other devices* as seen in Figure 22.

At this point, turn on your *biosignalsplux* or *BITalino* (if not done before) and wait until it appears in the list of Bluetooth devices. When your device has been found, click on the device and press *Ready to pair* in order to connect to your device.

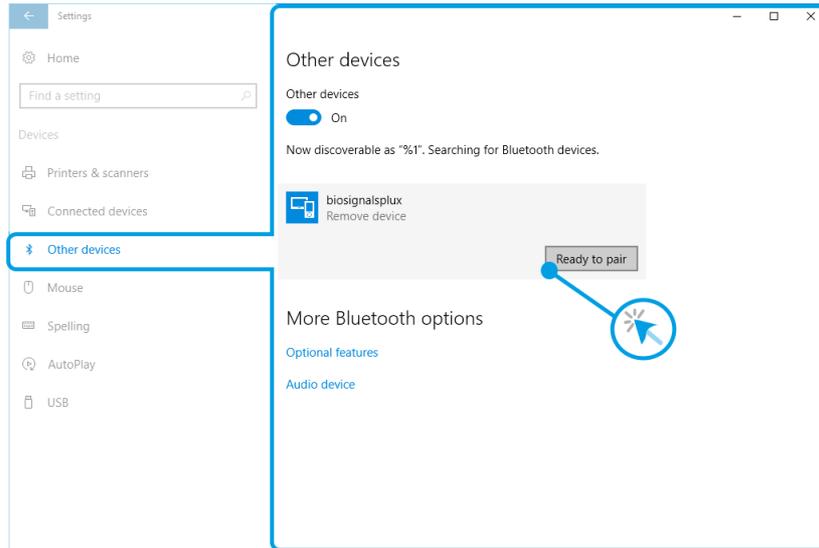


Figure 22: Bluetooth settings panel.

Windows will now try to connect to your device and pop up a window requesting you to enter the passcode (pin) of your device. Enter **123** (*biosignalsplux*) or **1234** (*BITalino*) and click on next to connect to the device.

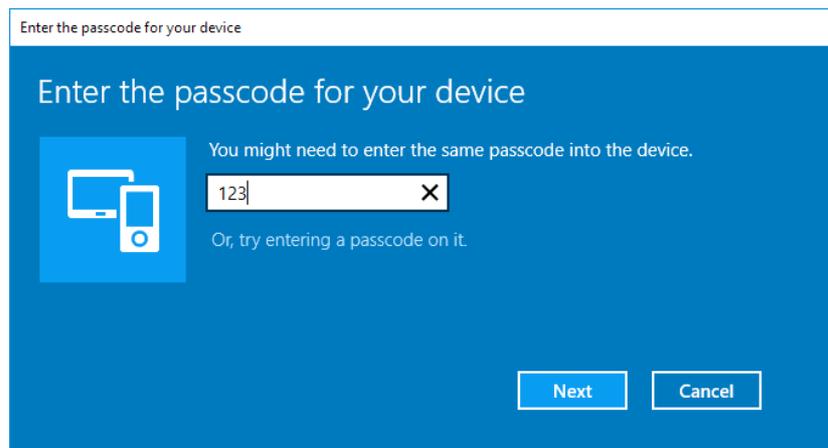


Figure 23: Requesting passcode (123).

If *Pair* is displayed under the name of your *biosignalsplux* device in the Bluetooth settings panel, the connection has been successfully established.

1.4.4 Linux

Ubuntu offers two options on how to connect PLUX's devices via Bluetooth to your computer. The first option allows you to connect the device via Ubuntu's system settings and the graphical interface. The second option allows you to connect the device via the terminal. However, the second option is recommended for advanced users only.

1.4.4.1 Connecting via System Settings

To connect your device to your computer via Bluetooth, click on the wheel in the top right corner of the display and click on *System Settings...* to access the settings panel of your computer.

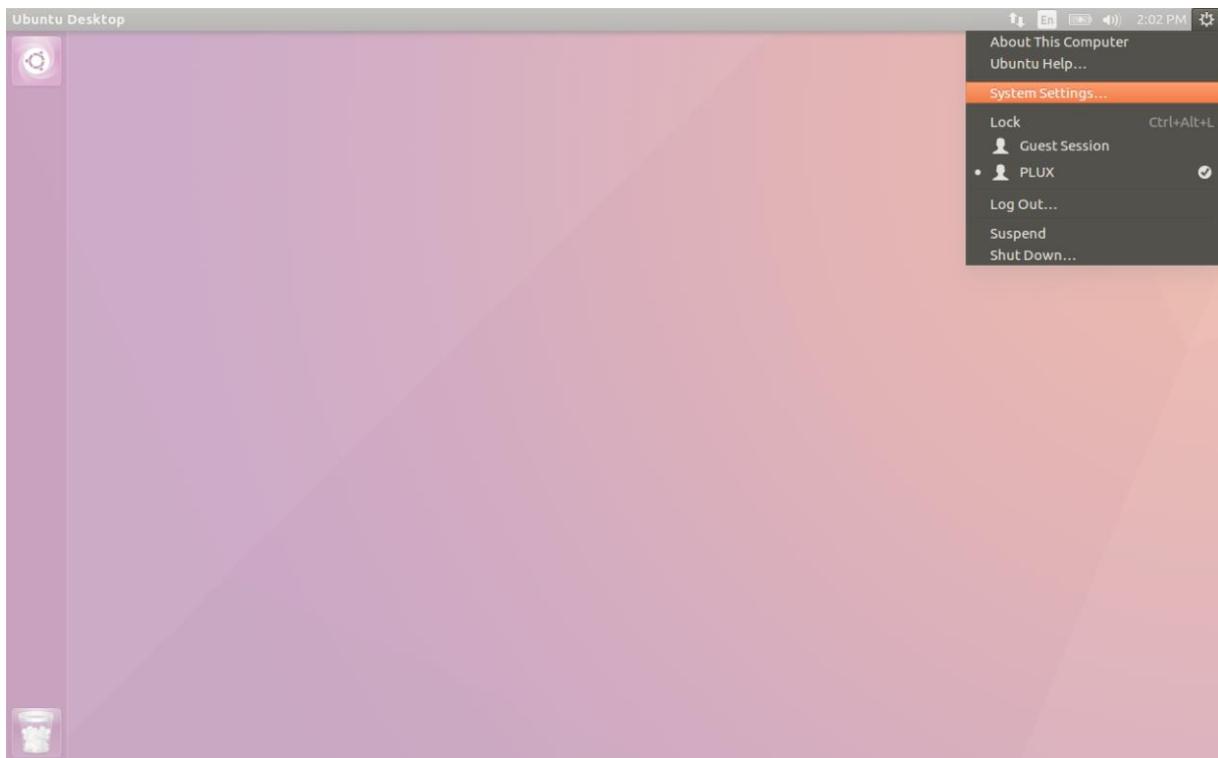


Figure 24: Access the system settings in Ubuntu.

In the settings panel, click on the Bluetooth logo to access the Bluetooth settings panel.

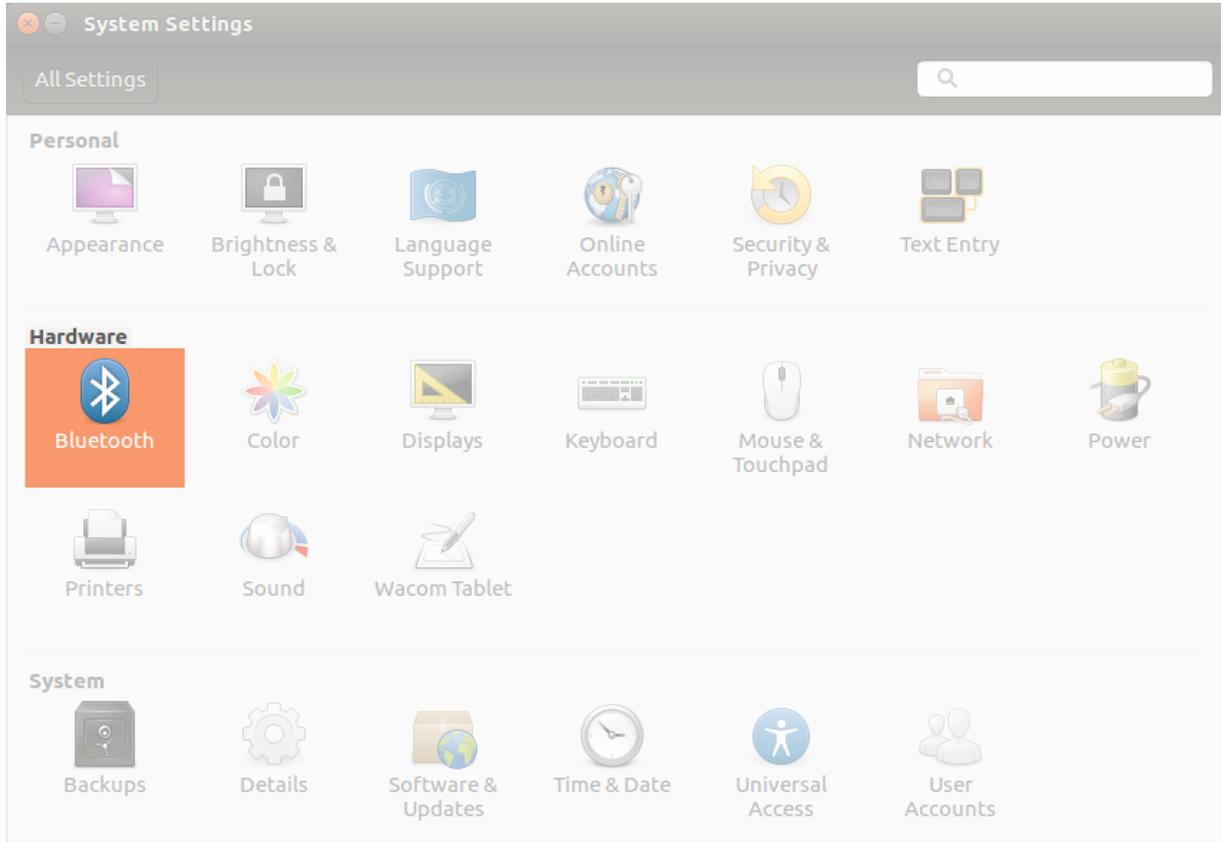


Figure 25: Click on the Bluetooth logo to access the Bluetooth settings.

At this point, turn on your *biosignalsplux* or *BITalino* (if not done before) and wait until it appears in the list of available devices. When your device has been found (here: *biosignalsplux*), it will be listed in the *Device Search* window: **a)** click on the device and **b)** select *PIN options...* to enter the device’s PIN to proceed.

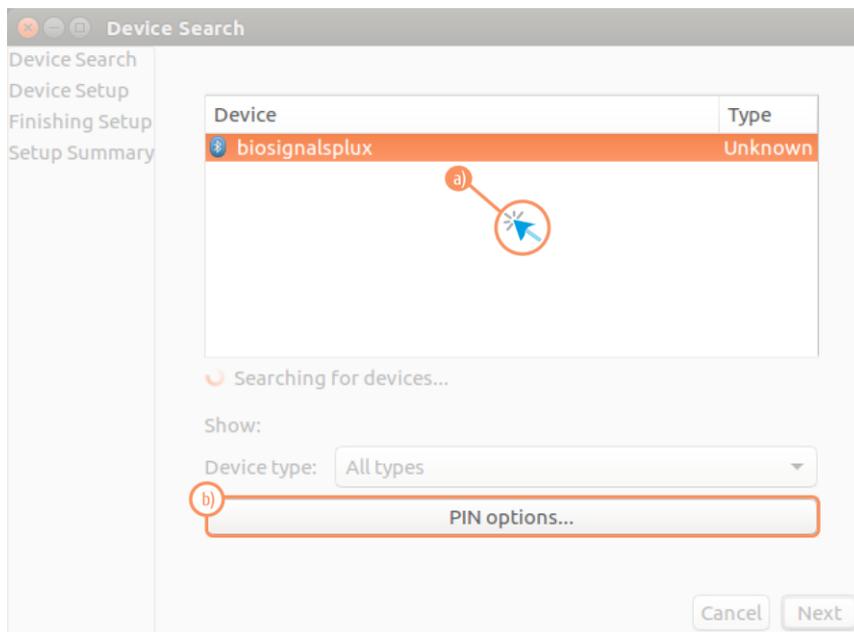


Figure 26: Select your device to add it to your computer.

In the new *PIN* Options window click on *Custom PIN* and enter the device's PIN to connect to your device. For *biosignalsplux* devices enter **123** (while for *BITalino* the pin will be **1234**) and click on *Next* to connect to your device.

NOTE

If the *Custom PIN* option is not available in this window, follow up with the instructions in *1.4.4.2 Connecting via Terminal* to connect your device to your computer using the terminal instead of proceeding with the configuration via the system settings.

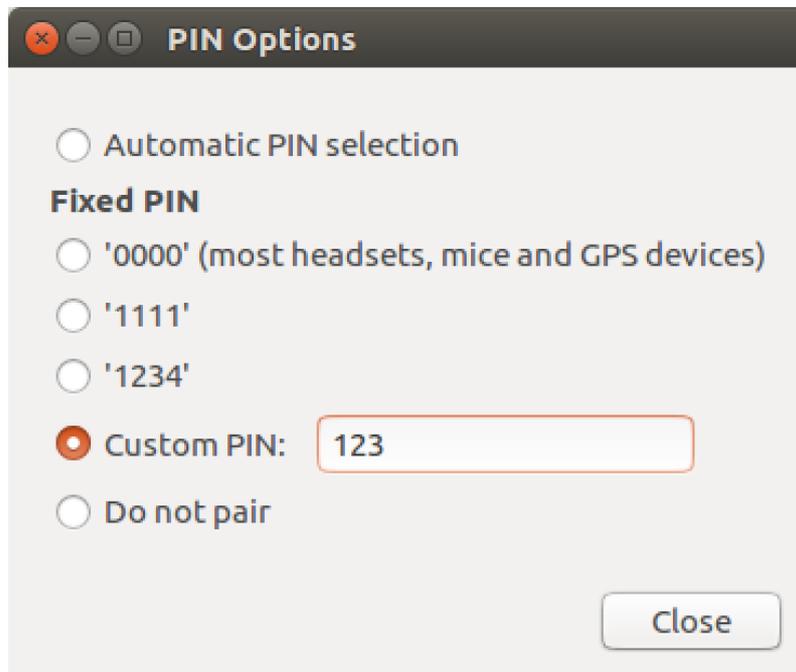


Figure 27: Enter the device's pin to pair.

Click on *Close* to close this window and click on *Next* on the *Device Search* window to connect to your device.

Click on your device in the *Devices* list. If paired is confirmed (*Paired Yes*) your device has been successfully connected to your computer².

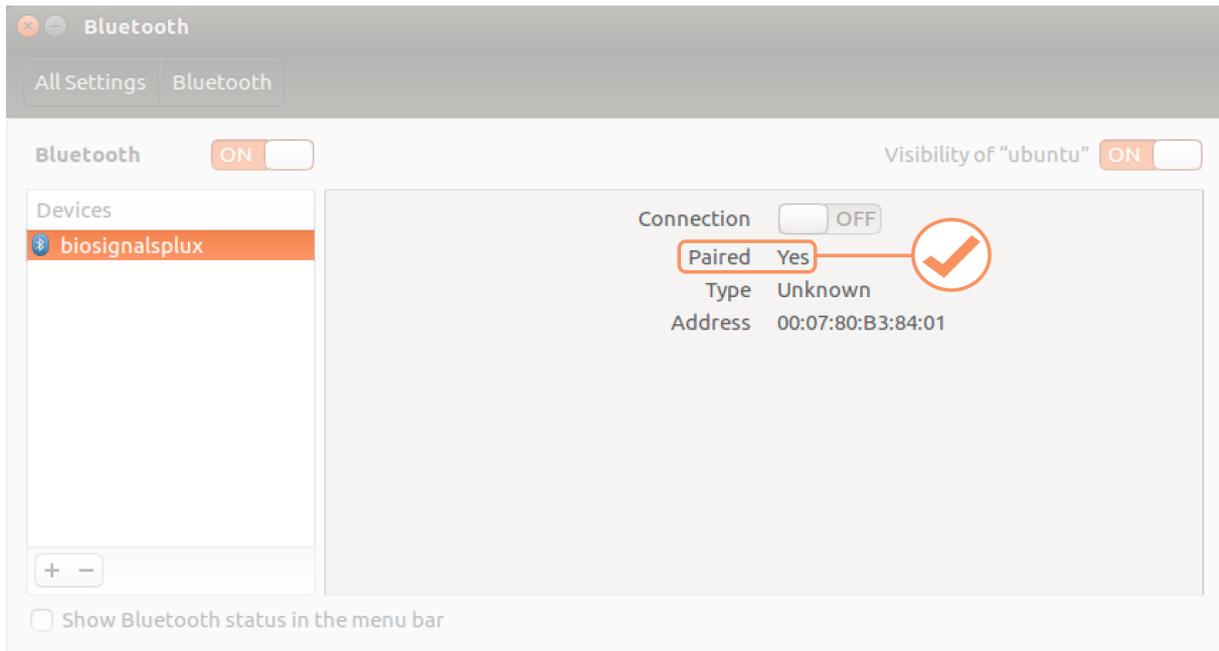


Figure 28: Successfully paired biosignalsplx.

² Do not worry if the connection is set off. It will automatically switch to on when connecting to your device using *OpenSignals (r)evolution* or other custom software for *biosignalsplx*.

1.4.4.2 Connecting via Terminal

NOTE

Using the terminal may harm your computer if it is not done correctly. Therefore, this method is recommended for advanced users only and following this method is done at your own risk.

To connect your device to your computer via Bluetooth, **a)** click on the *Unity Dash* logo in the toolbar on the left of your desktop to open the search function. In the search bar, **b)** enter terminal (or Terminal) and **c)** click on the terminal logo to open the *Terminal*.

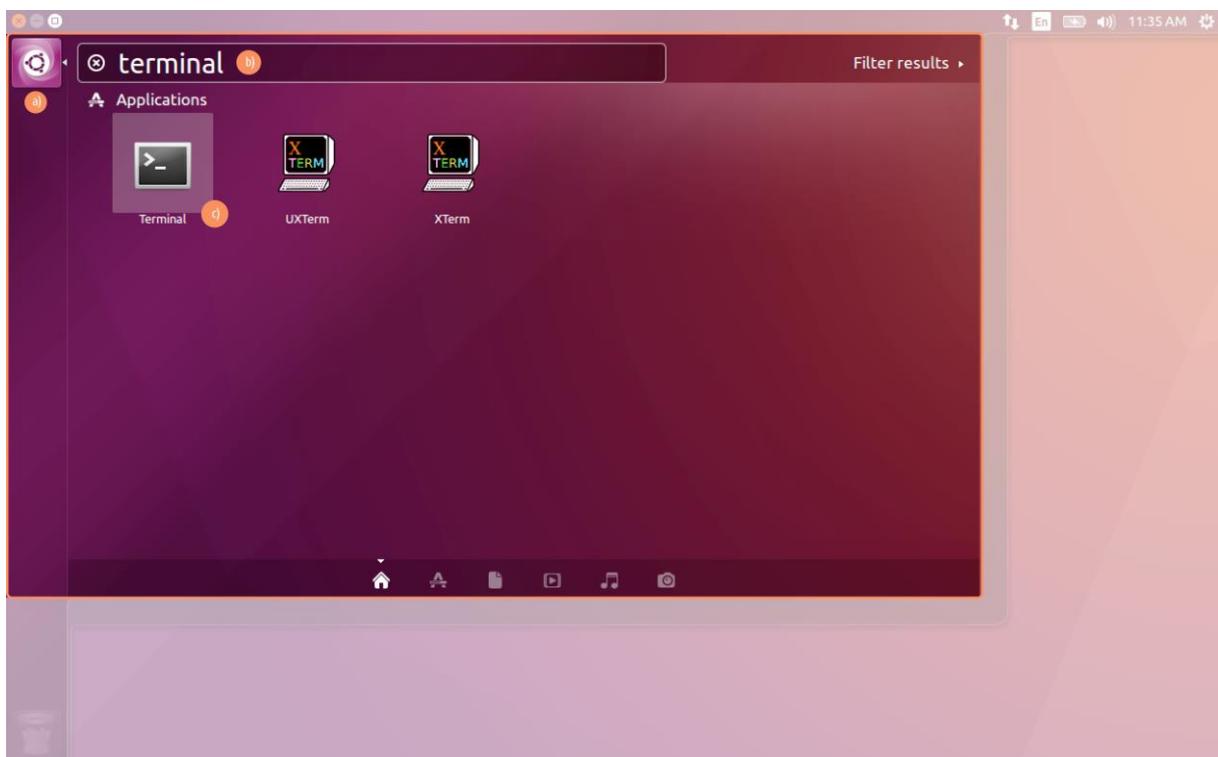


Figure 29: Access the terminal using the search function in Ubuntu.

Enter `bluetoothctl` into the terminal window (without the \$). Terminal should answer to this command with a line displaying the MAC-address of your internal Bluetooth adapter. The response should be similar to this (with AA:BB:CC:DD:EE:FF being replaced with your MAC-address):

```
$ bluetoothctl
[NEW] Controller AA:BB:CC:DD:EE:FF ubuntu [default]
[bluetooth]#
```

Next, turn on your internal Bluetooth module (if not done before) by entering *power on* into the Bluetooth prompt line. The response should be similar to this (again, with AA:BB:CC:DD:EE:FF being replaced with your MAC-address):

```
[bluetooth]# power on
Changing power on succeeded
[CHG] Controller AA:BB:CC:DD:EE:FF Powered: yes
```

Next, it is needed to turn on a Bluetooth agent. The Bluetooth agents manages the transmission of pin codes (or pairing codes) between devices and is needed here to send the pairing code to the device you want to connect. To do this, enter *agent on*. The terminal response should be similar to this:

```
[bluetooth]# agent on
Agent registered
```

Set up the *default-agent* to proceed with the pairing process.

```
[bluetooth]# default-agent
Default agent request successful
```

At this point, turn on your device (if not done before) and enter *scan on* into the terminal to search for available Bluetooth devices and wait until your device's MAC-address appears in the list (here: biosignalsplux, 00:07:80:B3:84:01). Your device's MAC-address can be found at the back of the device.

```
[bluetooth]# scan on
Discovery started
[CHG] Controller AA:BB:CC:DD:EE:FF Powered: yes
[NEW] Device 00:07:80:B3:84:01 biosignalsplux
```

If your device is listed, enter *pair* followed by the MAC-address of your device to start the pairing process which will be respond with requesting the PIN code of your device.

For *biosignalsplux* devices enter 123 and for *BITalino* devices enter 1234 to connect to your device.

The result should be similar to this:

```
[bluetooth]# pair 00:07:80:B3:84:01
Attempting to pair with 00:07:80:B3:84:01
Request PIN code
[agent] Enter PIN code: 123
[CHG] Device 00:07:80:B3:84:01 Connected: yes
[CHG] Device 00:07:80:B3:84:01 Modalias: bluetooth
[CHG] Device 00:07:80:B3:84:01 UUIDs: (device specific ID)
[CHG] Device 00:07:80:B3:84:01 UUIDs: (device specific ID)
[CHG] Device 00:07:80:B3:84:01 Paired: yes
Pairing successful
[biosignalsplux]
```

If the terminal outputs *Pairing successful*, your device has been successfully paired to your computer.

The entire text of your terminal should now be similar to the following:

```
$ bluetoothctl
[NEW] Controller AA:BB:CC:DD:EE:FF ubuntu [default]
[bluetooth]#
[bluetooth]# power on
Changing power on succeeded
[CHG] Controller AA:BB:CC:DD:EE:FF Powered: yes
[bluetooth]# agent on
Agent registered
[bluetooth]# default-agent
Default agent request successful
[bluetooth]# default-agent
Default agent request successful
Discovery started
[CHG] Controller AA:BB:CC:DD:EE:FF Powered: yes
[NEW] Device 00:07:80:B3:84:01 biosignalsplux
[bluetooth]# pair 00:07:80:B3:84:01
Attempting to pair with 00:07:80:B3:84:01
Request PIN code
[agent] Enter PIN code: 123
[CHG] Device 00:07:80:B3:84:01 Connected: yes
[CHG] Device 00:07:80:B3:84:01 Modalias: bluetooth
[CHG] Device 00:07:80:B3:84:01 UUIDs: (device specific ID)
[CHG] Device 00:07:80:B3:84:01 UUIDs: (device specific ID)
[CHG] Device 00:07:80:B3:84:01 Paired: yes
Pairing successful
[biosignalsplux]
```

1.4.5 Mac OS X

To connect your *biosignalsplex* device to your computer via Bluetooth, click on the Apple symbol at the top left corner of your display and select *System Preferences*...

In the system preferences window click on the Bluetooth symbol to open the Bluetooth settings.

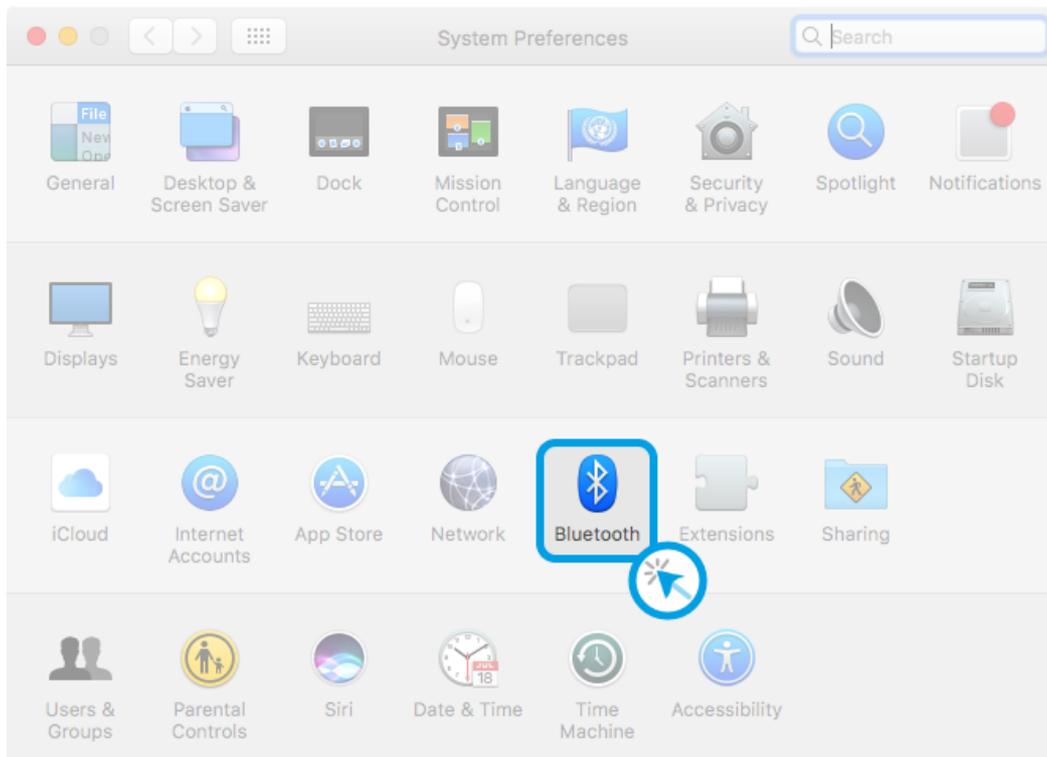


Figure 30: Mac OS X system preferences window.

At this point, turn on your *biosignalsplex* (if not done before) and wait until it appears in the list of Bluetooth devices. When your *biosignalsplex* or *BITalino* device has been found, click on the device and click on *Pair* to connect to your device.

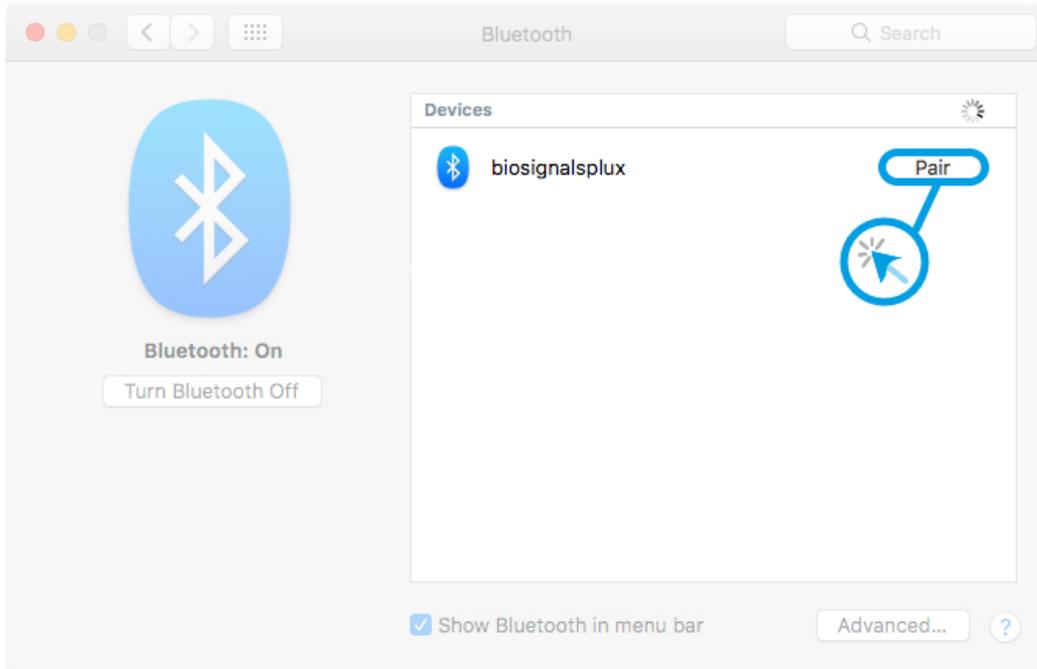


Figure 31: Click on Pair to connect to the biosignalsplux.

An error will occur indicating that the connection could not be established due to a mismatching passkey. Click on *Options...* to enter the passkey (pairing code).

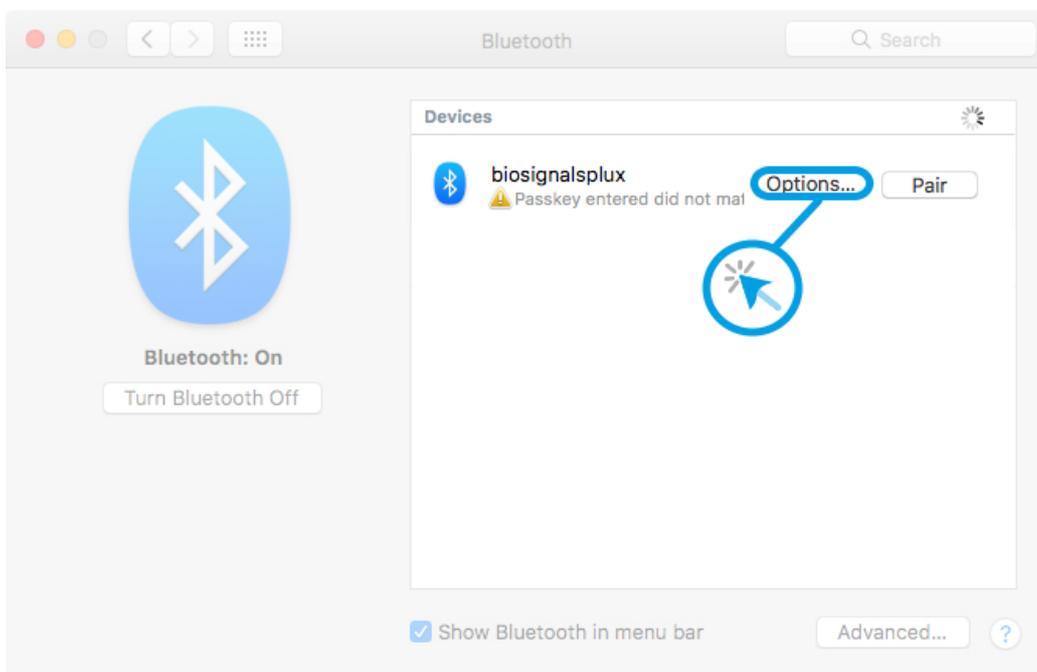


Figure 32: Click on Options... to enter the passkey.

Mac OS X will now try to connect to your device and pop up a window requesting you to enter the passcode (pin) of your device. Enter **123** (*biosignalsplux*) or **1234** (*BITalino*) and click on *Pair* to connect to *biosignalsplux*.

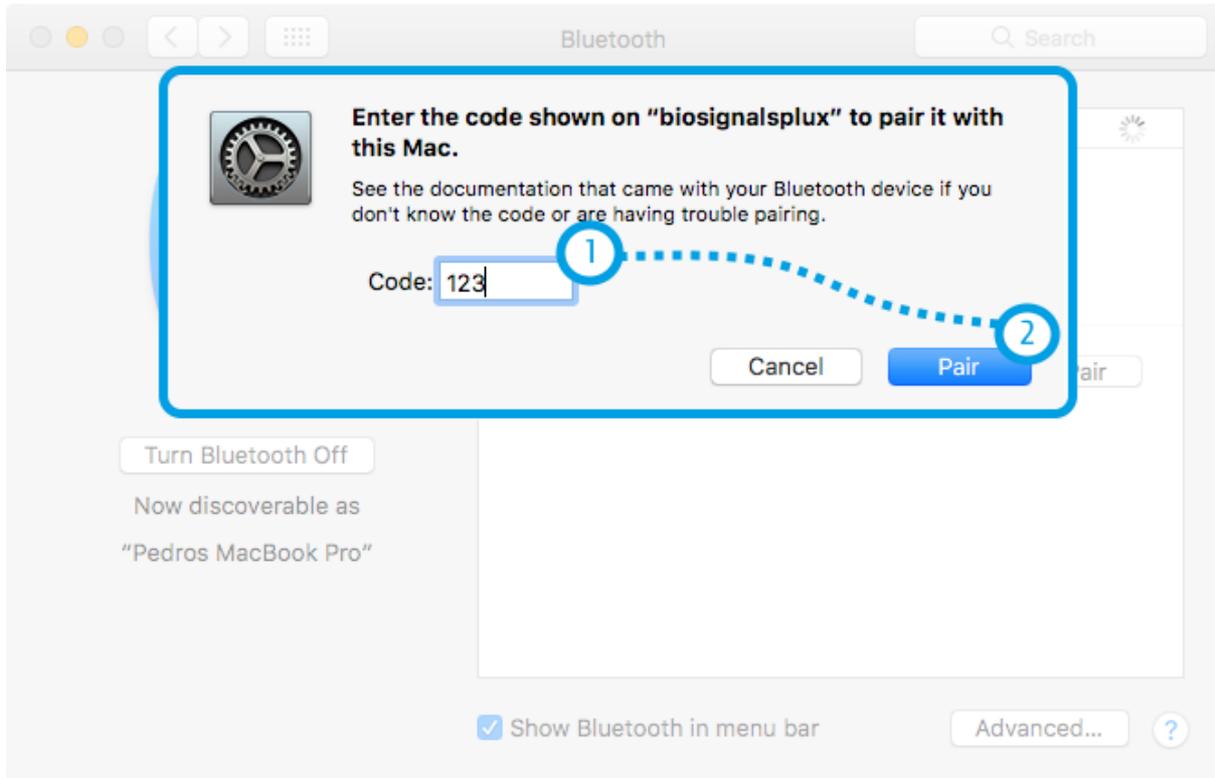


Figure 33: Enter the pairing code

If *Connected* is displayed under the name of your device in the Bluetooth settings panel, the connection has been successfully established.

1.5 OpenSignals Help

? contact information, license information, add-on activation and about section

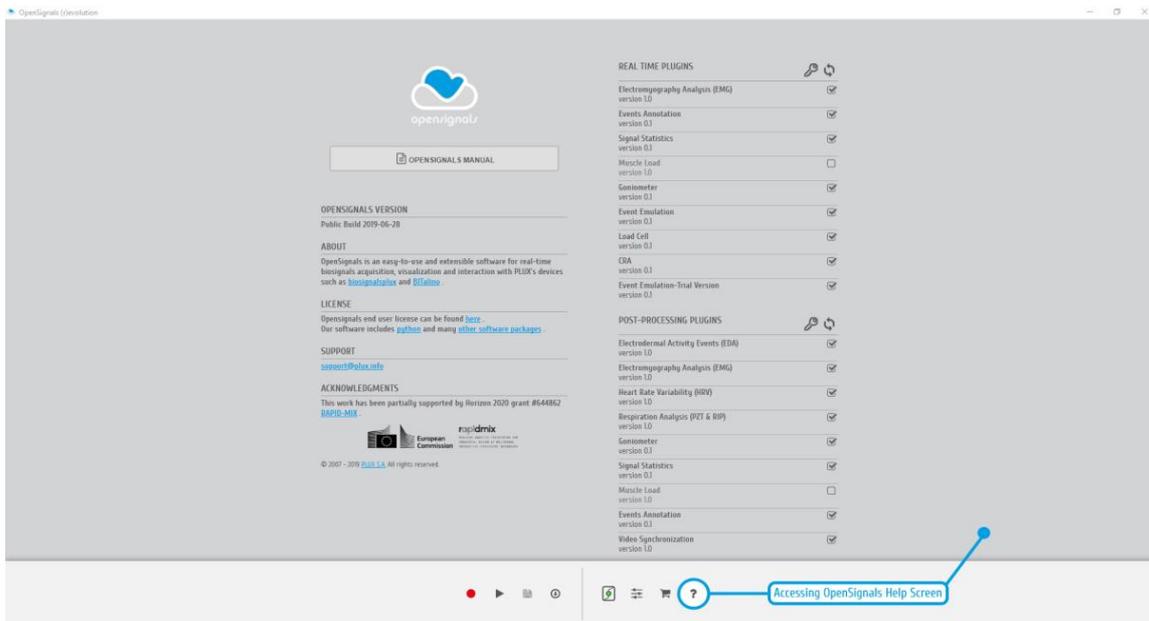


Figure 34: OpenSignals (r)evolution help interface

The help interface offers useful information and documents regarding *OpenSignals (r)evolution*. Here you can also check which plugins are activated for your device.

1.5.1 User Manual

The newest version of this manual can be accessed at any time by clicking on the *OPENSIGNALS MANUAL* button. We recommend accessing this manual via the *OpenSignals (r)evolution* help screen or by downloading it at:

<https://www.pluxbiosignals.com/pages/opensignals>

1.5.2 Support

Don't hesitate to contact us if you're struggling with any kind of problems that cannot be solved with the information displayed in this manual. Send us an e-mail with a brief description of your problem to (contact info available at PLUX support page: <https://support.pluxbiosignals.com/>) and we'll get back to you as soon as possible. Please do not forget to mention which version of *OpenSignals* you're using.

1.5.3 Access License Information

Access the *Open Signals Terms of Use* and license information and find out more about Python, the programming language behind *OpenSignals (r)evolution*, on the official Python homepage.

1.5.4 Activating & Deactivating Add-ons

Activate and deactivate add-ons that are available for your device (see section 4 *Add-Ons* for further information).

2 OpenSignals (r)evolution

After successfully connecting to a PLUX device on your computer, *OpenSignals (r)evolution* is ready to use.

We recommend to follow these steps in order to prepare and to start your acquisitions.

1. Turn on your device;
2. Start *OpenSignals (r)evolution*;
3. Open the device manager in *OpenSignals (r)evolution*;
4. Enable your device for acquisition and set up your preferred acquisition settings (sampling rate, channels configuration and sampling resolution);
5. Set up your general acquisition's settings (file name, save location(s)...);
6. Start the acquisition.

These steps are listed in the following sections in further detail to guide you through all the steps.

2.1 OpenSignals Main Screen

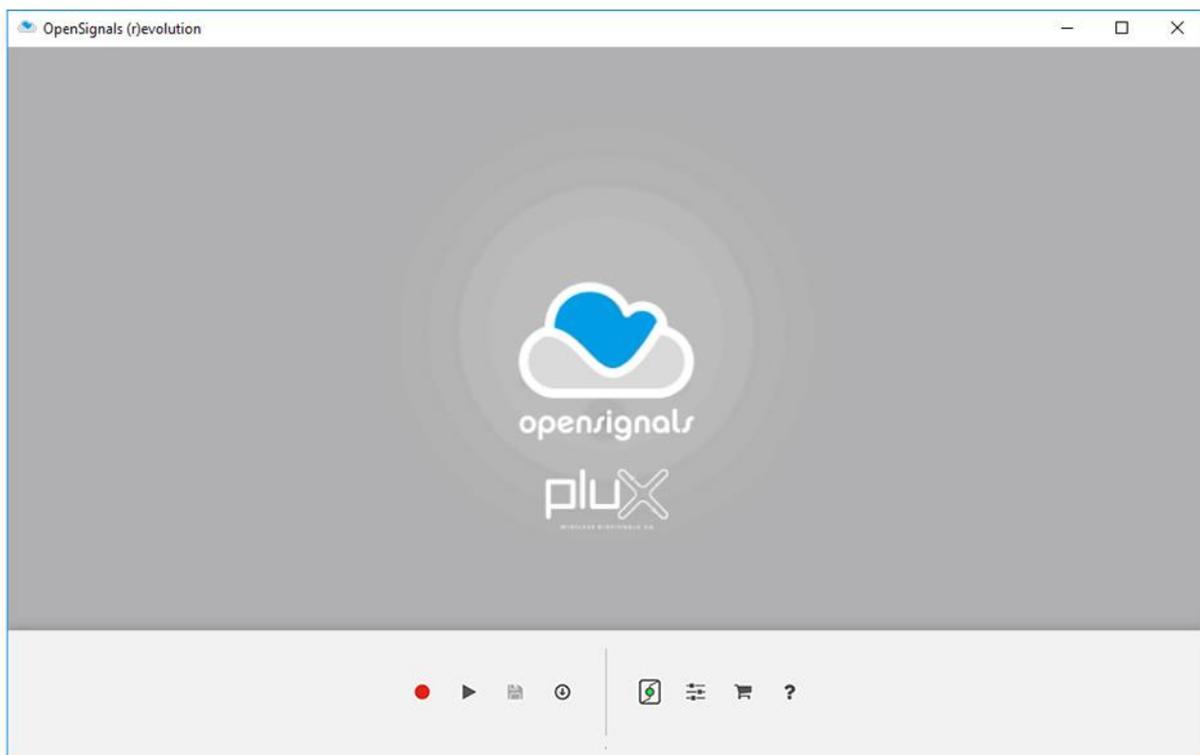


Figure 35: *OpenSignals (r)evolution* main interface.

All *OpenSignals (r)evolution* windows, settings and signal related modes are accessible via the main Screen.

-  Live (or Real-Time Acquisition) Mode
-  Playback (or Visualization and Post-processing) Mode
-  Save Last Recording
-  Download Logged Data from Device³
-  Device Manager – Find and configure your devices
-  Change *OpenSignals (r)evolution* settings
-  Check our online store
-  Help - Consult our manual, licensing, support, and add-ons

Active and inactive options are marked by blue and gray colors, respectively:

 Activated

³ Your device should have the scheduling functionality active. Only the devices with internal memory are eligible for “downloading” data and schedule an acquisition.

2.2 Device Manager



select device(s), channels, channel labels, trigger, sample rate & resolution

The *OpenSignals (r)evolution* device manager lists and manages all known and available *OpenSignals* compatible PLUX devices and their acquisition configurations.

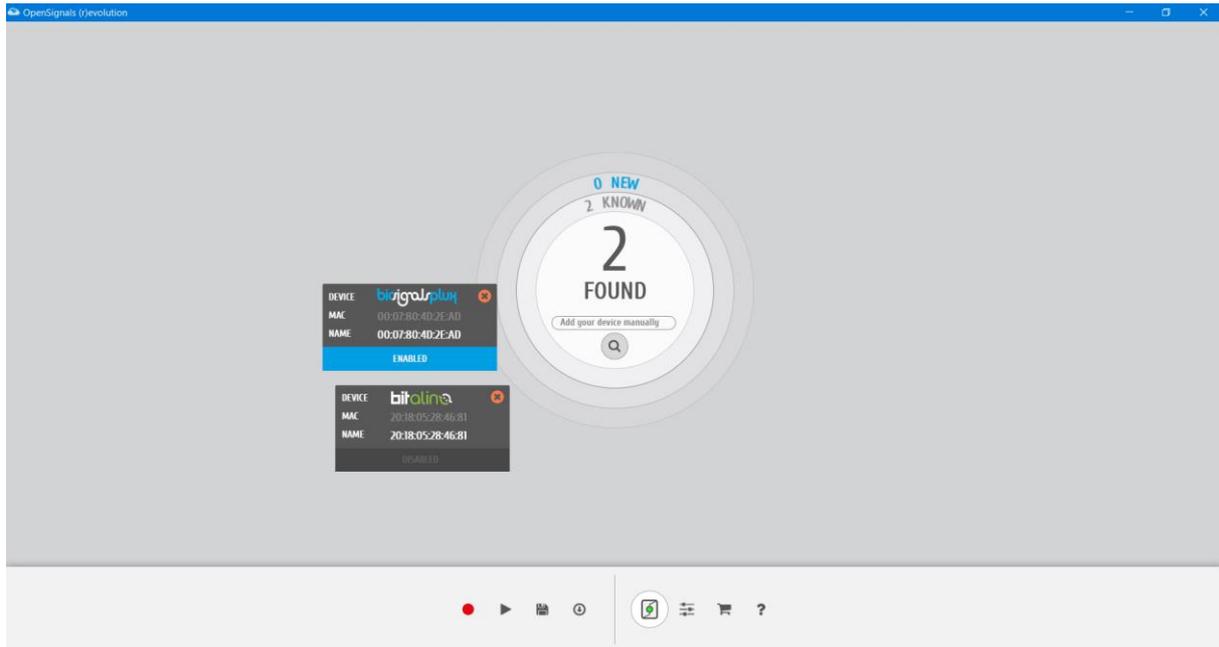


Figure 36: The device manager displays all the available PLUX devices.

NOTE

OpenSignals (r)evolution is able to establish a Bluetooth connection to your device whenever you add a PLUX device in the *OpenSignals (r)evolution* device manager. In this case, it should be needed to manually establish a Bluetooth connection with your device using your operating system's tools **if you are using Windows OS**.

Additionally, if you are experiencing any kind of connection issues, if your device cannot be found and listed in the device manager or if you are using macOS, we recommend to connect your device with your operating system first before adding your device in *OpenSignals (r)evolution's* device manager by following the instructions in [1.4 Connecting Bluetooth Devices](#).

TIP

If your screen is becoming populated with many devices, the most recent version of *OpenSignals (r)evolution* supports the “pan” feature, i.e., the user can drag the device manager screen to access devices out of the window view.

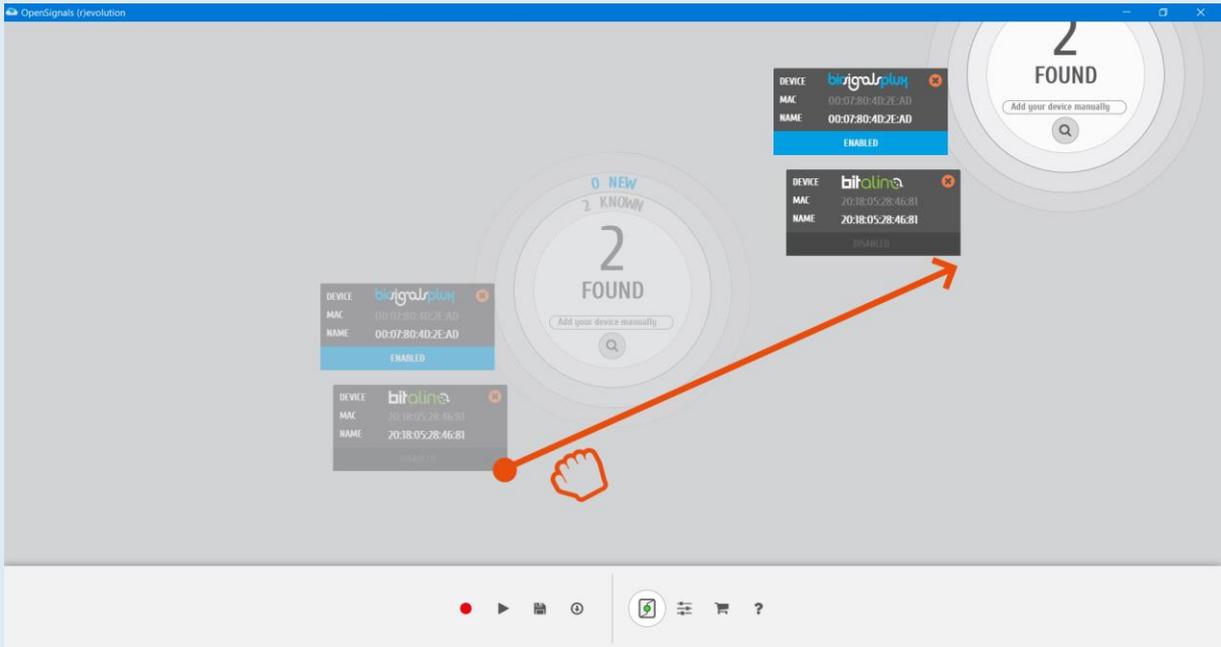


Figure 37: Illustrative example of the “pan” feature in the Device Manager screen.

Note, that the circle at the center of the icon provides additional information about whether there are known devices in the device manager or not.



known devices available in the device manager



no known devices available in the device manager

2.2.1 Connecting Devices

Click on the device manager to search for available devices. *OpenSignals (r)evolution* will automatically search for devices and display them in the device manager. Newly found devices will be displayed in blue with a loading circle in the top right corner of the device panel, indicating that *OpenSignals (r)evolution* is building up a connection to this device.

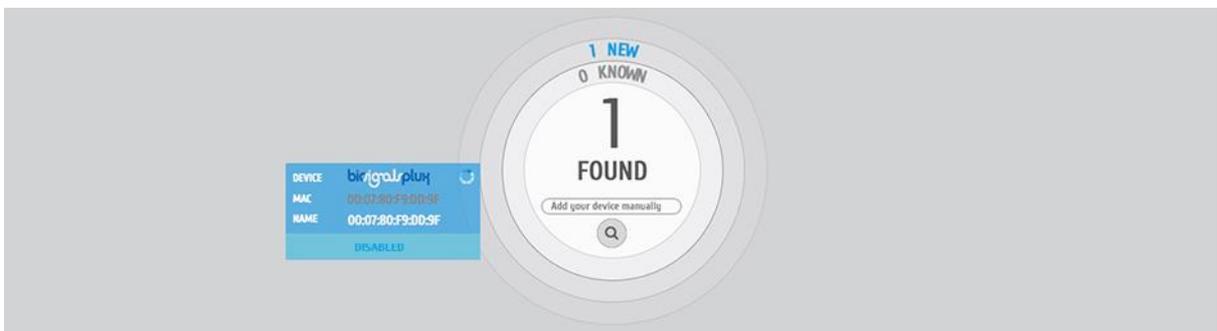


Figure 38: Building up connection with biosignalsplx.

If you haven't connected the device to your computer before, using your operating system's device manager, a new window will pop up requesting the device's pairing code or informing you about the need to execute this additional Bluetooth configuration. For *biosignalsplux* devices enter **123** and click on *Next* to connect to your device. For *BITalino* devices enter **1234** and click on *Next* to connect to your device.

2.2.2 Connecting Devices Manually

If your signal acquisition device is not listed as an available device to be used in *OpenSignals (r)evolution* you can manually add your device using its MAC-address. **a)** Enter the MAC-address of your device into the *Add your device manually* field and **b)** click on the magnifying glass to search for your device.

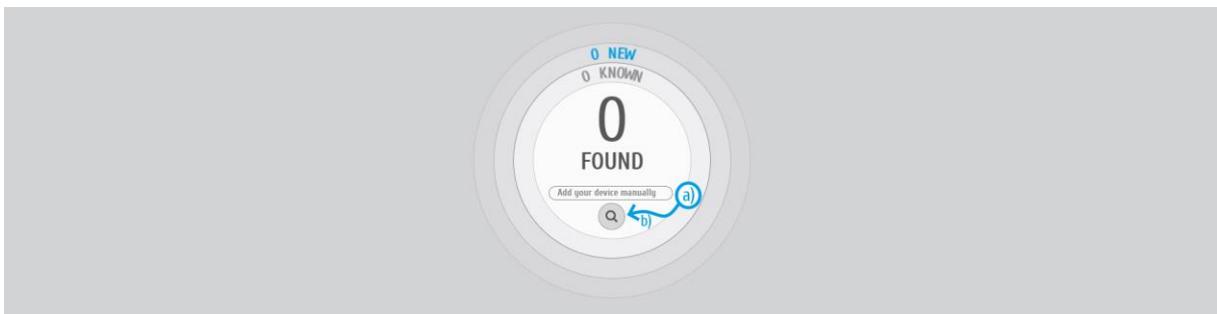


Figure 39: Use the 'Add your device manually' field to search for PLUX's devices using the MAC-address.

NOTE

You can find the MAC-address of your device on the backside of the device.

Follow the instructions displayed on your screen and your device should now appear and be available in the device manager.

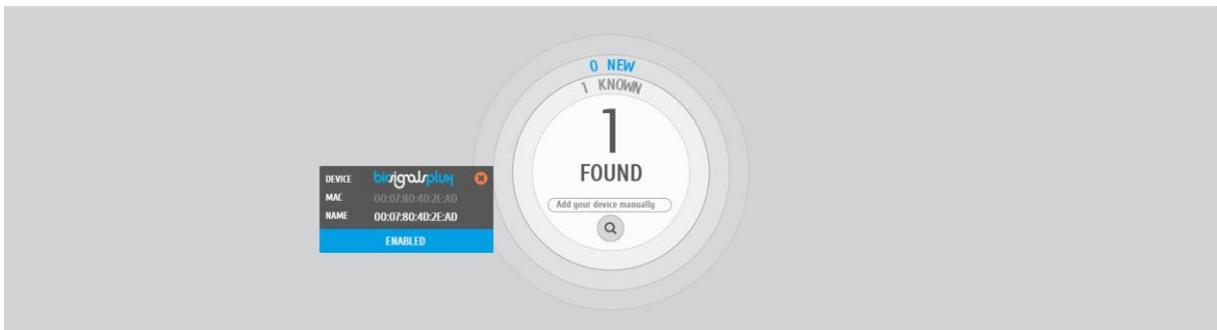


Figure 40: Paired PLUX devices.

2.2.3 Device States

Newly found devices are displayed entirely in blue in the device manager (here: *biosignalsplux*). Additionally, the number of all new and known devices are displayed in the center of the device manager (here: 1 new and 0 known).



Figure 41: One newly found biosignalsplux device.

PLUX devices can be displayed in the device manager without having any effect or being used in acquisitions. This happens when devices are turned on and available, but not activated nor configured for signal acquisitions. These devices are displayed with a gray *DISABLED* field in the device manager. *OpenSignals (r)evolution* will only acquire signals from active devices and ignore communication with deactivated devices.

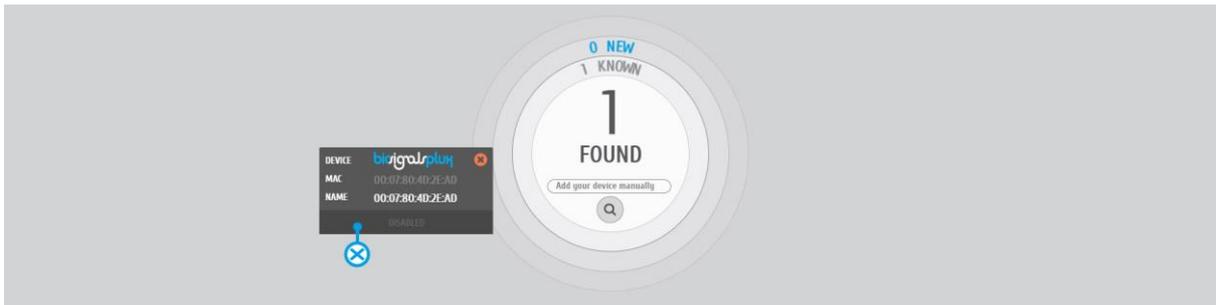


Figure 42: Deactivated biosignalsplux device.

Active devices will be used during the next acquisition with *OpenSignals (r)evolution*. To select and activate a device for signal acquisitions, click on the gray *DISABLED* field on the device. The *DISABLED* field will turn blue and the text should change to *ENABLED*, indicating that the selected device is both available and configured for signal acquisition.

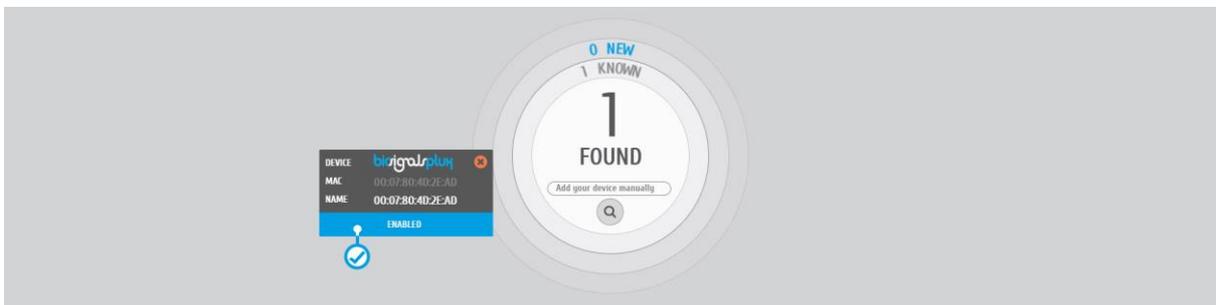


Figure 43: Activated biosignalsplux device.

2.2.4 Device Configuration

Click on the device you would like to set up in the *OpenSignals (r)evolution* device manager to open the device's configuration window.

NOTE

Please check section 5 *Advanced Device Configuration* for detailed information about the specific configuration options of your PLUX device.

2.2.4.1 Device Panel

The device panel displays the device's logo, MAC-address and name (which can be user-defined; see Figure 44).

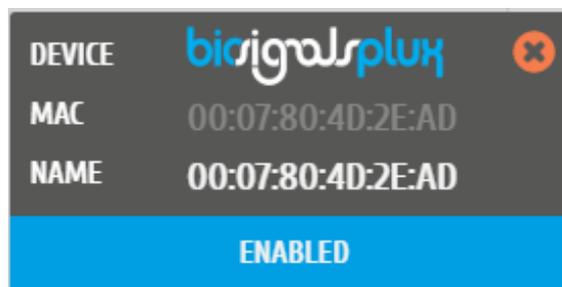


Figure 44: Available device in the *OpenSignals (r)evolution* device manager.

Click on the device panel of the device you would like to set up for signal acquisition. The device panel will expand and show all the available parameters that can be configured for acquisition.

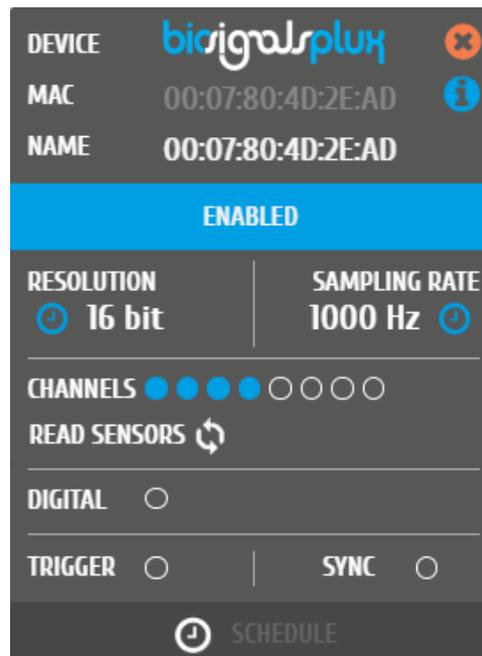


Figure 45: Extended device panel.

2.2.4.2 Device Name

To modify the device name, hover over the *NAME* field in the device panel and replace the MAC-address of the device which is displayed as the default name of your device. Note, that the new name will only take effect inside *OpenSignals (r)evolution* but will not change the device's Bluetooth name in your operating system.



Figure 46: Changing device name.

2.2.4.3 Enabled/Disabled

Click on *ENABLED/DISABLED* to activate or deactivate your device for your next signal acquisition. If the *ENABLED* field is displayed blue, the device is activated for acquisition. If this field message change to *DISABLED* and is displayed gray, the device is deactivated for acquisition (also see 2.2.3 *Device States* for more information).



Figure 47: Left: deactivated biosignalsplx; right: activated biosignalsplx.

2.2.4.4 Sampling Resolution

The sampling resolution sets the number of bits per sample for each channel. Click on the device panel to see all the available configuration options of your device (if not done before) and hover over the *RESOLUTION* field to see the available sampling resolutions for your device (see Figure 48; **here**: 8-bit and 16-bit).

For more detailed information about the available sampling resolution options of your device check the 5 *Advanced Device Configuration* section for the device under analysis.



Figure 48: Available sampling resolutions.

2.2.4.5 Sampling Rate

The sampling rate sets the number of samples that are acquired per second per each channel, i.e. a sampling rate of 1000Hz will acquire 1000 samples per second per channel. Click on the device panel to see all the available configuration options of your device (if not done before) and hover over the *SAMPLING RATE* field so see the available sampling resolutions for your device (see Figure 49; here: 50, 100, 200, 300, ... 4000Hz).

If using *OpenSignals (r)evolution* versions from December 2017 (and newer) and *biosignalsplux* devices with firmware version 3.5, it is even possible to select higher sampling rates of up to 4000Hz. However, note that high *SAMPLING RATES* for all channels are only supported for sampling rates of up to 3000Hz, while 4000Hz is limited to a maximum of 3 channels only. Sampling frequencies of 10Hz, 20Hz and 50Hz are supported with *OpenSignals (r)evolution* versions from June 2018 (and newer).

For more detailed information about the available sampling rate options of your device check the *5 Advanced Device Configuration* section of your device.

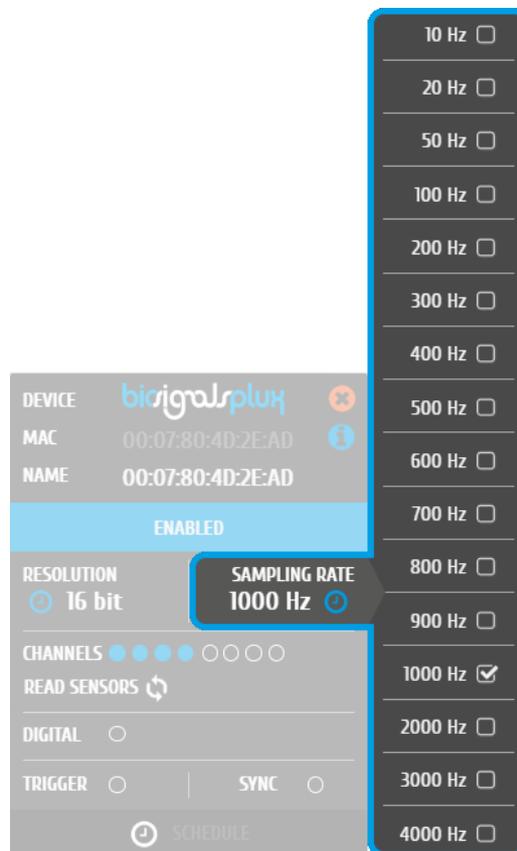


Figure 49: Available sampling rates.

NOTE

Higher resolutions & sampling rates lead to higher quality signal acquisitions but are linked to higher consumption of system resources & higher probability of Bluetooth transmissions issues which may lead to signal delays in the real-time acquisition mode.

We recommend the usage of a PLUX proven [Bluetooth dongle](#) to prevent Bluetooth connection related acquisition issues or to reduce the sampling resolution (or sampling rate) to prevent such issues whenever it is possible.

2.2.4.6 Channel Configuration

The channel configuration allows you to activate and deactivate channels for your signal acquisition session and to configure which sensors are connected to the single channels of your device.

a) Hover over the *CHANNELS* field to open the list of available channels, **b)** activate or deactivate channels by clicking on the circles which are placed on the *CHANNELS* field or next to each single channel. Active channels are marked with blue circles, while inactive channels are marked by empty circles (e.g. in Figure 50 channels 1-4 are active, while channels 5-8 are not). Note, that if you have no sensor specific type selected for an active channel, a warning sign will appear on the respective channel (seen in the image below).

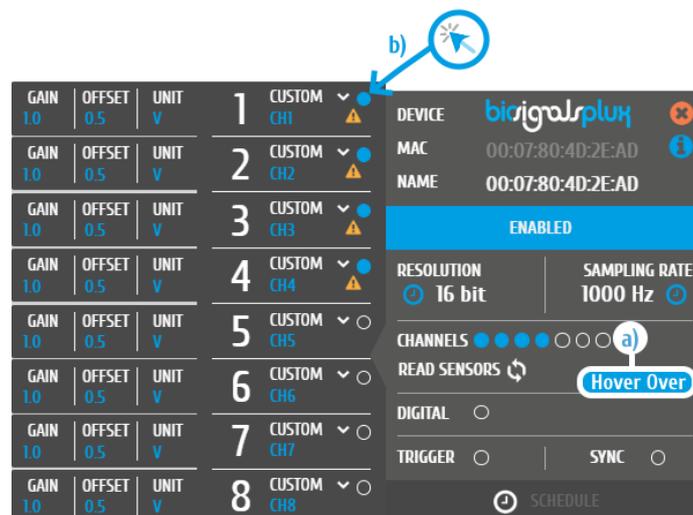


Figure 50: Channel configuration.

Note, that if you have activated no sensor at all, a warning sign will appear next to CHANNELS as seen in the figure below.

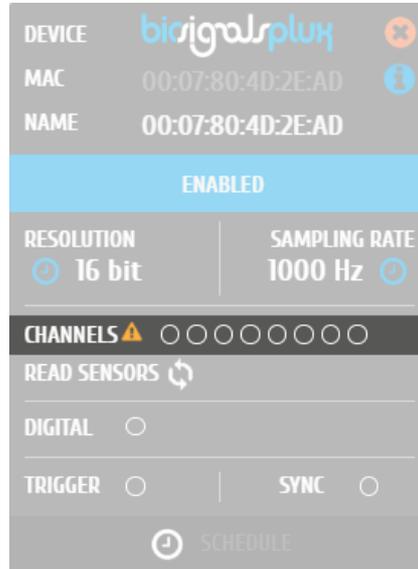


Figure 51: Device panel with no active channel.

To select a pre-set sensor configuration for a channel, hover over the channel that you want to configure and click on the arrow next to the channel. Select the sensor type that fits to the sensor that is connected to your channel (see Figure 52).

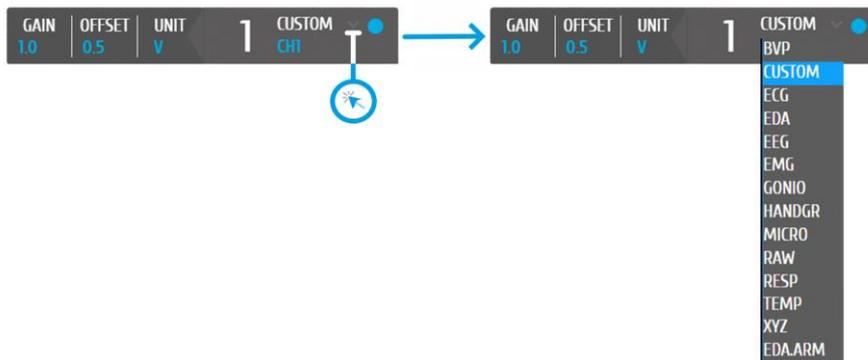


Figure 52: Channel type selection.

A possible channel configuration can be similar to the configuration displayed in Figure 53 with an 8-channel *biosignalslux* device.

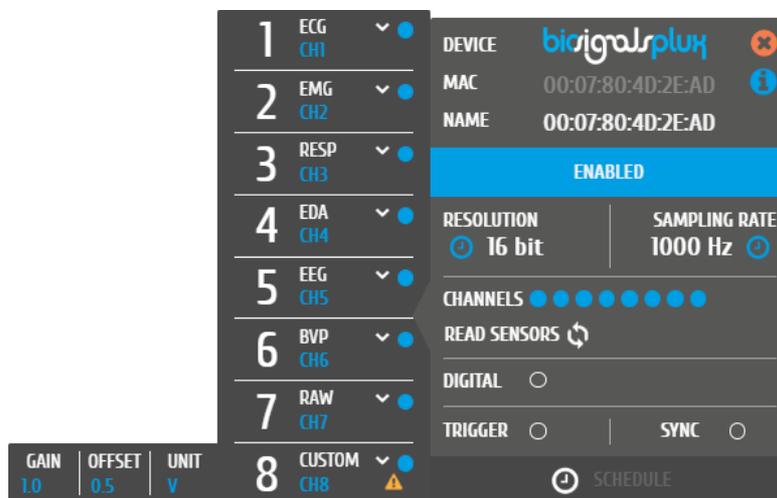


Figure 53: Possible channel configuration.

This configuration will convert the digital data, which is received from the device into the signals' original physical units. The action of choosing a sensor specific channel type ensures that the signal will be displayed in its physical units in the real-time acquisition mode and in the visualization mode, however, the files storing the acquisition signals will only store the digital values (raw data). The visualization in physical units will be restored when reloading *OpenSignals (r)evolution* files.

If a channel is activated, but no selection of sensor type has been made, the warning sign will be displayed on the selected channel as seen in Figure 50 (only in *OpenSignals (r)evolution* versions of June 2018 and newer).

Note that the **READ SENSORS** option, seen in the previous figures, will only be available in **biosignalsplux** devices with firmware version higher than **v3.7**.

This option is extremely useful, providing an intuitive and automatic way to configure the device, i.e., after clicking on **READ SENSORS** button, *OpenSignals* will check which channels of the device have a sensor plugged to it and, additionally, the sensor label (type of sensor) is also identified.

The device should be turned on in order to apply the previous automatism.

As a consequence of this new automatic configuration functionality, while starting a real-time acquisition, *OpenSignals* checks if your device is correctly configured (i.e., if the sensors configured in *OpenSignals* match the sensors connected to the device hub). In case there's a mismatch, a warning message will be raised, recommending the correction.

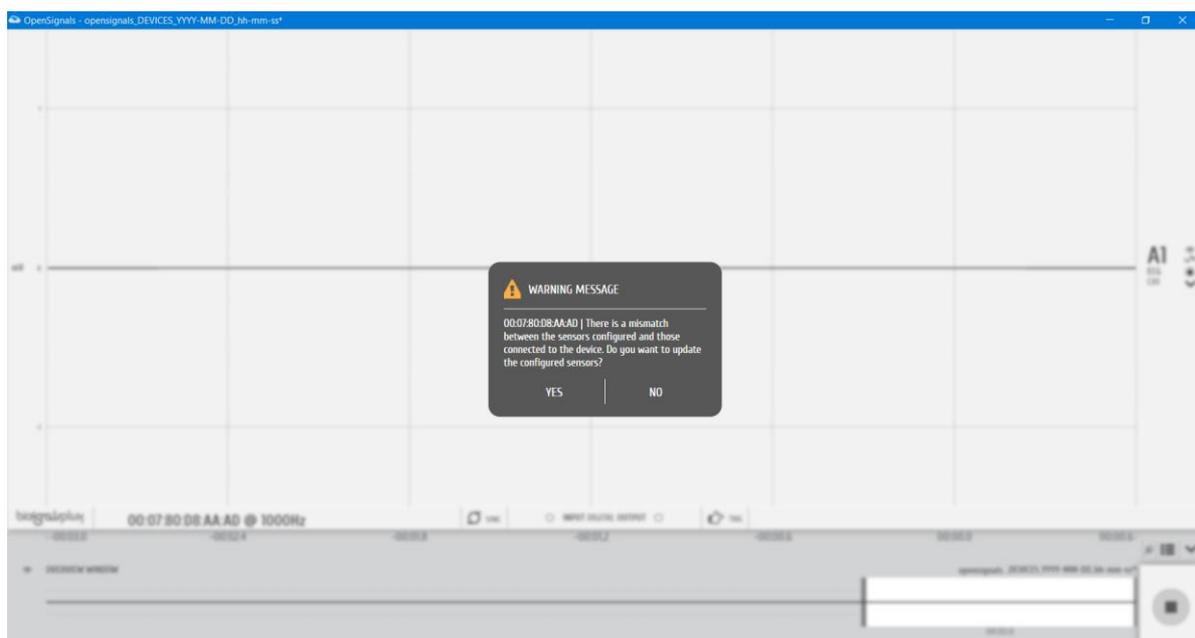


Figure 54: Warning message raised when an incorrect configuration occurs.

Clicking in **YES** will cause the update of the sensor configuration (solving the identified mismatch) and the start of the real-time acquisition. On the other hand, selecting **NO** option generates the start of a real-time acquisition with the unmodified configurations.

NOTE

Some devices allow you to connect different sensor combinations and can be configured according to your individual set up (e.g. *biosignalsplux*). Other devices have built-in sensors which are already pre-connected to the internal device channels. These devices have to be set up correctly according to the internal channel connections for the signals to be displayed correctly during acquisitions.

For more detailed information about the correct channel connections of your device and for a full list of all available and compatible sensor types please check the *5 Advanced Device Configuration* section.

OpenSignals (r)evolution allows you to manually change the label of your channel, which can be useful when using multiple sensors of the same type, for example when using several EMG sensors applied on several muscles of the same muscle group. The channel label will be displayed during the real-time acquisition mode and can be helpful to distinguish the different channels.

Hover over the channel label that is located below the channel type field until a blue pen symbol appears. Click on it and enter the new channel label to change the channel label of the selected channel (see Figure 55).



Figure 55: Changing channel label.

2.2.4.7 Digital Output

Some accessories and components can be connected to the digital I/O port of the devices and controlled within the *OpenSignals (r)evolution* software. Such components are for example the *biosignalsplux* LED or the buzzer on the *BITalino (r)evolution* board.

Digital outputs are deactivated by default. Click on the circles on the *DIGITAL* field of the device panel to activate the digital output and to turn on components or accessories that are connected to the digital outputs at the start of the signal acquisition. Active digital outputs are marked by blue circles, while inactive outputs are marked by empty circles. The outputs can be turned off and on again as needed during the real-time acquisition mode.



Figure 56: Digital channel configuration.

Note, that *biosignalsplux* does only have a single digital output port to control any accessory that is connected to this port, while *BITalino (r)evolution* devices has two digital outputs pins with the first being connected to the on-board LED and the second being connected to the buzzer. However, additional digital ports can be available when using the [multi-sync splitter](#) for *biosignalsplux* devices.

Please check section 3.1.2.4 *Digital Inputs & Outputs* to learn how to control the digital outputs (and inputs) during the real-time acquisition mode.

2.2.4.8 Trigger Options

Trigger accessories allow you to manually start acquisitions using the *biosignalsplux pushbutton*, the [footswitch](#) or the *BITalino (r)evolution pushbutton* instead of automatically starting acquisitions when entering the real-time acquisition mode. *OpenSignals (r)evolution's* trigger option is deactivated by default.

To use this feature, make sure your trigger accessory is connected to your device⁴ and activate the trigger option in the *TRIGGER* field of the device panel (see Figure 57).



Figure 57: Trigger configuration.

⁴ biosignalsplux accessories have to be connected to the digital port of your device.

If the acquisition start is set to wait for a manual trigger, the circle will be marked blue. Otherwise, the circle will remain empty. Open the real-time acquisition mode and press your trigger whenever you are ready to start your acquisition.

2.2.4.9 Synchronizing Devices

8-Channel *biosignalsplex* devices offer the possibility of synchronized acquisitions by connecting up to 3 *biosignalsplex* hubs using the synchronization kit. Activating the SYNC function will synchronize the acquisition between all devices and prevent signal drifts within different devices.

Please check section 3.4 *Synchronization* for more information on how to use this feature.

2.2.4.10 Removing Devices

To remove devices from the *OpenSignals (r)evolution* device manager, click on the 'x' inside the orange circle (top right corner of the device panel) to remove the device from the device manager.

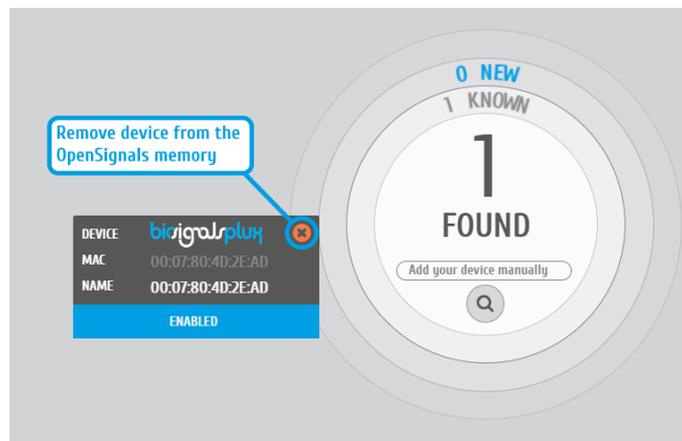


Figure 58: Remove device by clicking on the 'x' in the orange circle.

2.2.4.11 Device Info

Since version **2.2.0**, *OpenSignals (r)evolution* provides more detailed information about each device. This information can be accessed by the user after clicking on the *Blue Info* button available in the device panel.

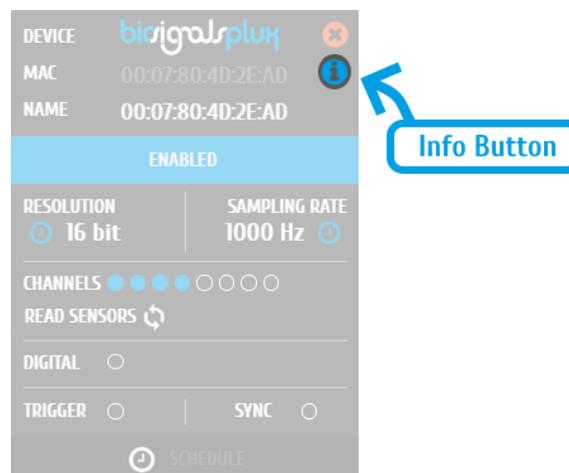


Figure 59: Device info button.

Once the *Blue Info* button is pressed, different fields may appear in accordance with the type of device under analysis.

All available fields can be observed in the *biosignalsplux* example, that will be presented below.

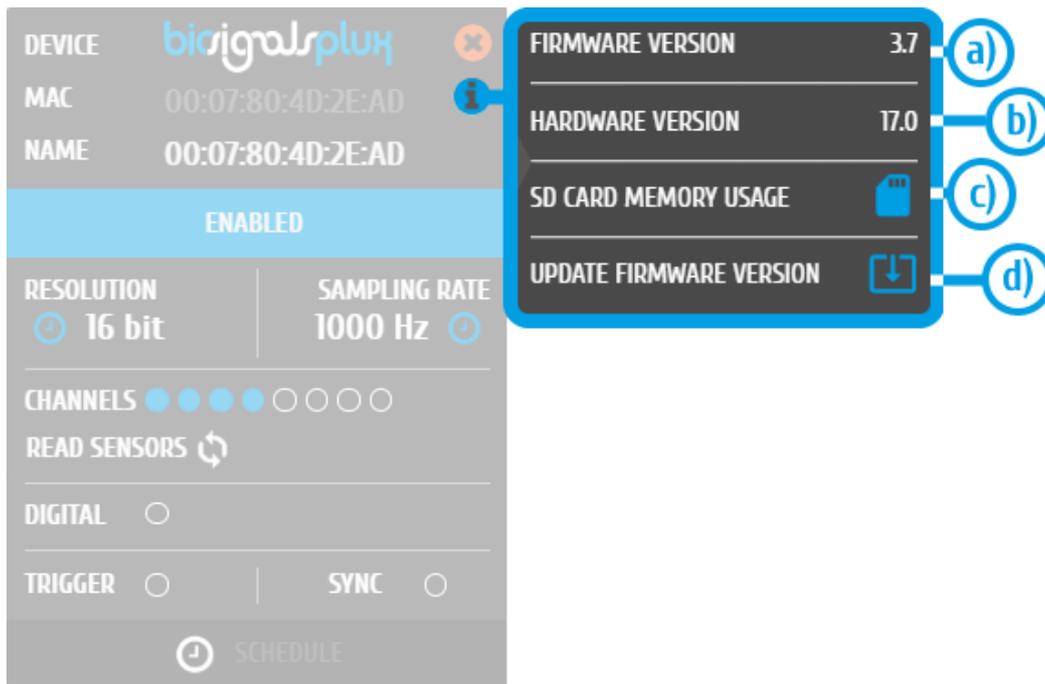


Figure 60: Device info content/fields.

As can be perceived, there are two “read-only” fields (available in the great majority of PLUX devices):

- a) **Firmware Version** | identifies the version of the firmware that is installed in the device, i.e., the low-level software that controls and interconnect the device components (being upgradable or downgradable);
- b) **Hardware Version** | shows the production version of the physical device (immutable property).

Regarding the two “interactable” fields, they are only available on certain types of devices, such as the *biosignalsplux*.

The first one, labelled **c**) in Figure 60, is the *SD CARD MEMORY USAGE* being intended to evaluate the percentage of memory used, i.e., in devices with an internal memory card how much space is being occupied by the acquisition files.

If the device under analysis supports the internal memory feature, then, a message similar to the one presented below should be raised.



Figure 61: SD card memory usage statistics.

On the other hand, when the device under analysis doesn't contain an internal memory card, then, an informative message is presented to the user in order to provide some feedback about this limitation.

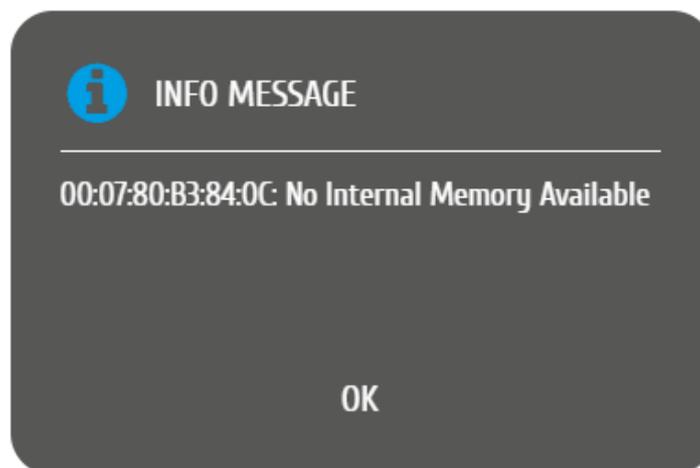


Figure 62: Message informing that the SD card memory is not available in the current device.

Finally, the *UPDATE FIRMWARE VERSION* button provides a way to the user update/downgrade the firmware of his device through a semi-automatic workflow that continuously presents feedback messages informing about the operations that the software is conducting, and which actions are needed by the user.

The first step of the workflow includes the selection of the firmware version that the user wants to install in the device:

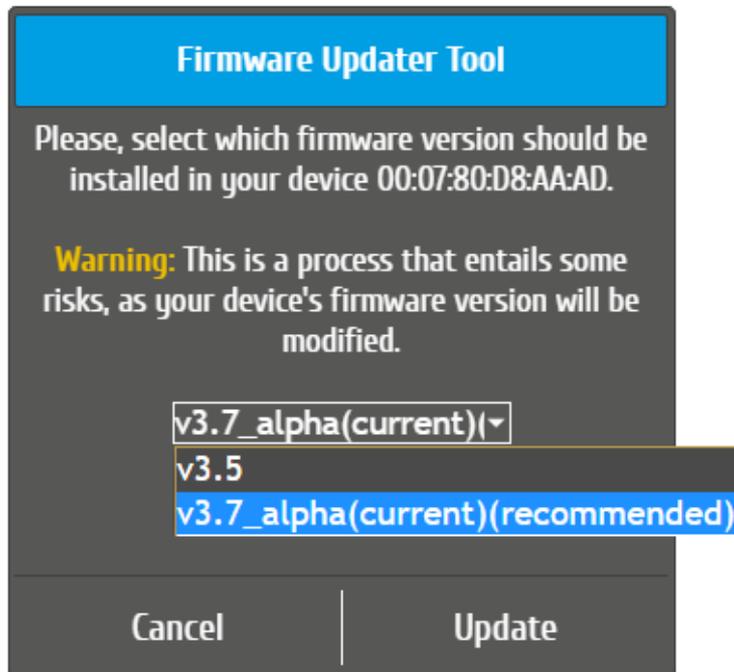


Figure 63: Firmware version selection dialog.

Next, when the user clicks in the *UPDATE* button, the firmware upgrade/downgrade workflow will start, being formed by following stages:

1. Message requesting the user to *Turn On* the device (10 seconds);

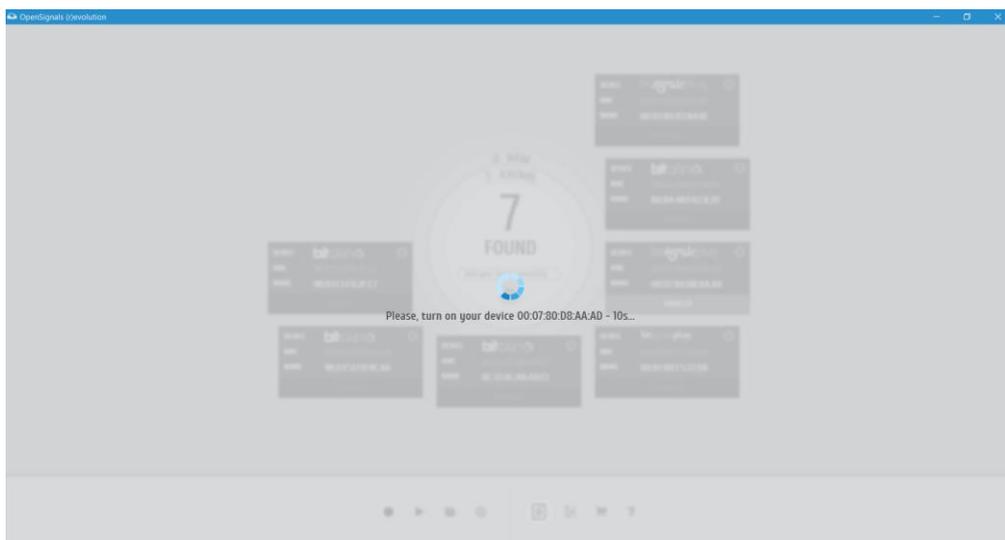


Figure 64: Requesting the user to turn on the device.

2. Connection try;



Figure 65: Connecting to the device for the first time.

a. Show error message **if** the battery level is smaller than **50%**.

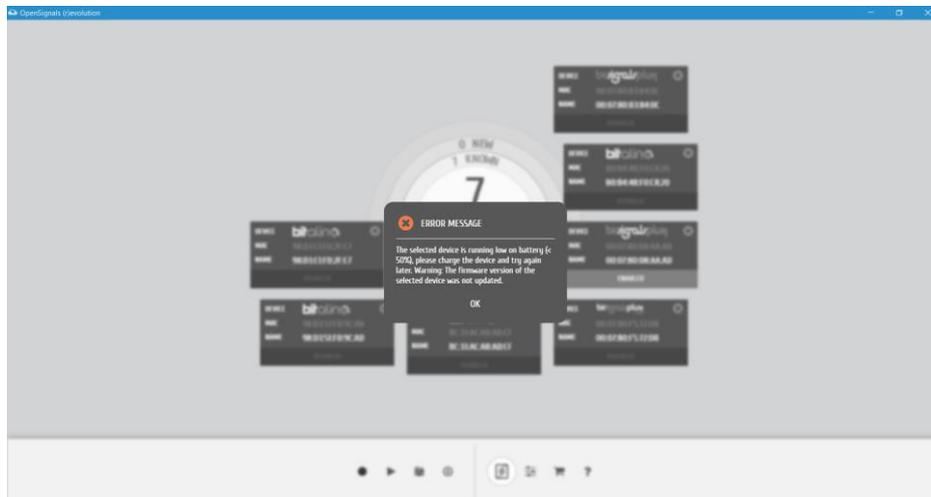


Figure 66: Error dialog informing about the low battery level of the device.

3. Assessment to the *Bootloader Mode* (7 seconds);

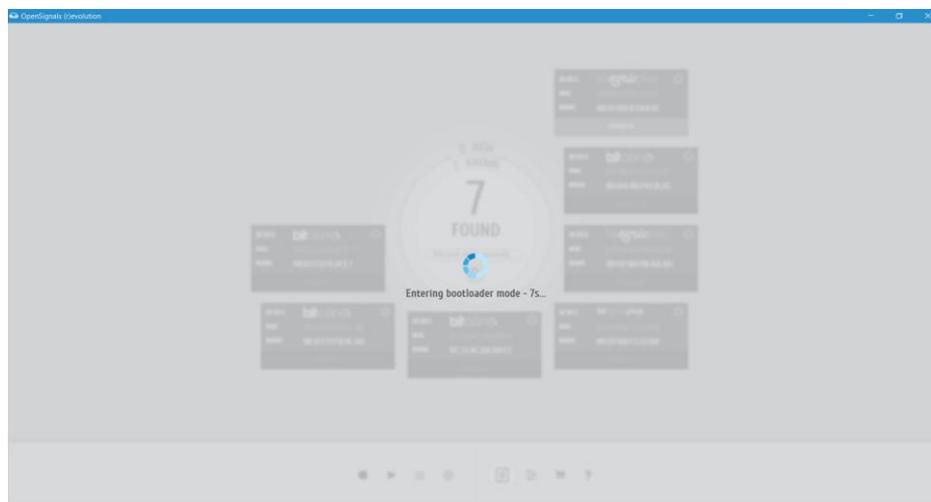


Figure 67: Bootloader mode waiting screen.

4. Reconnection to the device;

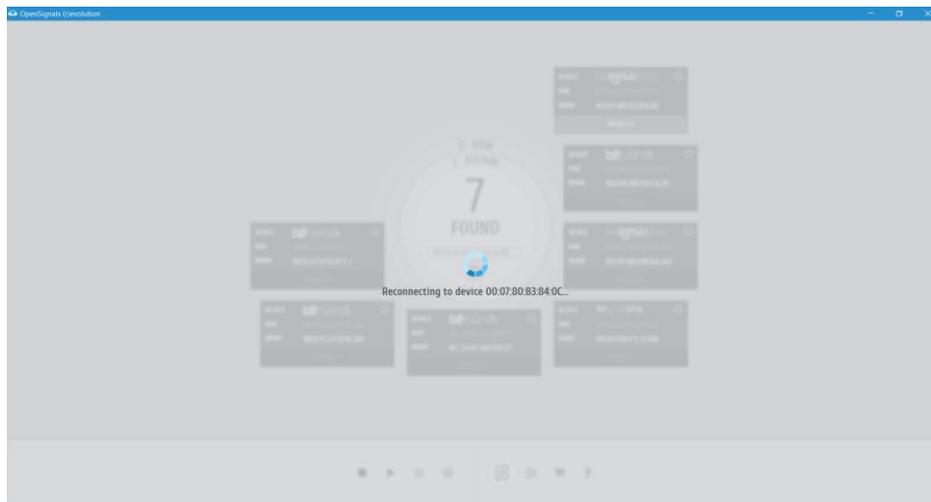


Figure 68: Reconnection to the device before sending the new firmware version.

5. Sending firmware to the device;



Figure 69: Informing the user that the new firmware version is being sent to the device.

6. Installation of the new firmware (15 seconds);

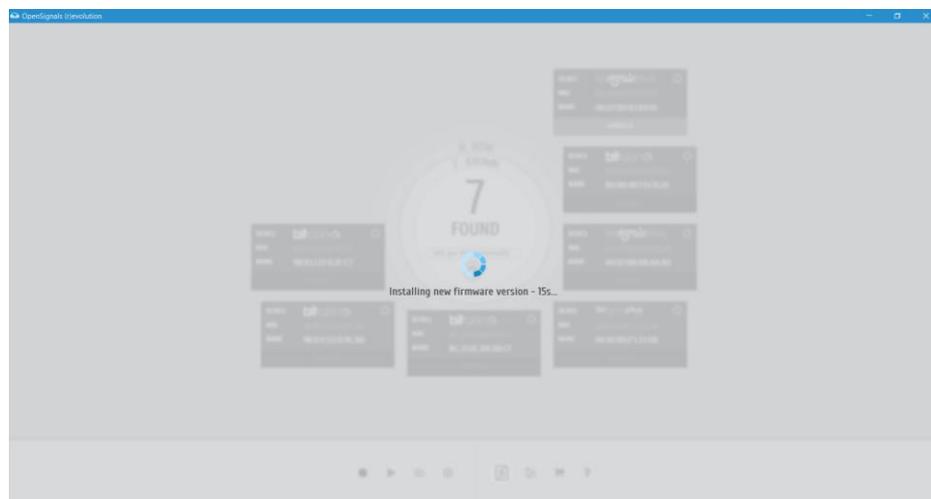


Figure 70: Providing some feedback about the installation of the new firmware.

7. Request the user to turn the device on again (the only non-automatic step that requires the intervention of the user that has a 10 seconds duration);



Figure 71: Request user to turn on the device after the firmware update.

8. Reconnection to the device;
 - a. Present a dialog informing the user that the reconnection was not completed with success and requesting to turn on the device before trying to reconnect for a second time.

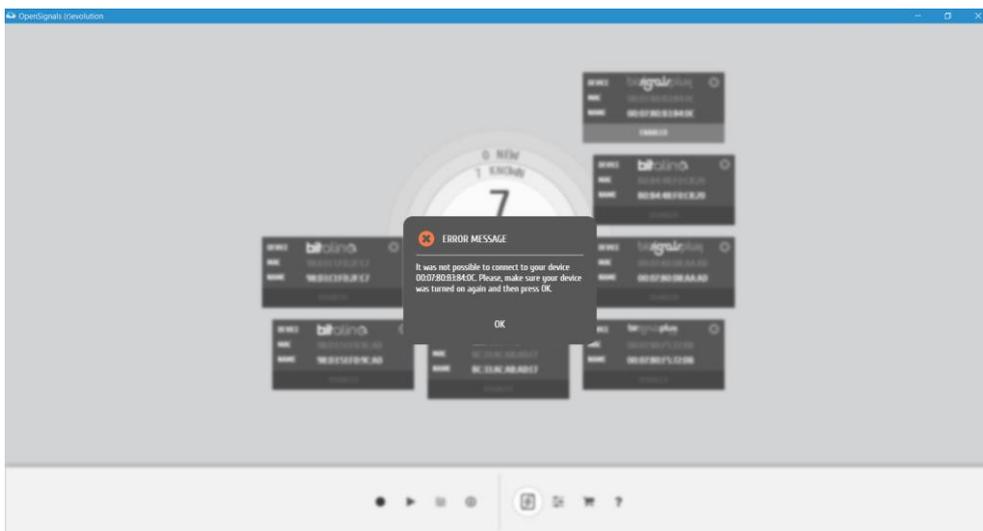


Figure 72: Inform the user about the difficulties in reconnecting to the device.

9. Check if the update/downgrade was successfully completed;
 - a. Inform the user about the success/failure of the firmware update workflow.

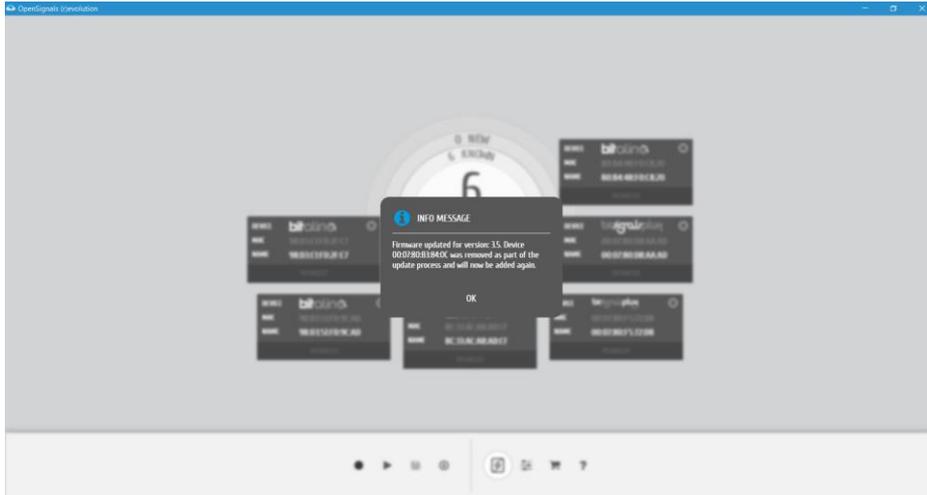


Figure 73: Firmware update success dialog.

10. Remove device from the list;
11. Re-add the device with the new configuration.

2.3 General Acquisition Settings



save location, file prefix, file format and auto-save options

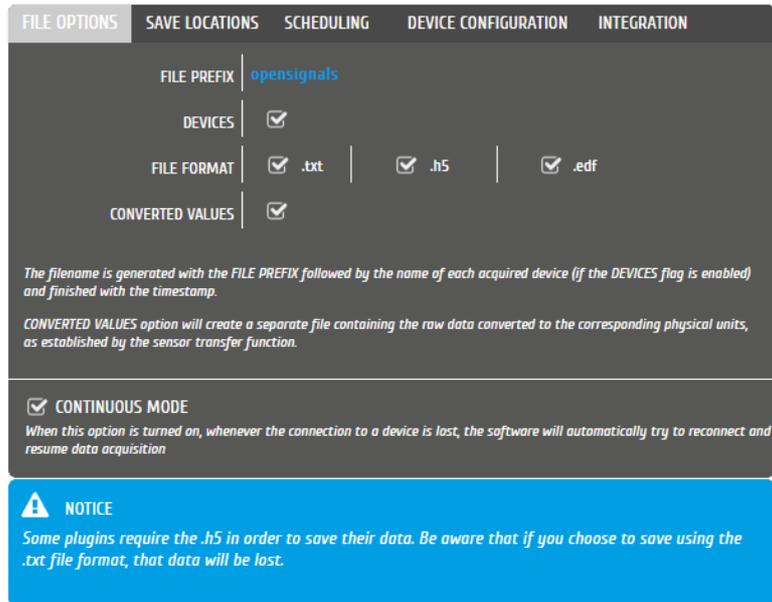


Figure 74: File options settings window.

2.3.1 Continuous Mode

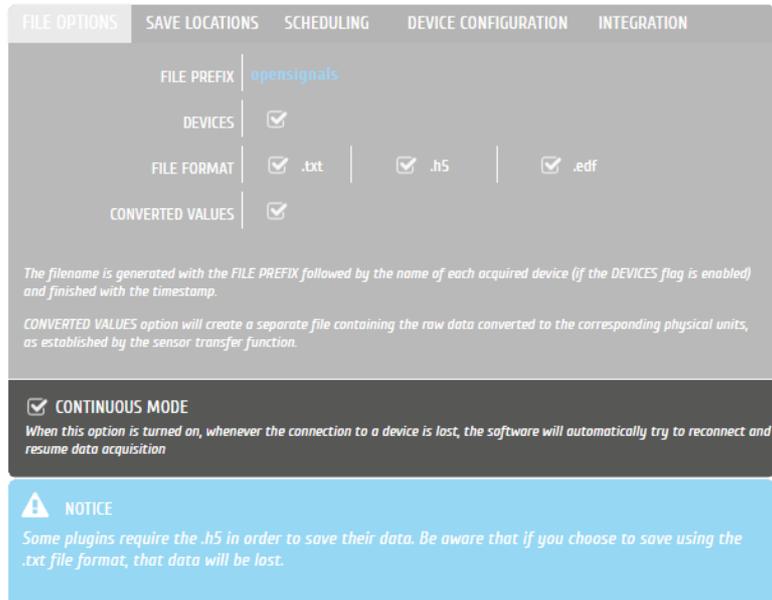


Figure 75: Highlighting Continuous Mode option.

The continuous mode will allow you a continuous acquisition until the acquisition is manually stopped. Activating this mode will try to continue your acquisition even if your device has lost connection to your computer by reconnecting when the device is available and within a reachable distance. The signal acquisition will then be restarted.

If this function is deactivated, the acquisition will be stopped after a communication break up.

2.3.2 File Prefix



Figure 76: Highlighting File Prefix option.

The file prefix is a user-defined text that will be added to the acquisition's file name. An additional timestamp is automatically set after this prefix (example here: *opensignals_devicenames_YYYY-MM-DD_hh-mm-ss.format_extension*).

2.3.3 Devices

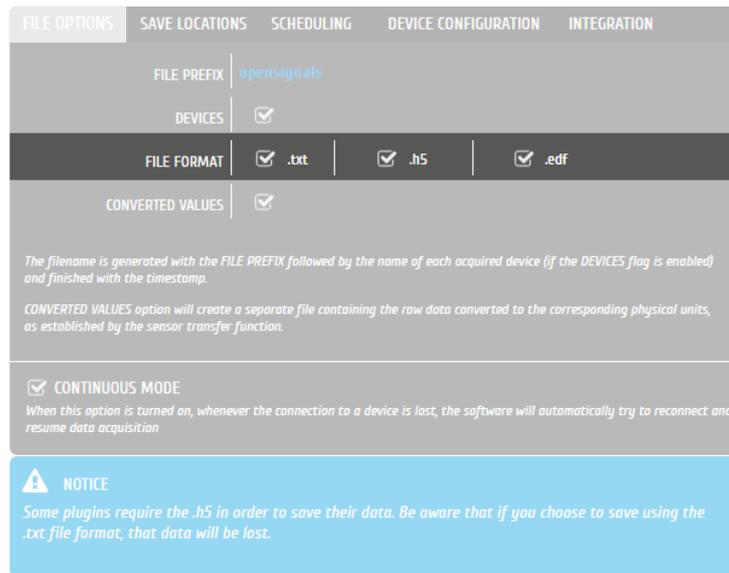


Figure 77: Highlighting Devices option.

When this option is selected, the name of the device being used will be automatically added to the acquisition's file name. This is useful for example when recording data from multiple devices simultaneously, to easily distinguish the files generated by each device.

If you have not changed the name of your device (see 2.2.4.2 *Device Name*), the default device's MAC-address will be used instead.

2.3.4 File Formats & Specifications



FILE OPTIONS | SAVE LOCATIONS | SCHEDULING | DEVICE CONFIGURATION | INTEGRATION

FILE PREFIX: opensignals

DEVICES:

FILE FORMAT: .txt | .h5 | .edf

CONVERTED VALUES:

The filename is generated with the FILE PREFIX followed by the name of each acquired device (if the DEVICES flag is enabled) and finished with the timestamp.

CONVERTED VALUES option will create a separate file containing the raw data converted to the corresponding physical units, as established by the sensor transfer function.

CONTINUOUS MODE
When this option is turned on, whenever the connection to a device is lost, the software will automatically try to reconnect and resume data acquisition

NOTICE
Some plugins require the .h5 in order to save their data. Be aware that if you choose to save using the .txt file format, that data will be lost.

Figure 78: Highlighting File Format options.

OpenSignals (r)evolution allows you to save the acquired sensor data into .TXT, .H5 or .EDF files. TXT files can be used to export signal data to 3rd party software. When opening TXT files, *OpenSignals (r)evolution* will convert your file into an H5 file as this file is more suitable to store additional metadata which is created when using *OpenSignals (r)evolution*.

The European Data Format (EDF) is a simple and flexible format for exchange and storage of multichannel biological and physical signals. Visit <https://www.edfplus.info/> for more information.

NOTE

The TXT files do only contain the raw signal data from all sensors and digital I/O which were used during the acquisition. These files do not contain any processed data or results from *OpenSignals (r)evolution* add-ons (see section 4 *Add-Ons*). If you need to export the results of these add-ons, please follow the instructions in section 4.3 *Add-on Reports*).

2.3.5 Converted Values

OpenSignals (r)evolution stores, by default, only the raw digital sensor values in the acquisition files. Importing this acquired sensor data into 3rd party software usually requires a conversion of the raw digital sensor data into the original physical units, which can be done by using the sensor-specific transfer functions found in the sensor datasheets (see section 5.14.1 *biosignalsflux* and 5.14.2 *BITalino* for links to all sensor datasheets).

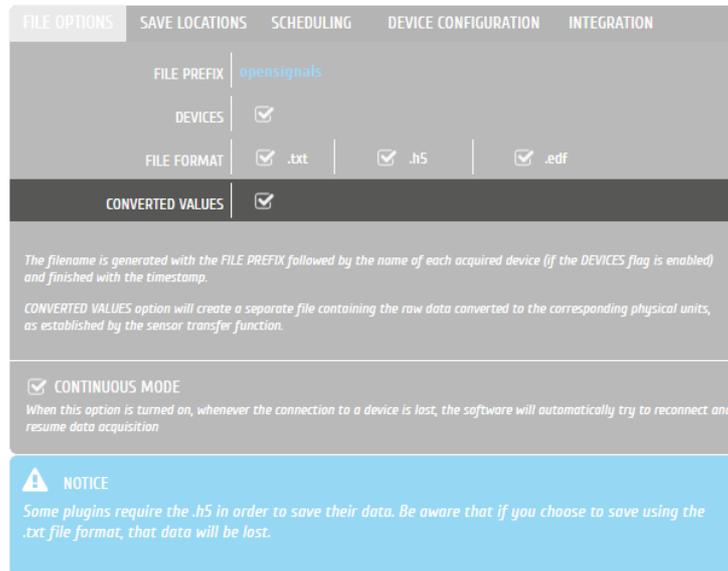


Figure 79: Highlighting Converted Values option.

If the converted values option is activated, *OpenSignals (r)evolution* will store the converted values in additional acquisition files.

Note that raw digital sensor data can be distinguished from converted data, as data provided in raw format ranges between 0 and $(2^n - 1)$, with n being the configured resolution for the signal acquisition (e.g. between 0 and 65 535 $(= 2^{16} - 1)$ when using 16-bit resolution), while converted sensor values lie within much smaller ranges (e.g. between -1.5 and 1.5 for ECG or EMG sensor data).

2.3.6 Save Locations

Sets the path where the acquired data and files will be saved.

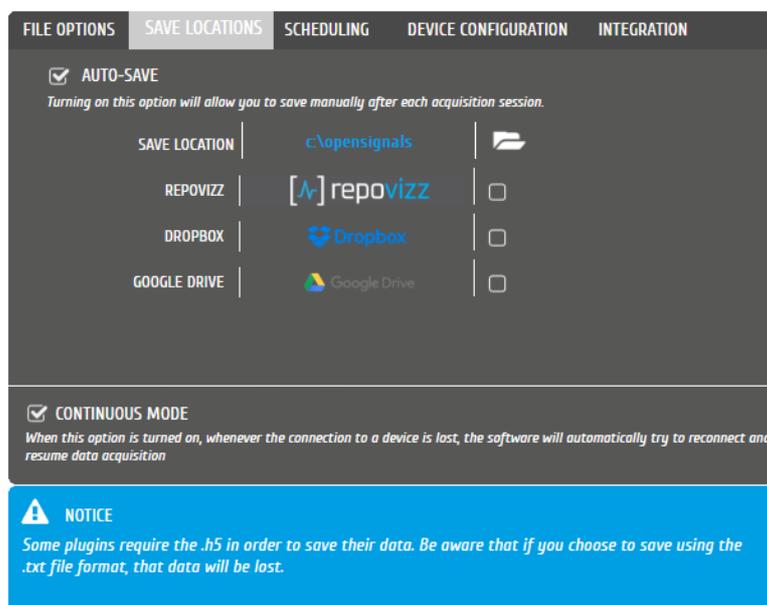


Figure 80: Save Locations settings window.

2.3.6.1 Local

To store files on your computer, click on the folder icon at *SAVE LOCATION* and select a path where you want to store your acquisition files. *OpenSignals (r)evolution* files will then be stored into the selected path.

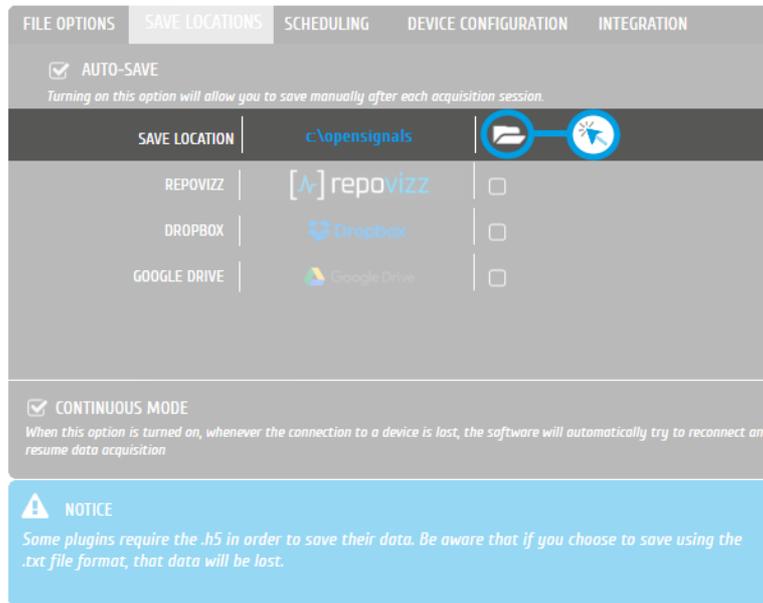


Figure 81: File options local Save Location button.

2.3.6.2 RepoVizz

RepoVizz is an integrated online system capable of structural formatting and remote storage, browsing, exchange, annotation, and visualization of synchronous multi-modal, time-aligned data. This online platform allows you to store your *OpenSignals (r)evolution* files online and to easily share them if needed.

To use *RepoVizz*, activate *RepoVizz* in the *SAVE LOCATIONS* window (see Figure 80) and log in into your *RepoVizz* account using your username and the API key which has been generated for your account during the registration process (see Figure 82).

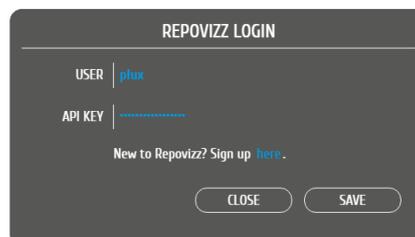


Figure 82: RepoVizz login window.

If you do not have a *RepoVizz* account yet, select *Sign up here* in the login window to access the *RepoVizz* homepage or follow this link:

<https://repovizz.upf.edu/repo/Home>

Note that selecting *RepoVizz* as save location will not have any influence on other selected save locations (local, *Dropbox* or *Google Drive*).

2.3.6.3 Dropbox

To use *Dropbox*, activate Dropbox in the *SAVE LOCATIONS* window (see Figure 80). A new browser window will be opened in which you can log in into your Dropbox account (see Figure 83).

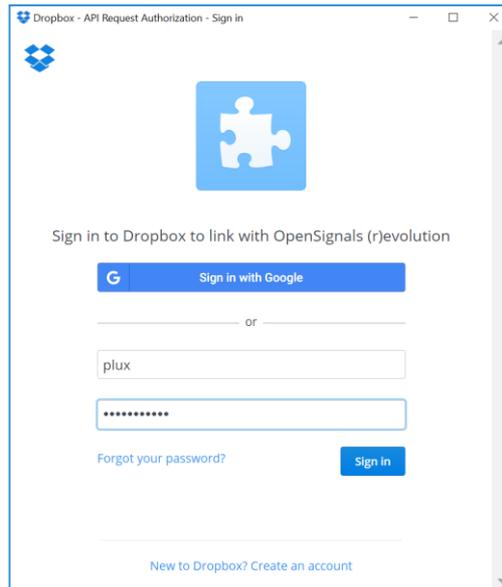


Figure 83: Dropbox login window.

Enter your *Dropbox* account information into the required fields and click on *Sign In* to proceed. *Dropbox* will require your confirmation to allow *OpenSignals (r)evolution* to access your *Dropbox*. Click on *Allow* to proceed. Your files can now be stored into your *Dropbox* account.

If you do not have a *Dropbox* account yet, click on *New to Dropbox? Create an account* in the login window or access the *Dropbox* sign up page using the following link and follow the instructions on the *Dropbox* website: <https://www.dropbox.com/>

Note, that using *Dropbox* as save location will not have any influence on other selected save locations (local, *RepoVizz* or *Google Drive*).

2.3.6.4 Google Drive

To use *Google Drive*, activate *Google Drive* in the *SAVE LOCATIONS* window (see Figure 80). A new browser window will be opened in which you can log in into your *Google Drive* account (see Figure 84).

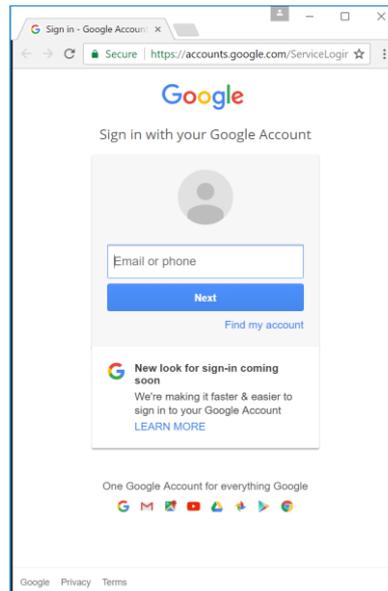


Figure 84: Google Drive login window.

Enter your *Google Drive* account information into the required fields and click on *Next* to proceed, enter your account's password and click on *Sign In*. Your files can now be stored into your *Google Drive* account.

If you do not have a *Google Drive* account yet, sign up using the following link and follow the instructions on the *Google Drive* website:

<https://accounts.google.com/SignUp>

Note, that using *Google Drive* as save location will not have any influence on other selected save locations (local, *RepoVizz* or *Dropbox*).

2.3.7 Auto-save Options

When this option is enabled, *OpenSignals (r)evolution* will automatically save every acquisition session in a new file according to the configured file prefix, format, and other configurations mentioned in the last sections.



Figure 85: File options Auto-Save check-box.

Auto-save is activated by default. Uncheck the box if you prefer to manually save your acquisitions after each recording session.

NOTE

After the previous configuration steps regarding the file “Save Locations” you will be able to load/save data from/to your repositories, through the *OpenSignals* file explorer:

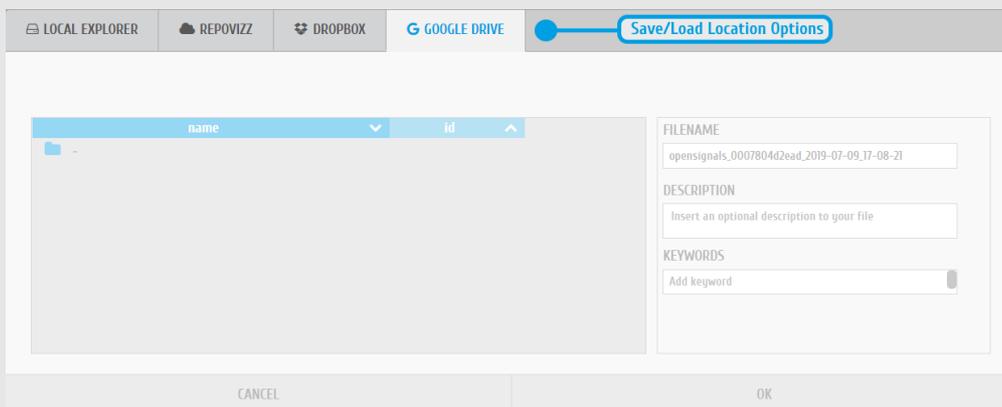
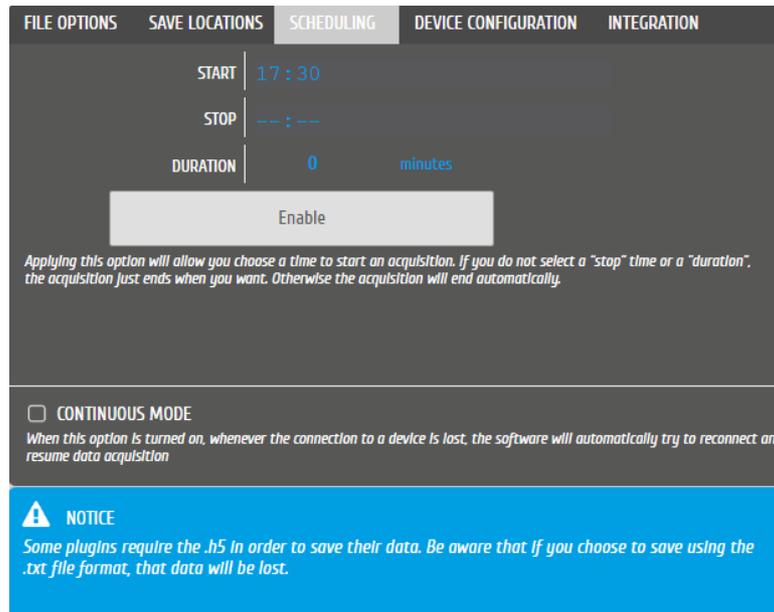


Figure 86: File Explorer – Save Location options.

2.3.8 Online Scheduling



The screenshot shows a software interface with a dark grey background and a blue header bar. The header bar contains five tabs: 'FILE OPTIONS', 'SAVE LOCATIONS', 'SCHEDULING' (which is highlighted), 'DEVICE CONFIGURATION', and 'INTEGRATION'. Below the tabs, there are three input fields: 'START' with the value '17:30', 'STOP' with the value '--:--', and 'DURATION' with the value '0' and the unit 'minutes' in blue text. Below these fields is a large grey button labeled 'Enable'. Underneath the button is a paragraph of text: 'Applying this option will allow you choose a time to start an acquisition. If you do not select a "stop" time or a "duration", the acquisition just ends when you want. Otherwise the acquisition will end automatically.' Below this text is a checkbox labeled 'CONTINUOUS MODE' which is currently unchecked. Underneath the checkbox is another paragraph: 'When this option is turned on, whenever the connection to a device is lost, the software will automatically try to reconnect and resume data acquisition'. At the bottom of the interface is a blue banner with a white warning icon and the text: 'NOTICE Some plugins require the .h5 in order to save their data. Be aware that if you choose to save using the .txt file format, that data will be lost.'

Figure 87: Online scheduling.

This scheduling feature allows to schedule the start and end-time or duration of an acquisition. To configure, set up your device in the device manager as described in section 2.2 *Device Manager*, specify the start and stop times for the scheduled acquisition and then click on *Enable* button.

If no end-time or duration is set for a scheduled acquisition, the acquisition can be stopped manually.

For instructions on how to schedule offline acquisitions with *biosignalsplex* devices with internal memory, please check section 3.3 *Data Logging*.

2.3.9 Replicate Device Configuration



Figure 88: Replicate device configuration.

This feature allows to select a master device from which the configuration will be used as the new standard configuration of newly added devices.

To use this feature, set up your master device in the device manager as described in section 2.2 *Device Manager*. When done, return to the *DEVICE CONFIGURATION* seen in Figure 88 and define the master device by selecting its MAC-address in the dropdown menu (for the graphical example our master device is `00:07:80:4D:2E:AD`) and check the checkbox next to *REPLICATE DEVICE CONFIGURATION* to set it as the master device. Every new device will then inherit the configuration of the selected master devices (channel configuration, sampling rate & resolution).

2.3.10 Integration & Data Streaming – TCP/IP & Lab Streaming Layer

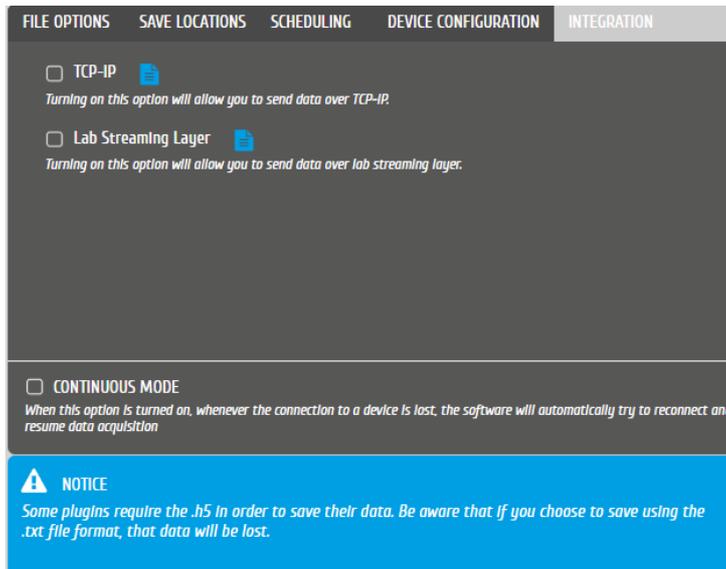


Figure 89: OpenSignals (r)evolution streaming options.

OpenSignals (r)evolution comes with built-in streaming capabilities which allows you to stream all the acquired sensor data in real-time to third party software over your network. The following two streaming options are available:

TCP/IP

The TCP/IP (Transmission Control Protocol / Internet Protocol) module of the *OpenSignals (r)evolution* software is aimed to facilitate the support and data exchange between 3rd party and the *OpenSignals (r)evolution* software. It has been introduced with the release of October 2018.

You can find detailed information on how to use this streaming feature in the following document:

https://www.downloads.plux.info/OpenSignals_TCP_IP_Manual.pdf

Lab Streaming Layer (LSL)

The Lab Streaming Layer (LSL) module of the *OpenSignals (r)evolution* software is, like the previous TCP/IP functionality, aimed to facilitate the support and data exchange between the 3rd party and the *OpenSignals (r)evolution* software. It has been introduced with the *OpenSignals (r)evolution* release of December 2018 and is based on the open-source LSL system which can be found on GitHub:

<https://github.com/sccn/labstreaminglayer>

As found in its official description, “*LSL is a system for the unified collection of measurement time series in research experiments that handles both the networking, time-synchronization, (near) real-time access as well as optionally the centralized collection, viewing and disk recording of the data*”.

This system enables *OpenSignals (r)evolution* to stream multi-channel sensor data acquired using *biosignalsplux* and *BITalino* kits to third 3rd party applications where only few lines of code are required to receive real-time sensor data.

You can download a detailed guide and example scripts demonstrating how to receive signals streamed via *OpenSignals* in Python at the following links:

https://downloads.plux.info/OpenSignals/opensignals_lsl_guide.zip

<https://www.downloads.plux.info/OpenSignals/OpenSignals%20LSL%20Manual.pdf>

[https://www.downloads.plux.info/OpenSignals/misc/user-manual/OpenSignals_Lab_Streaming_Layer_Guide_\(MATLAB\).pdf](https://www.downloads.plux.info/OpenSignals/misc/user-manual/OpenSignals_Lab_Streaming_Layer_Guide_(MATLAB).pdf)

2.4 Save Last Recording

 save data from last real-time acquisition

This graphical element provides a path to manually store the files generated during the last real-time streaming session.

It is an extremely important functionality, particularly when the **AUTO-SAVE** option (section 2.3.7) is not active, avoiding the potential long-term loss of collected data (when data from the *OpenSignals temp* folder is erased).

Clicking in this button will raise a standard **File Explorer** interface. The user should navigate until the destination directory, i.e., the directory where the collected data will be stored.

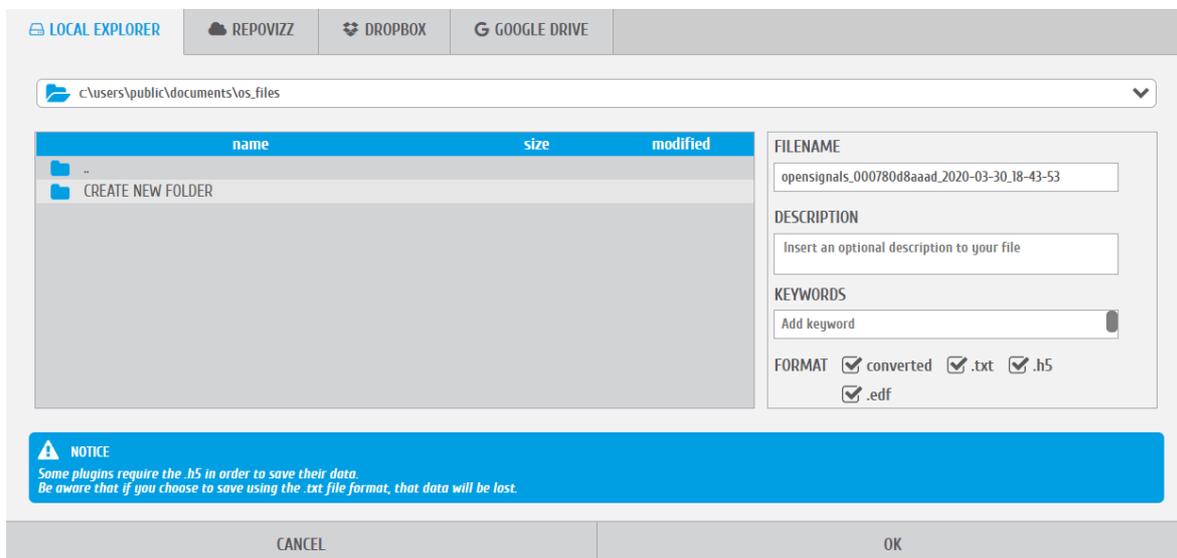


Figure 90: OpenSignals (r)evolution manual save File Explorer.

Accordingly to the active **Save Location** options (section 2.3.6), namely **a) Google Drive; b) Dropbox; c) RepoVizz**, the **File Explorer** interface will have a correspondent number of available tabs:



Figure 91: OpenSignals (r)evolution local and remote save locations tabs.

After selecting the destination directory and the desired local/remote folder, the user has total freedom to edit the **a) Filename**, include a **b) Description** and also some **c) Keywords** (that will be stored in the metadata fields of the files), in addition to the desired **d) File Formats**.

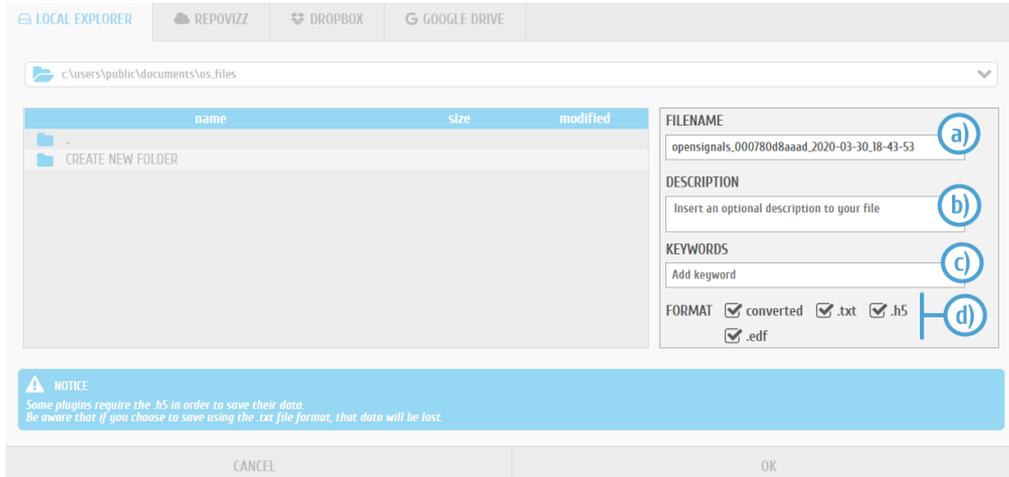


Figure 92: Available customization options in the manual save process

All the previous configurable options will be effective once the **OK** button is clicked:

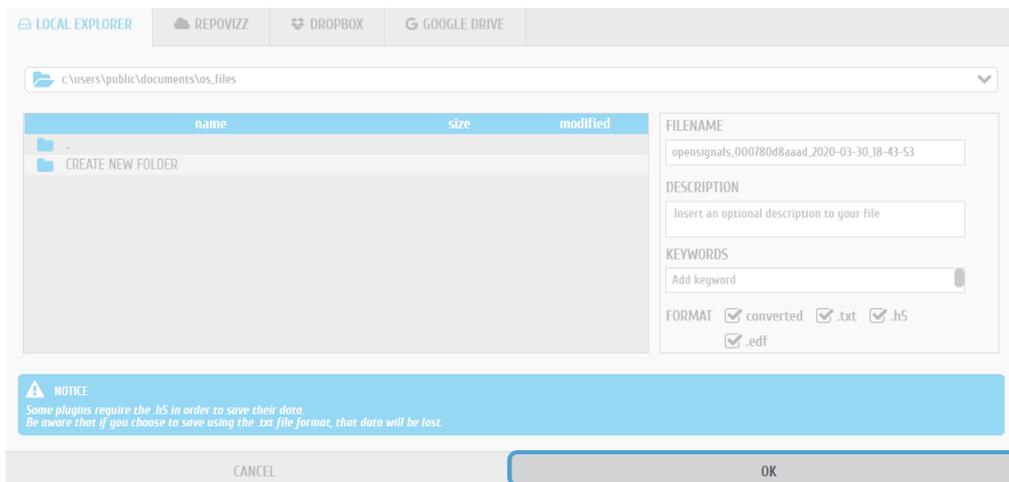


Figure 93: OK button location in the File Explorer

3 Recording & Visualizing Signals

OpenSignals (r)evolution has a live signal acquisition mode which enables real-time signal acquisition using PLUX devices and a signal visualization mode which allows you to visualize acquired signal acquisition sessions.

Both modes share common functions, which can be accessed via the symbols displayed on the following pages. We recommend going through these explanations to take advantage of all *OpenSignals (r)evolution* features.

3.1 Signal Acquisition Mode

NOTE

We recommend configuring your device before entering the signal acquisition mode by following the instructions described in *2.2.4 Device Configuration* as modifications in device configurations cannot be done afterwards.

Recording signals with *OpenSignals (r)evolution* is made in signal acquisition mode (see Figure 94) which can be accessed in the main interface by clicking on the red button.

- Enter signal acquisition mode to record signals.

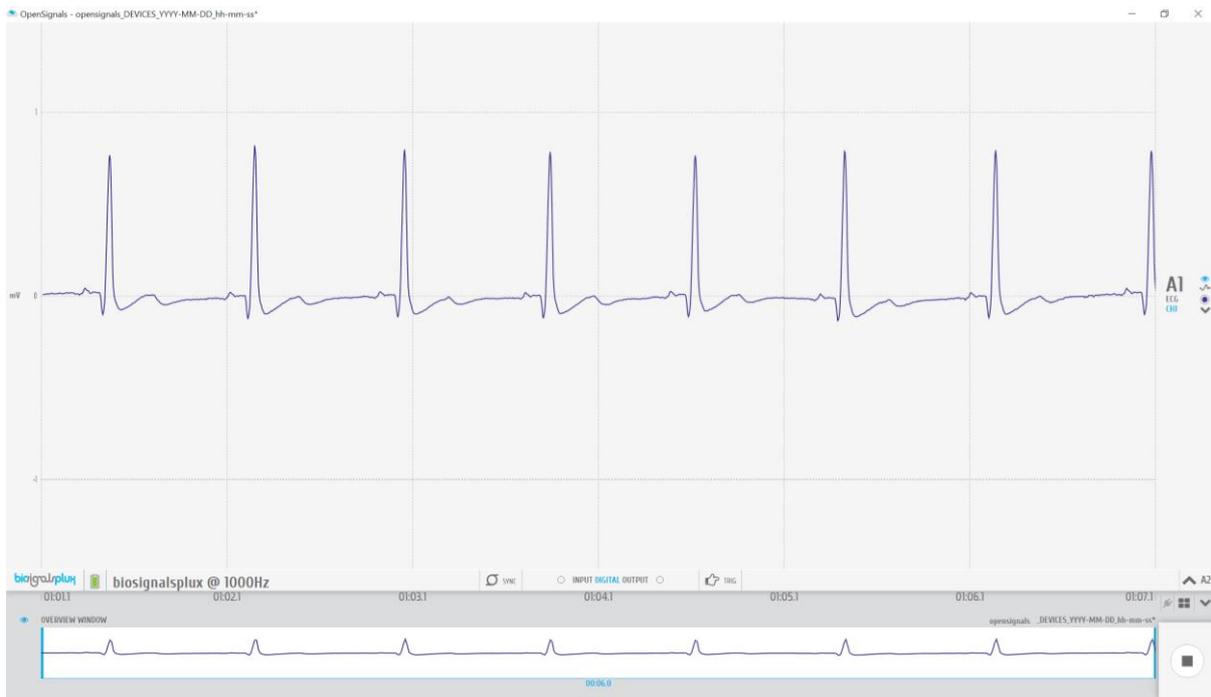


Figure 94: Signal acquisition mode.

When entering in the signal acquisition mode, *OpenSignals (r)evolution* will automatically connect to your device to configure and start the signal acquisition. If your devices is configured to start by a manual trigger, for example using a [trigger button](#) or a [foot switch](#), the recording will only start when the manual trigger has been set.

OpenSignals (r)evolution will automatically issue warning notifications if any error occurs during connection and/or configuration attempts with your device(s). If any of these errors occur, please check if your problem can be solved with the information given in [7 Troubleshooting](#) or contact our support if the problems cannot be solved (see [9.3 Contact & Support](#)).

The visualization mode consists on 3 different interface sections which allow you to control the visualization of the selected files and signals:

- Signal acquisition (1 in Figure 95; see [3.1.1 Signal Acquisition](#))
- Information bar (2 in Figure 95; see [3.1.2 Information Bar](#))
- Control bar (3 in Figure 95; see [3.1.3 Control Bar](#))

Please read the instructions on the following pages to take advantage of all the available feature of *OpenSignals (r)evolution*.

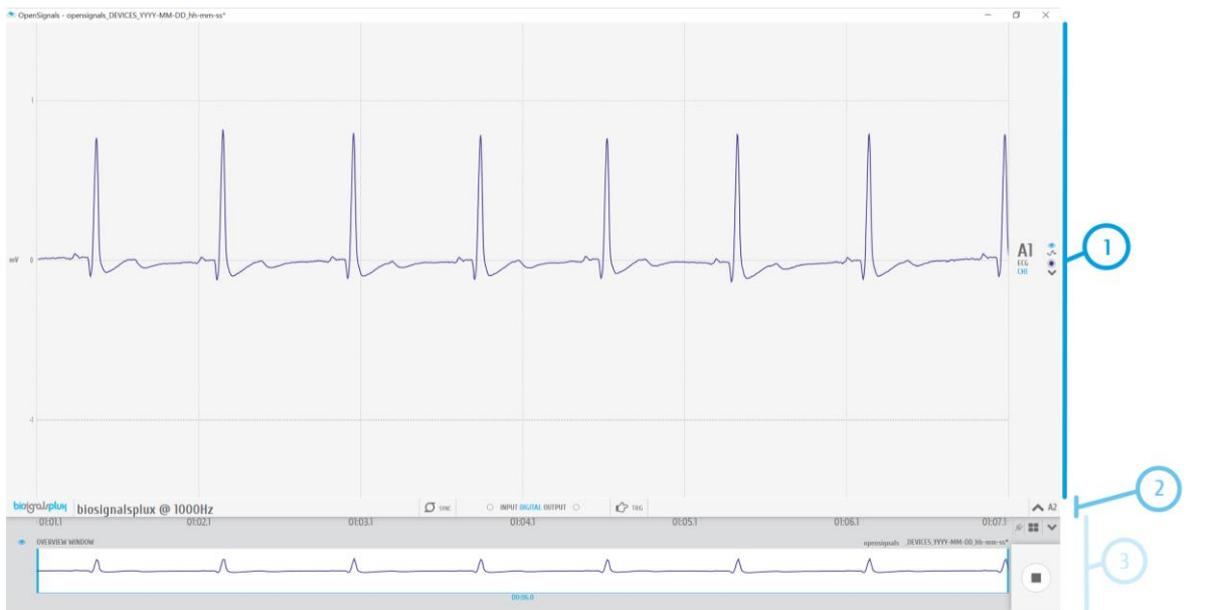


Figure 95: Sections of the visualization mode.

3.1.1 Signal Acquisition

The acquired signals are visualized in the signal acquisition window, which contains the signals of all acquiring channels (see Figure 96).



Figure 96: Visualization of the acquired signal(s).

In this window, there are several fields of information and configuration that lets you make some modifications on the displayed signals.

3.1.1.1 Channel info

The channel info provides information of the displayed channel (see Figure 96; here: channel 1 of a *biosignalsplux* device; **A1**), the type of channel which has been configured before the acquisition was made (here: **ECG**) and the label set for this channel and sensor during the acquisition (here: **CH1**)

Changes of these indications have to be made when configuring your device before starting an acquisition. Please review and follow the instruction in section 2.2.4.6 *Channel Configuration* for more information on channel types, number, and labelling.

3.1.1.2 Channel unit

The channel unit displays the original physical unit of the acquired signal, if a selection of channel type has been made during the configuration process of your device. The channel unit varies between types of sensor and selected sensor type (here: ECG signal displayed in mV). If no selection has been made, the default unit to be displayed is Volt (V).

Like referred on section 3.1.1.1, changes of these indications have to be made when configuring your device before starting an acquisition (section 2.2.4.6 *Channel Configuration*).

3.1.1.3 Visual Setup

The visual setup allows you to modify the visualization of your signal(s) (e.g. signal color) and to hide single channels, if their visualization is not required.

Additional functions to modify the visual setup of each channel can be accessed when hovering over each available channel (see Figure 97).



Figure 97: Visual setup options (blue box).



Display channel in the overview window

Hides or displays the selected channel in the overview window at the bottom of the visualization mode (see 3.2.4.5 *Overview Window* for more information).



Signal smoothing

This visualization option mode allows to change the plotting of the chart from the raw data to a sampling frequency weighted sub-sampled interpolation, which smooths the chart, showing only the general shape. This option is saved into the device configurations to a new property called *smoothChannels*, which defines, for each channel, whether to smooth or not the chart.



Signal color

Changes the color of the visualized signal. Click on this symbol to see all the available colors.



Hide channel

Click on this symbol to hide a single channel from the visualization mode. This feature is useful when the visualization of specific channels is not required at the moment. The hidden channel can be displayed again by selecting it in the information bar (see 3.1.2 *Information Bar*).



Move baseline up

Moves the baseline of the selected channel up (incl. signal).



Move baseline down

Moves the baseline of the selected channel down (incl. signal).



Zoom in

Zooms into the selected channel.



Auto-Zoom

Automatically scales the visualization in accordance to the available data.



Zoom out

Zooms out of the selected channel.



Link channels

Links channels of the same type to each other and adjusts the same visual setup to all linked channels (e.g. scale, offset).

3.1.2 Information Bar

In the information bar, information about the acquiring device is displayed (device logo, name or MAC-address, sampling rate) as well as all the hidden channels (channels which contain acquiring signals but are not being visualized during this acquisition). Additional information is given about the synchronization status (see 3.4 Synchronization) and the trigger status (see 2.2.4.8 Trigger Options).

The information bar does also provide options to control both digital input and outputs and, depending on the device being used, an adjustable output to control the analog output of a device (*BITalino* only).

When using *biosignalsplux* devices, the state of the digital inputs and outputs are displaced at the center of the information bar (see Figure 98).

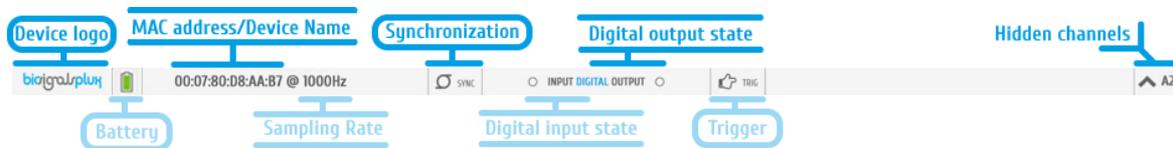


Figure 98: Information bar when using *biosignalsplux* devices.

When using *BITalino* devices, additional control functions are displayed to control both digital outputs and digital inputs of the device as well as adjusting the analog output using pulse width modulation (see Figure 99).

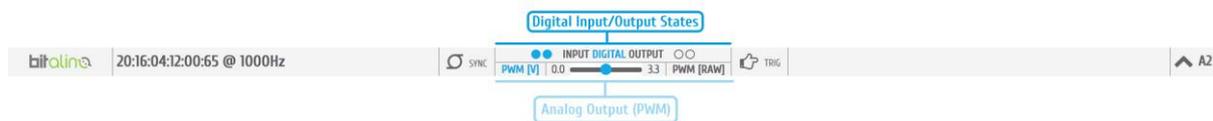


Figure 99: Information bar when using *BITalino* devices.

For more information on digital input and output control see section 3.1.2.4 Digital Inputs & Outputs. For more information on analog output control see 3.1.2.5 Analog Output.

3.1.2.1 General Controls

When using multiple devices during an acquisition, the synchronization function should be activated to ensure that all devices are synchronized during the acquisition (see 3.4 Synchronization). In the information bar it can be seen if the synchronization function is active or not.

Synchronization function

Indicates if *OpenSignals* is using the synchronization function to synchronize the acquisition of the multiple devices being used (function is active when symbol is blue; function is inactive when symbol is gray).

Hidden channels (see Figure 98; here: **A2**, channel 2) can be displayed in the acquisition window again by clicking on the arrow next to a hidden channel.

Show channel

Activates selected channel to be displayed in the visualization window.

3.1.2.2 Battery Indicator

Information about the *biosignalsplux's* battery level is shown in the information bar and is visualized as follows:

-  battery level between 50% & 100%
-  battery level between 10% & 50%
-  battery level between 0% and 10%
-  battery discharged; please proceed with the charging process

3.1.2.3 Trigger

Accessories such as the [pushbutton](#) or the [foot switch](#) can be used to trigger the start of an acquisition. Triggered acquisitions have to be configured in the device manager before starting the acquisition. Make sure you have the trigger function activated during the configuration process of your device as described in 2.2.4.8 Trigger Options.

In the information bar it can be seen if the synchronization function is active or not.



Figure 100: Trigger state.

Triggered acquisition

Indicates if the current acquisition is configured to start by a trigger input (function is active when symbol is blue; function is inactive when symbol is gray).

3.1.2.4 Digital Inputs & Outputs

Accessories such as the [pushbutton](#) or the [foot switch](#) can be used as digital markers to mark special events during an acquisition.

In the information bar, it is possible to see the current input state of such devices. If the acquiring device is receiving a digital input (e.g. handheld switch is pressed), the circle next to *INPUT DIGITAL* turn blue. Otherwise, it will remain empty (see Figure 101).

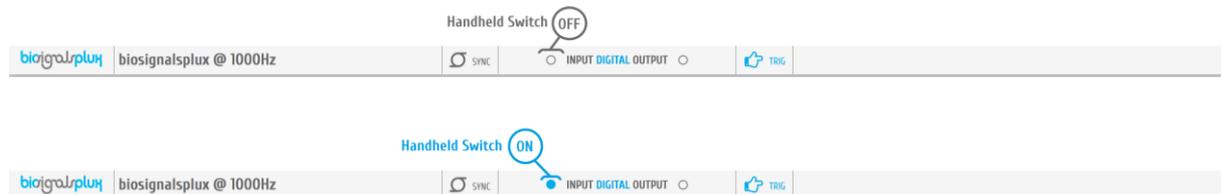


Figure 101: Digital input states (input device being used: handheld switch).

Accessories such as [LEDs](#) can be used as visual markers, for example, to synchronize PLUX devices with 3rd party hardware if needed. To control the output state of the LED during the acquisition, click on the circle next to *DIGITAL OUTPUT* in the information bar.

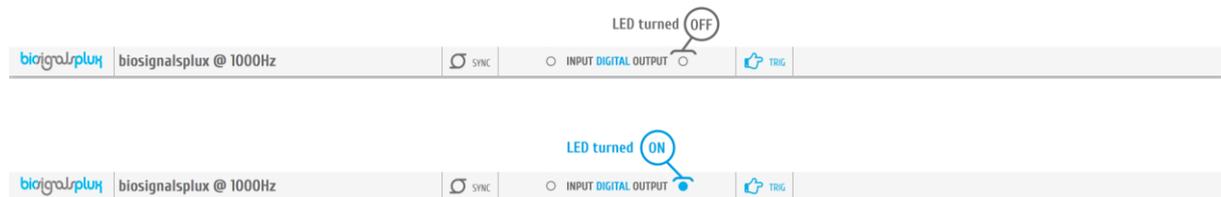


Figure 102: Digital output states for LED.

NOTE

BITalino devices allows you to control two digital outputs and two digital inputs. On *BITalino (r)evolution* boards the input and output indications are the following (referring to the circles in the information bar):

- Left digital input: button state
- Right digital input: -
- Left digital output: LED
- Right digital output: buzzer

This configuration can vary when using *BITalino Freestyle* or *BITalino Plugged* devices, taking into consideration that the configuration of these devices may not be identical to the one available at *BITalino Board*.

3.1.2.5 Analog Output

BITalino devices allows you to control an analog output to control additional hardware or 3rd party hardware (e.g. custom sensors). To control the analog output, click and drag the blue button to the preferred analog output (see Figure 103).



Figure 103: Adjusting analog output.

3.1.3 Control Bar

The following features and options can be accessed in the control bar:

- Time axis (x-axis; 1 in Figure 104)
- Access *OpenSignals (r)evolution* add-ons (2 in Figure 104)
- Multi-channel visualization modes (3 in Figure 104)
- Hide/Unhide all channels (4 in Figure 104)
- Overview window (x-axis adjustments; 5 in Figure 104)
- Return to *OpenSignals*' main interface (6 in Figure 104)



Figure 104: Control bar.

All features are explained on the following pages in more detail (except file information and access to open file dialog; see section 3.2.1 *Open OpenSignals Files* for more details).

3.1.3.1 Time Axis

The x-axis provides the time information and scaling for the acquired signals. These values are not changeable. However, the time span which is displayed can be adjusted by scaling the x-axis in the overview window.

3.1.3.2 Access *OpenSignals* Add-ons

OpenSignals (r)evolution can be extended by signal-processing add-ons to extract meaningful information from specific signal and sensor types. All available add-ons can be accessed by using the following symbol in the control bar to open the add-ons menu (see section 4 *Add-Ons* for more information).



Add-ons

Displays all available offline add-ons for post-processing purposes.

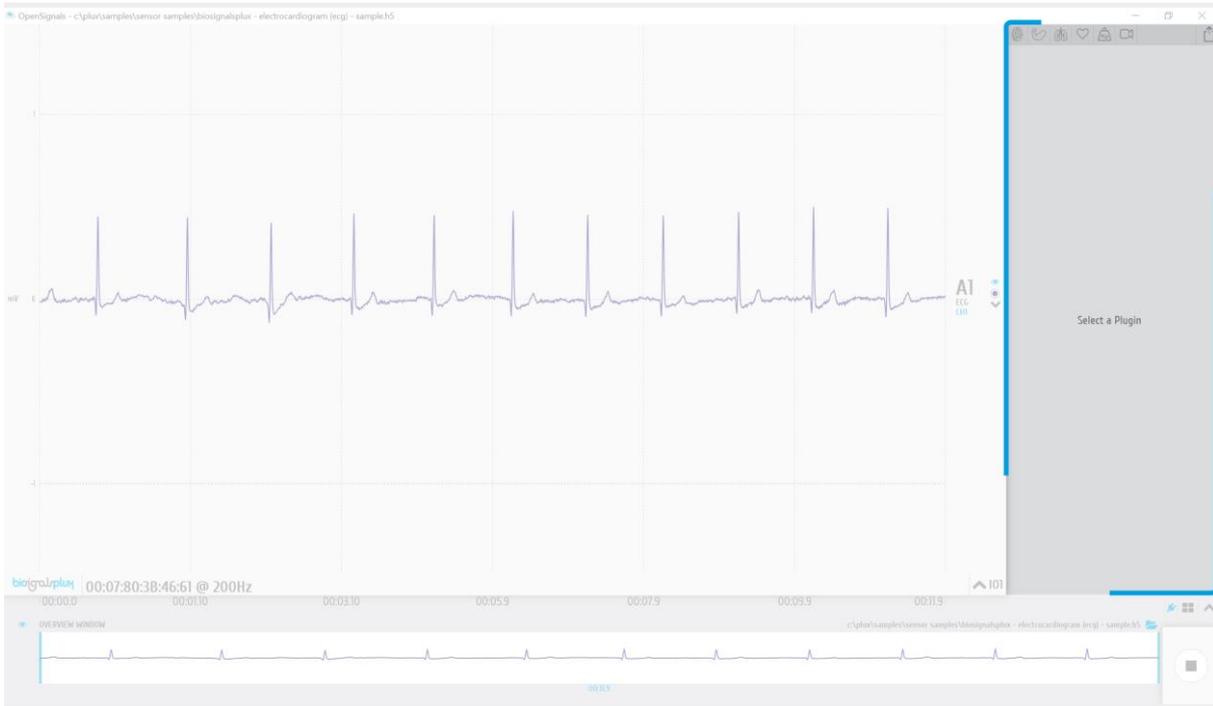


Figure 105: Add-ons menu.

3.1.3.3 Multi-channel Visualization Modes

OpenSignals (r)evolution offers two different channel visualization modes (see Figure 106). The first (default) mode displays all the channels stacked vertically in order of the channel number (see Figure 106, left visualization mode). The second type splits the visualization mode vertically in half, displaying half of the channels on the left side of the visualization window and the other half on the right side (see Figure 106, right visualization mode). This mode can be useful when visualizing multiple channels at the same time (e.g. using an 8-channel biosignalsplex).

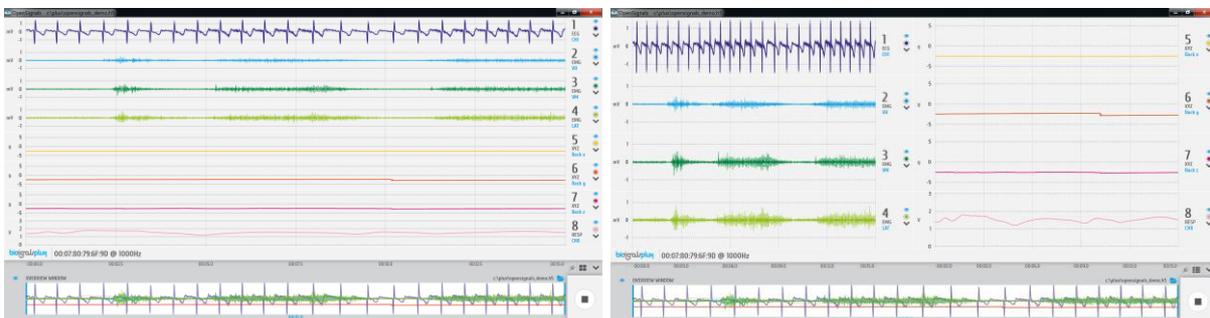


Figure 106: Channel visualization modes.



Multi-channel visualization modes

Switch between available multi-channel visualization modes.

3.1.3.4 Hide and Unhide Hidden Channels

By default, *OpenSignals (r)evolution* will display all the available channels and signals that are stored in a loaded *OpenSignals* file. As a permanent visualization of all available channels is not always required, it is possible to hide the visualization of selected channels as described in section 3.1.1.3 *Visual Setup*.

All channels can be hidden or unhidden at once by clicking on the following symbols:



Unhide all channels

Unhides all hidden channels, resulting in a visualization of all available channels and signals.



Hide all channels

Hides all channels.

3.1.3.5 Overview Window

The overview window gathers and displays all available signals in the bottom frame of the signal visualization mode. This window serves to change the displayed time interval of the signals in the signal visualization window (adjustment of the x-axis).

To change the displayed time interval, click on the blue vertical bars in the overview window and move them to the start or ending point of your preferred time interval. Only the time interval which lies within the white frame in the overview window will be displayed in the visualization window (here: from 6 sec interval to 3 sec interval).

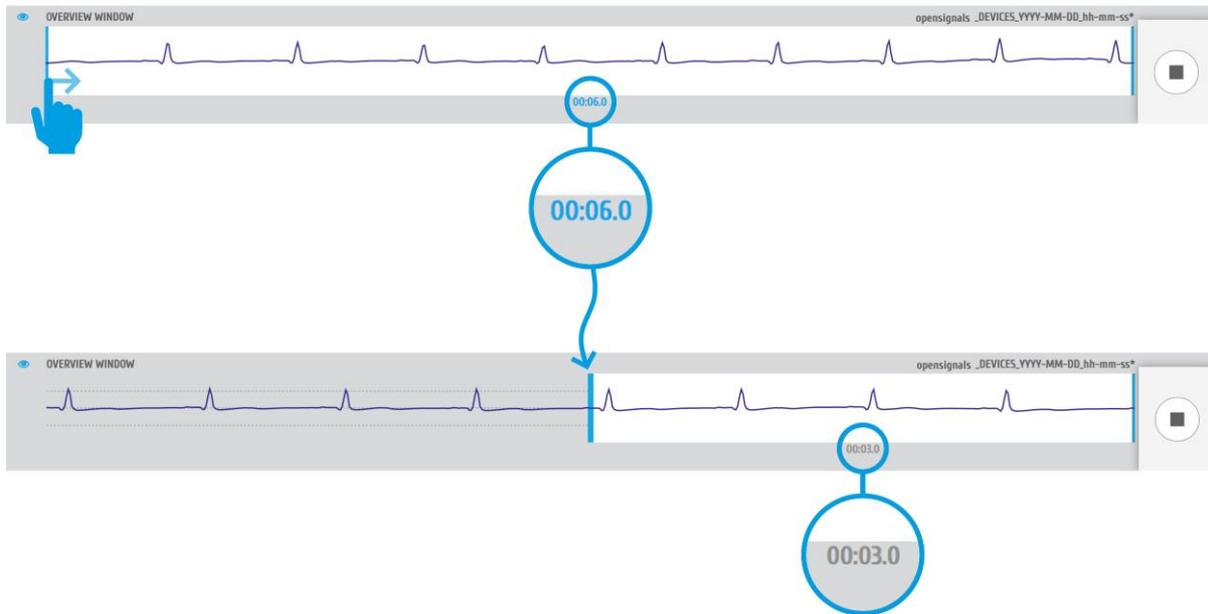


Figure 107: Changing the visualized time interval in the overview window.

Note that the signals will be displayed in the overview window with the color that has been selected for the signals/channels in the visualization window.

If not needed, the overview window can be hidden by clicking on the following symbol placed to the top-left corner of the overview window. When hidden, the overview window can be displayed by clicking on this symbol again.

Hide/Show overview window

Hides the overview window when it is displayed and displays the overview window when it is hidden.

3.1.3.6 Return to the Main Interface

The signal acquisition mode can be stopped and exit at any time by clicking on the square symbol at the bottom right of the signal visualization mode. Exiting this mode will result in returning to the main screen.

Stop signal recording and exit signal acquisition mode

Stops signal recording and exits signal acquisition mode (returns to main screen).

3.2 Signal Visualization Mode

OpenSignals (r)evolution allows you open files containing the previous signal acquisitions and/or to use *OpenSignals* add-ons to post-process your data (see 4 *Add-Ons*). To enter the signal visualization mode, click on the following symbol, which you can find in the main screen of *OpenSignals (r)evolution* (see Figure 35).

▶ enter signal visualization mode to review and/or post-process your data

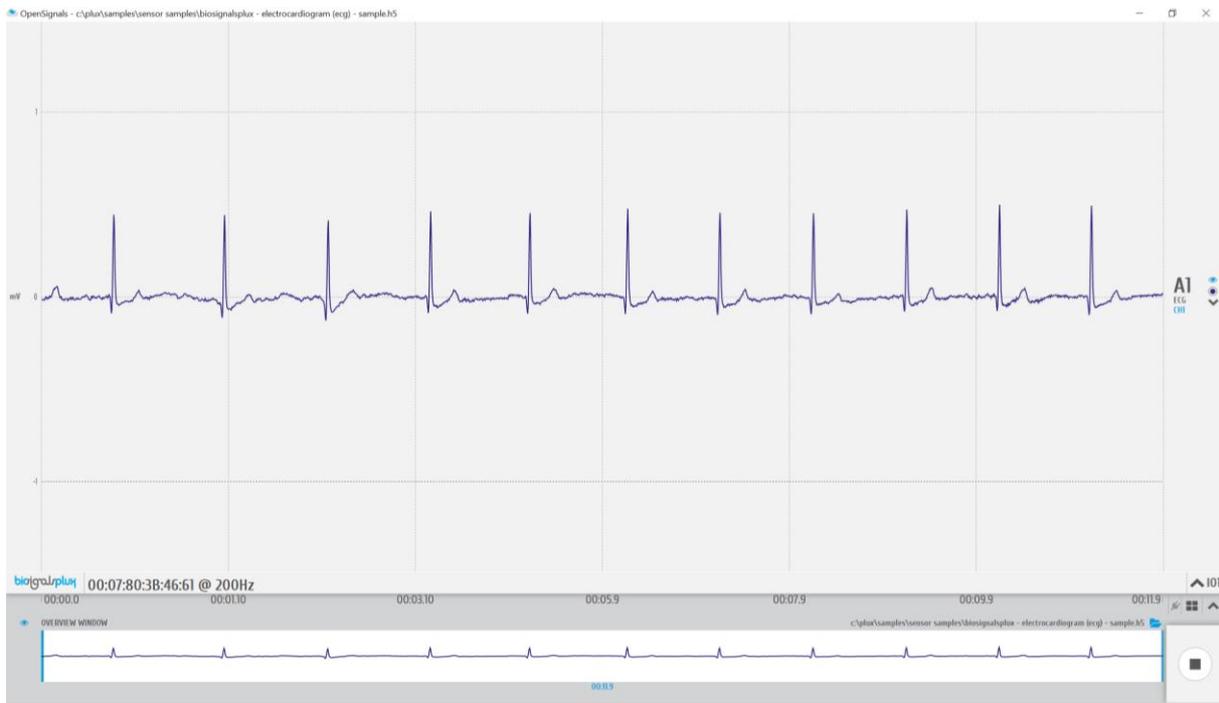


Figure 108: Signal visualization mode.

The visualization mode consists on 3 different interface sections which allow you to control the visualization of the selected files and signals (identical to the ones presented on section 3.1):

- Signal visualization (1 in Figure 109; see section 3.2.2 *Signal Visualization*)
- Information bar (2 in Figure 109; see section 3.2.3 *Information Bar*)
- Control bar (3 in Figure 109; see section 3.2.4 *Control Bar*)

Please read the instructions on the following pages to take advantage of all the available features, including the instructions on how to select and open *OpenSignals* compatible files.

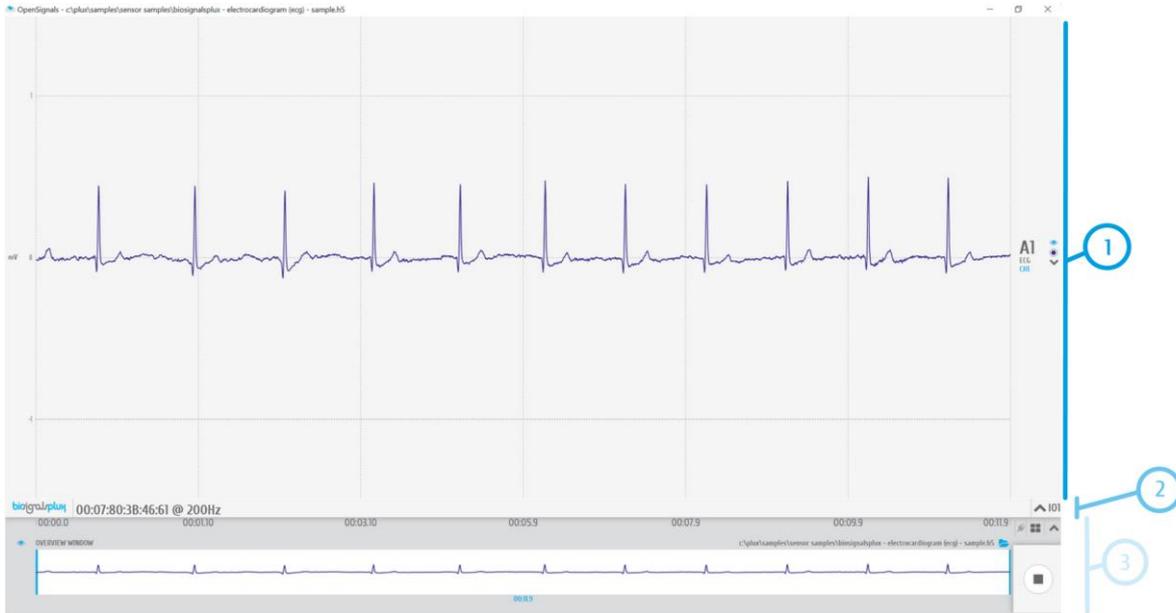


Figure 109: Sections of the visualization mode.

3.2.1 Open OpenSignals Files

To visualize your previous acquisitions, it is needed to select and open the file which contains acquired signals. For this, click on the folder symbol inside the signal visualization mode, to open the open file dialog which allows you to navigate through the folders on your computer and to select and open your preferred *OpenSignals* files (see Figure 110).

 navigate and select *OpenSignals* file



Figure 110: Open the open file dialog.

At the top of the open file dialog, all the available save locations and platforms are displayed (*Local, Repovizz, Google Drive...*), which are connected to your *OpenSignals (r)evolution* (as described in *2.3.6 Save Locations*). Click on the save location you have your files stored and wait for it to load the list of available files.

To navigate through the folder structures of your save location, click on a folder which is listed in the open file dialog to enter the folder or click on the first folder in the list (named “..”) to return to the previous folder. To select and open a file, click on your preferred file and confirm your selection by clicking on the *OK* button to open the selected file.

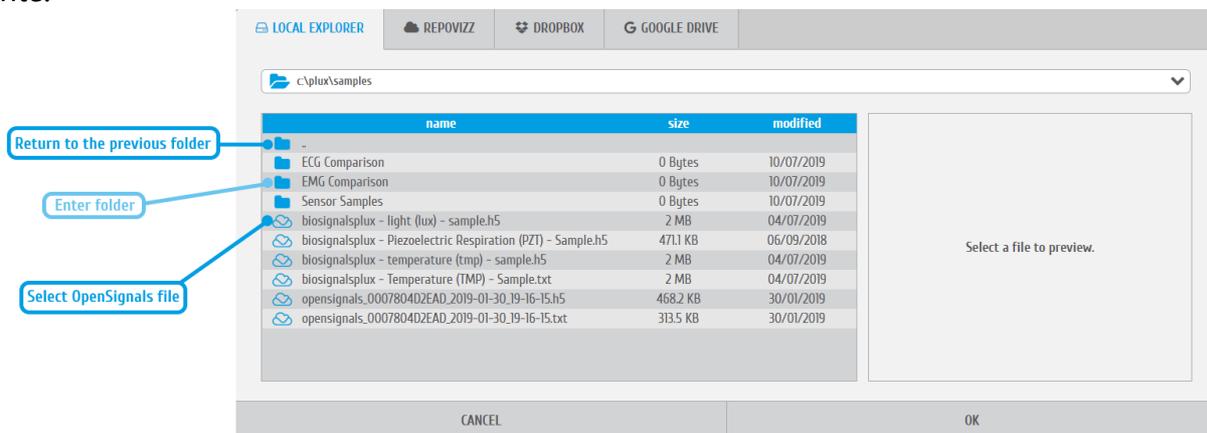


Figure 111: Open file dialog.

NOTE

Besides *OpenSignals (r)evolution* there are other 3rd party applications available which have been tested to work with PLUX devices and selected by our staff.

AcqKnowledge software is an example of a 3rd party application whose files can be opened, imported and visualized by *OpenSignals (r)evolution*.

<https://www.biopac.com/product/acqknowledge-software/>

3.2.2 Signal Visualization

The Signal Visualization screen is similar to the one described at section 3.1.1 Signal Acquisition.

At the current section we will only highlight some additional features.

3.2.2.1 Channel info

On the **Signal Visualization** mode it is possible to edit the sensor label, which you can change directly on the signal visualization screen through the arrow located near the sensor label (Figure 112).



Figure 112: Signal visualization window highlighting the Sensor Label edit arrow

3.2.2.2 Channel unit

The channel unit displays the original physical unit of the acquired signal, if a selection of channel type has been made during the configuration process of your device (or during the signal visualization, as presented before on section 1.1.1.1.) The channel unit varies from the type of sensor and selected sensor type (**here:** ECG signal displayed in mV). If no selection has been made, the default unit to be displayed is Volt (V).

3.2.3 Information Bar

In the information bar, information about the device which has been used for the selected acquisition is displayed (e.g. device logo, name or mac-address, sampling rate) and all the hidden channels (signals which have been used during this acquisition but are not visualized at the moment).



Figure 113: Information bar.

Hidden channels (see Figure 113; here IO1 (digital input/output)) can be displayed in the visualization window again by clicking on the arrow next to a hidden channel.

Show channel

Activates selected channel to be displayed in the visualization window.

3.2.4 Control Bar

The following features and options can be accessed in the control bar:

- Time axis (x-axis; 1 in Figure 114)
- File information and access to open file dialog (2 in Figure 114)
- Access *OpenSignals (r)evolution* add-ons (3 in Figure 114)
- Multi-channel visualization modes (4 in Figure 114)
- Unhide hidden channels (5 in Figure 114)
- Overview window (x-axis adjustments; 6 in Figure 114)
- Return to *OpenSignals*' main interface (7 in Figure 114)
- Switch Time-axis (new functionality; 8 in Figure 114)



Figure 114: Control bar.

All needed features are explained on the following pages in more detail (except file information and access to open file dialog; see section 3.2.1 *Open OpenSignals Files*).

3.2.4.1 Time Axis

The x-axis provides the time information for the acquired signals. These values are not changeable. However, the time span which is displayed can be adjusted by scaling the x-axis in the overview window.

3.2.4.2 Access *OpenSignals* Add-ons

OpenSignals (r)evolution can be extended by signal-processing add-ons to extract meaningful information from specific signal and sensor types. All available add-ons can be accessed by using the following symbol in the control bar to open up the add-ons menu (see section 4 *Add-Ons* for more information and).



Add-ons

Displays all available offline add-ons for post-processing purposes.

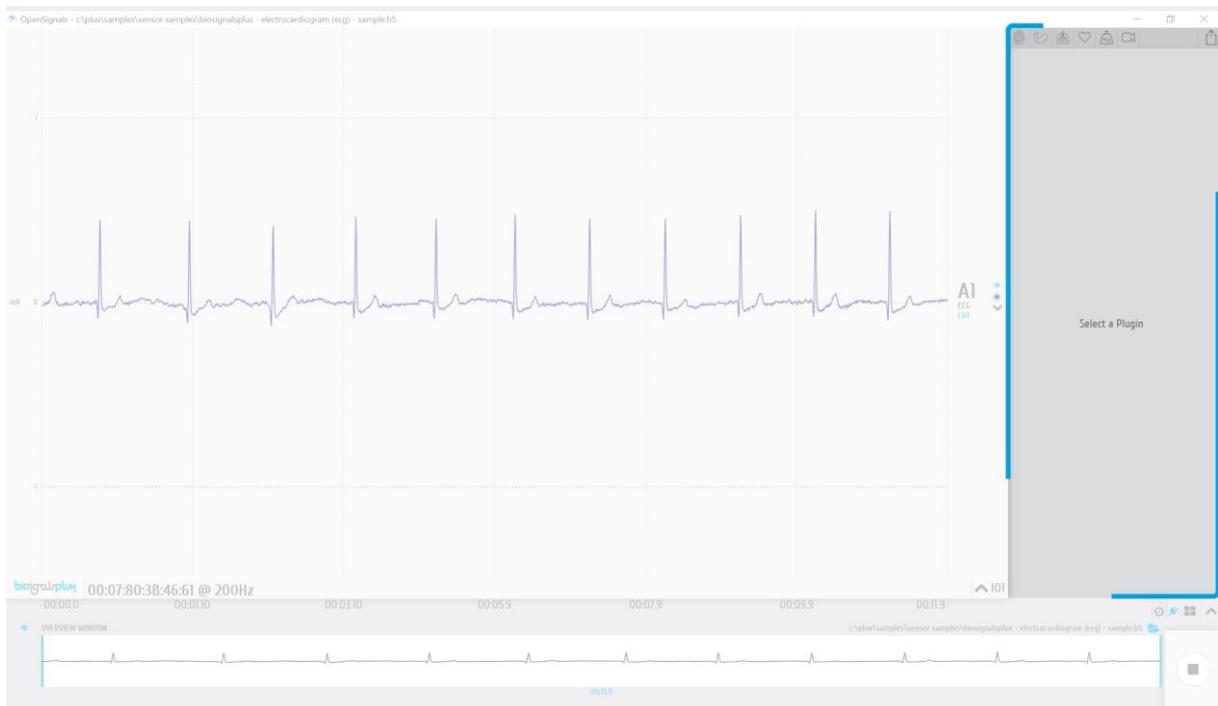


Figure 115: Add-ons menu.

3.2.4.3 Multi-channel Visualization Modes & Unhide Hidden Channels

The current functionalities are also available on the signal acquisition mode, so, in order to get access to a more detailed description you can check the contents of sections 3.1.3.3. and 3.1.3.4.

3.2.4.4

3.2.4.4 Switch Time-Axis format

The current functionality was recently included on *OpenSignals (r)evolution* and ensures two paradigms of time-axis values.

The first option (predefined) for presenting the time-axis consists in relative values, i.e., the start of acquisition is always the reference value corresponding to 00:00.0.

So, in a 10m50s acquisition, it will be presented a time-axis within the range [00:00.0; 10:50.0].

The second available option corresponds to the presentation of absolute timestamps, i.e, each signal sample will have as x coordinate the exact daytime hour when it was acquired (from the device internal clock).

As an example, in a 10m50s acquisition started at 15h21m00s., the respective time-axis will be [15:21:00; 15:31:50].



Switch Time-Axis

Button responsible for interchanging between relative and absolute time formats.

3.2.4.5 Overview Window & Return to the Main Interface

As stated at the beginning of section 3.2 some screens/functionality are shared between the Signal Acquisition and Signal Visualization modes, being these two elements common.

So, in order to get access to a detailed description about the Overview Window and Return to the Main Interface we recommend to check sections 3.1.3.5 and 3.1.3.6.

3.2.5 Visualization of Digital Input and Output Signals

Digital inputs/outputs from your devices can be controlled during signal acquisition by following the instructions in *3.1.2.4 Digital Inputs & Outputs*. The digital input/output signals are visualized with all the other acquired signals in the visualization mode, with each digital input and output being displayed in an individual channel (see Figure 116).

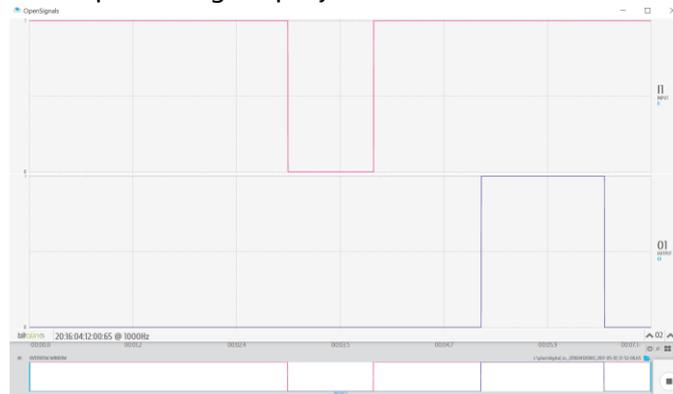


Figure 116: Visualization of digital input (I1) and output (O1) signals.

3.2.6 Changing the Sensor Type of a Channel in Post-Acquisition

It is recommended to configure the sensor type of each acquisition before executing an acquisition, as described in the instructions in Section 2.2.4 *Device Configuration*.

However, in case this has not been done before (as shown in the screenshot below in highlight 1 and 2), it is possible to change the sensor type in the visualization mode for previous acquisition sessions.



Figure 117: Acquisition data without sensor type specification in the OpenSignals visualization mode.

The following instructions demonstrate how to change the sensor type in the visualization mode.

To change the sensor type in the visualization mode, click on the arrow next to "RAW" (3) to open the dropdown menu with all the available sensor types, and select the type that corresponds to your sensor (in this case ECG).



Figure 118: Dropdown menu to select one of the available sensor types.

Afterwards, the raw digital sensor data will automatically be converted to the original physical unit of the sensor, as seen in the screenshot below where the sensor values now range between -1.5mV and 1.5mV (4) as the sensor type is now set to "ECG" (4).

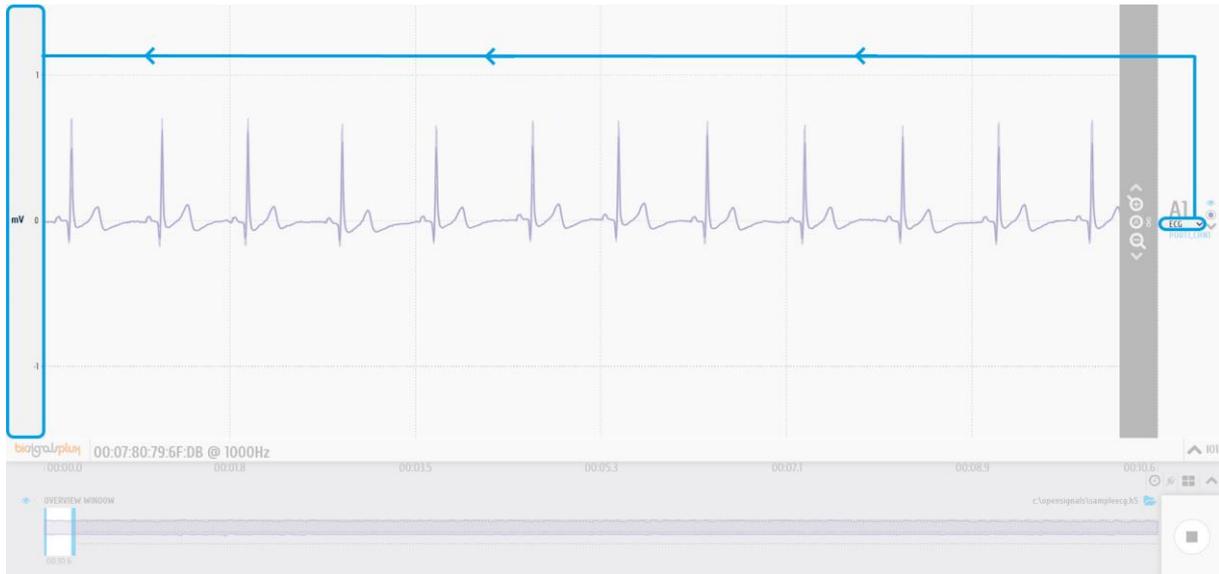


Figure 119: Acquisition data without sensor type specification in the OpenSignals visualization mode.

The newly converted raw signal is stored in its original units in a new .H5 file with a “_converted” filename extension.

Example

- *Original acquisition filename:* SampleECG.txt
- *New converted signal's filename:* SampleECG_converted.h5

3.3 Data Logging

biosignalsplux devices with internal memory allow you to schedule and manage offline acquisitions which can be initialized and conducted without the requirement of a Bluetooth connection with your computer and *OpenSignals (r)evolution*.

The following devices are compatible with data logging features (scheduling, recording, and downloading recorded data):

- biosignalsplux Professional
- biosignalsplux Force
- biosignalsplux Hybrid-8
- MotionBAN
- RespiBAN Professional

Internal memory can also be added posteriorly for all *biosignalsplux* devices. Please access our support via <https://support.pluxbiosignals.com/> if you'd like to add internal memory to your device to use all data logging features.

Devices with internal memory are displayed with an additional *SCHEDULE* button at the end of their device panels (see Figure 120). These devices are compatible with all data logging features.



Figure 120: Data logging feature for devices with internal memory.

3.3.1 Scheduling Offline Acquisitions

NOTE

Before scheduling offline acquisitions, we recommend to set up the device and channels first. To do so, configure your device to fit your set up and channel configuration according to the instructions in section 2.2.4 *Device Configuration* before entering the schedule panel.

Click on the *SCHEDULE* button of your device with internal memory in the device panel to access the scheduling panel to schedule and configure your offline acquisition (see Figure 121).

The computer Bluetooth needs to be turned on during this stage (to guarantee an establishment of a temporary connection with the device in order to schedule the acquisition). After the scheduling has been completed it won't be necessary during the real-time acquisition, considering that the acquired data will be stored in the internal memory instead of being transmitted.

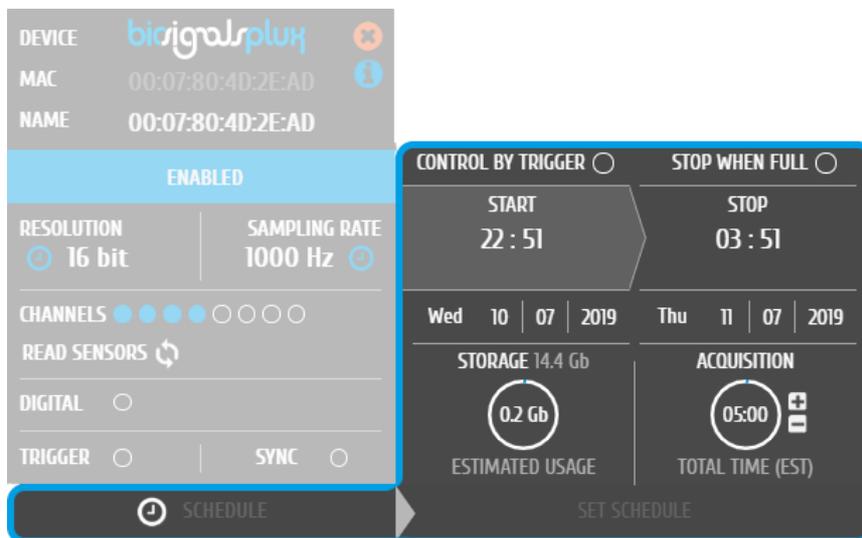


Figure 121: Scheduling panel.

The scheduling panel allows to configure your offline acquisition by defining the type of initiator for the start of your acquisition (timed or triggered), to stop the acquisition when the limits of the internal memory are reached (*STOP WHEN FULL*), to time acquisition at a specific time or duration (*STOP* and *ACQUISITION*), and displays information about the available free memory and the estimated memory usage of your scheduled acquisition.

NOTE

The indications of the estimated end of the acquisitions are calculated estimations and might vary from real-life behavior.

NOTE

It is recommended to fully charge your device's battery before using offline acquisitions as the acquisitions can be of longer period than the battery life-time of your device, for example when using the *STOP WHEN FULL* option. The estimated end will not be reached if the battery runs out before.

NOTE:

Your device does not have to be turned on before your scheduled acquisition starts, as it will automatically be turned on at the scheduled start-point.

To check if your device is acquiring data, check the LED information of your device as listed in chapter 5 *Advanced Device Configuration*.

3.3.1.1 Control by Trigger

This option allows you to manually start your acquisition using one of the available trigger accessories for *biosignalsplux* (footswitch or trigger button). To activate this feature, click on the circle next to *CONTROL BY TRIGGER* and make sure that your trigger accessory is correctly connected to the digital port of your *biosignalsplux* device.

This feature is activated if this circle is filled blue and inactive if the circle is empty. Activating this feature will also remove every time information in the *START* and *STOP* fields.

To set up a fixed duration of your offline acquisition, set up your preferred duration in the *ACQUISITION* field by using the '+' and '-' buttons. The acquisition will then stop when the duration has been passed, with the duration starting at the point where the trigger signal has been given.

Click on the *STOP WHEN FULL* option to stop the acquisition only when there is no memory left on your device. Read the following section for more information about this option.

WARNING

It should be noted that the ***CONTROL BY TRIGGER*** option is only applicable to the "start" of the acquisition. To ensure an automatic "stop" of the acquisition, the ***DURATION*** should be specified or the ***STOP WHEN FULL*** option activated.

3.3.1.2 Stop When Full

The *STOP WHEN FULL* option will stop your offline acquisition when the internal memory of your device has run out of available memory. This function can both used with trigger-based or timer-based initiated offline acquisitions. If a fix time has been selected for the start of the acquisition, the estimated end of the acquisition will be calculated based on the available free memory and will be displayed in the *STOP* field of the scheduling panel (see Figure 122).

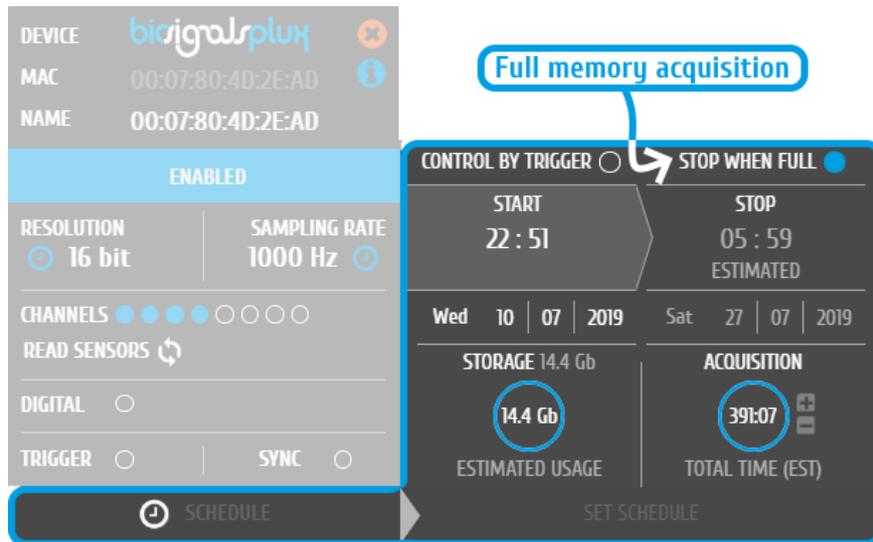


Figure 122: Scheduling panel displaying the estimated end of the acquisition

3.3.1.3 Scheduling Start & End Times

Click on the displayed time in the *START* or *STOP* fields to change the preferred start or end of your acquisition. In the new pop-up, select the date and time of the start or end of the acquisition. To select the date, click on the day of your choice in the calendar. To select the time, use the *Hour* and *Minute* sliders to configure your preferred time. By clicking the *NOW* button, the current date and time will be configured automatically.

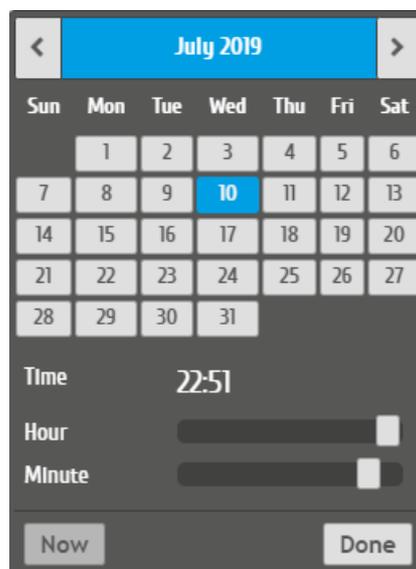


Figure 123: Time schedule for start and end points for offline acquisitions.

Alternatively, you can also set up a fixed duration for your acquisition instead of configuring an end time. To do so, set up your preferred duration in the *ACQUISITION* field by using the '+' and '-' buttons. The acquisition will then stop when the duration has been passed, with the duration starting at the scheduled time configured in the *START* field.

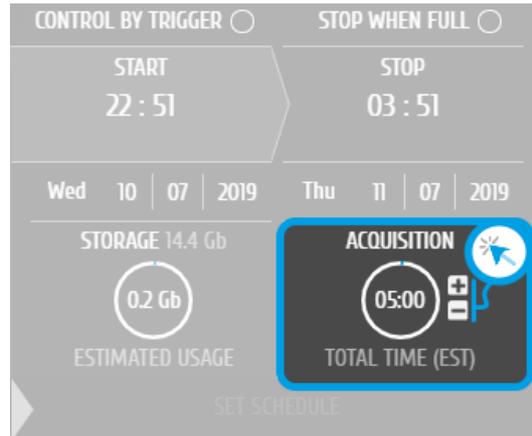


Figure 124: Configuring an acquisition with fixed duration.

3.3.1.4 Setting & Clearing Acquisition Schedules

If you've finished configuring your device and your offline acquisition, click on the *SET SCHEDULE* button at the bottom of the scheduling panel to transmit all the configurations to your device. The transmission and configuration has been successful if the button at the bottom of your device panel in *OpenSignals (r)evolution* has changed from *SCHEDULE* to *SCHEDULED* followed by the day and time when the acquisition has been configured to start (non-triggered acquisitions only) (see Figure 125; here: Wednesday 22:51h).

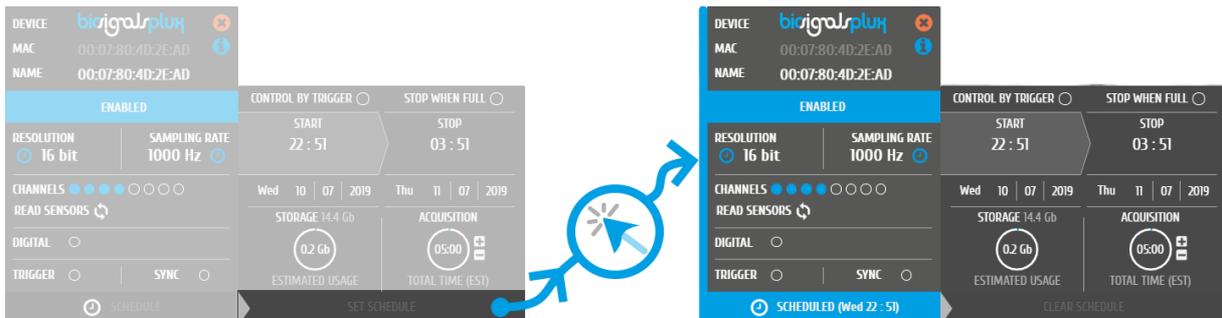


Figure 125: Completing the scheduling procedure.

To clear a scheduled acquisition, click on the *SCHEDULED* button and click on the *CLEAR SCHEDULE* button of the scheduling panel. The scheduled offline acquisition will be removed, and your device can now be configured for a new offline acquisition.

NOTE

If the scheduled acquisition is already running, instead of clearing the schedule (once the *SCHEDULED* button is pressed), the user can stop the running acquisition.

3.3.2 Downloading Offline Acquired Data

After your offline acquisition have finished, you can download the recording by clicking on the download data button in the *OpenSignals (r)evolution* main screen:

- Ⓣ Download an acquisition from the memory card

The offline data can be downloaded from your device’s internal memory within *OpenSignals (r)evolution* via Bluetooth or via the *fast USB data transfer* cable (additional accessory). The following sections provide the instructions on how to download data using both methods.

3.3.2.1 Download via Bluetooth

Make sure your *biosignalsplux* is turned on and click on the download button in the *OpenSignals (r)evolution* main screen to access the download panel. In this panel, all the available offline acquisitions which are available for download from your device’s internal memory are listed.

Each offline acquisition creates its own file where the acquired data is stored. In the download panel, the files are listed with their file name (*FILE*) their start time (*START*) and their entire duration (*DURATION*). Click on the indicated arrow in a file’s row to download the selected file (see Figure 126) or click on the arrow in the top row of the downloading panel to download all available files. The downloaded files will be stored on your default save location (see. 2.3.6.1 *Local*).

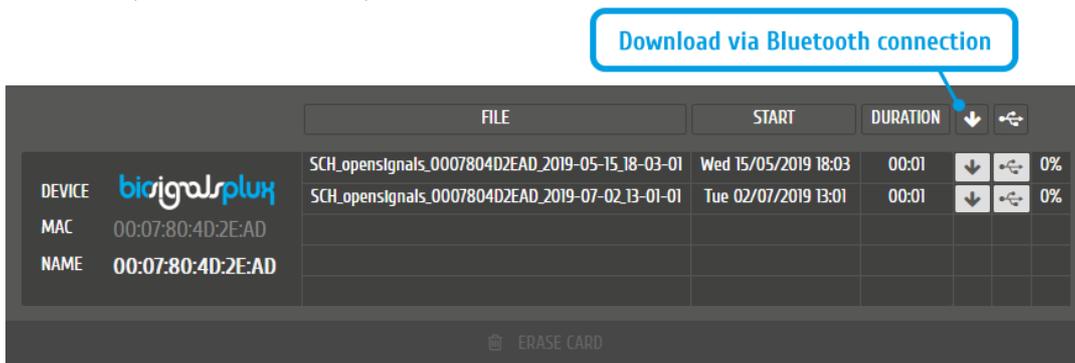


Figure 126: Downloading offline acquired data via Bluetooth.

NOTE

The time needed to download an acquisition file depends on the size of the file and, therefore, on the duration of your scheduled acquisition. The download process might need several minutes to end. Do not use nor turn off your device when *OpenSignals (r)evolution* is downloading data, as doing so might make the downloaded file unreadable.

3.3.2.2 Download via Fast USB Data Transfer Cable

NOTE

This method requires accessories which have to be bought additionally. These accessories are available in our store.

biosignalsplux USB Adapter & fast USB data transfer cable:

<https://www.pluxbiosignals.com/products/fast-usb-data-transfer-cable-for-biosignalsplux>

We should refer that, currently, this accessory is only compatible with *Microsoft Windows* operating system.

Make sure your *biosignalsplux* is turned on and connected to your computer via USB with the *biosignalsplux USB adapter*. Click on the download button in the *OpenSignals (r)evolution* main screen to access the download panel. In this panel, all the available offline acquisitions which are ready for download from your device's internal memory are listed.

Click on the indicated USB symbol in a file's row to download the selected file (see Figure 127) or click on the USB symbol in the top row of the downloading panel to download all available files. The downloaded files will be stored on your default save location (see. 2.3.6.1 *Local*).

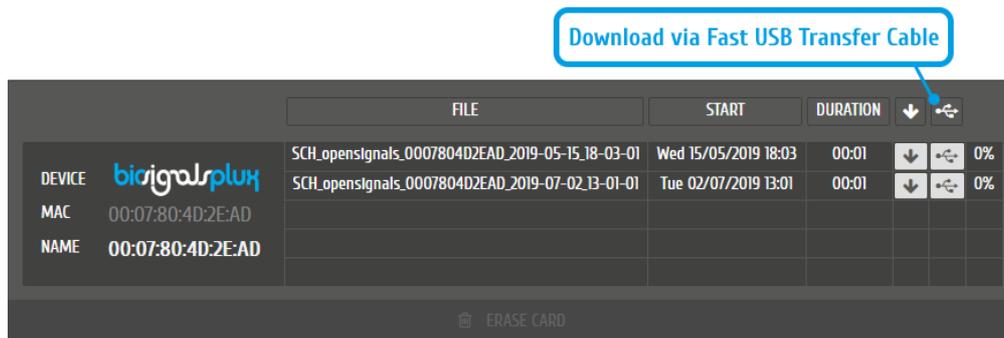


Figure 127: Downloading offline acquired data via USB.

NOTE

The time needed to download an acquisition file depends on the size of the file and, therefore, on the duration of your scheduled acquisition. The download process might need several minutes to end.

Do not use nor turn off your device when *OpenSignals (r)evolution* is downloading data, as doing so might make the downloaded file unreadable.

3.3.3 Deleting Offline Acquired Data

To delete all files stored on the internal memory of your device, click on the *ERASE CARD* button at the bottom of the download panel.

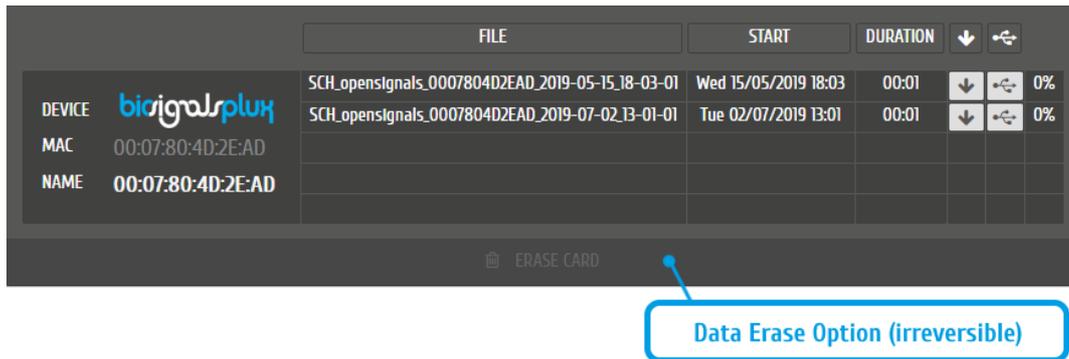


Figure 128: Available option to erase all data inside the internal memory of the device.

NOTE

Deleted files **cannot be restored**. Make sure you have downloaded all the needed files **before** deleting or erasing the internal memory of your device to prevent accidental data loss.

3.4 Synchronization

One of *OpenSignals (r)evolution's* features is the possibility of executing synchronized multi-device data acquisitions. This feature allows together up to three 8-channel *biosignalsplux* devices with each other using the *biosignalsplux* [sync-cables](#) and the [multi-sync splitter cable](#) to acquire signals from all devices simultaneously in a synchronized acquisition, preventing signal drifts between the single devices (time based differences) which could occur in non-synchronized long-time acquisitions.

NOTE

Signal drift may still occur when using accessories to synchronize PLUX devices with other devices (e.g. LED).

NOTE

The *biosignalsplux* synchronization cable/kit is needed for the use of this function. This additional accessory can be found here:

<https://www.pluxbiosignals.com/products/synchronization-kit>

3.4.1 Connecting Devices

NOTE

Components that are used for this feature require a digital port to synchronize the acquisition. By default, only the 8-channel *biosignalsplux* has a digital port that works with these accessories and this feature. A digital port can additionally be added to 4-channel *biosignalsplux* devices (contact our support for more information).

Connect the sync-cable to the digital ports of your devices. The digital port is the last input on the right side of the *biosignalsplux* device as marked in Figure 129. The devices should be connected as displayed in Figure 130.



Figure 129: Digital port of the 8-channel *biosignalsplux*.



Figure 130: biosignalsplux devices connected with a sync-cable.

When using 3 *biosignalsplux* devices, connect the single end of the multi-splitter to the digital port of one *biosignalsplux* device and connect two sync-cables to 2 of the available 3 inputs of the multi-sync splitter. Connect the other ends of the sync cables to the digital ports of the two *biosignalsplux* devices which are going to be used in the synchronized acquisition.

3.4.2 Device Configurations in OpenSignals (r)evolution

Configure all your devices as needed by following the instructions in [2.2.4 Device Configuration](#). When done, enable all devices which will be used during the acquisition by clicking on the *ENABLED* field of each device, click on their device panels and activate the sync-function (*SYNC*; circle must be blue) if it is not activated. When done, your device panel should look similar to the one seen in Figure 131.

NOTE

Variations in resolution, sampling rate, active channels, and/or trigger function of your configuration compared to the one displayed in Figure 131 will not influence your synchronized acquisition.

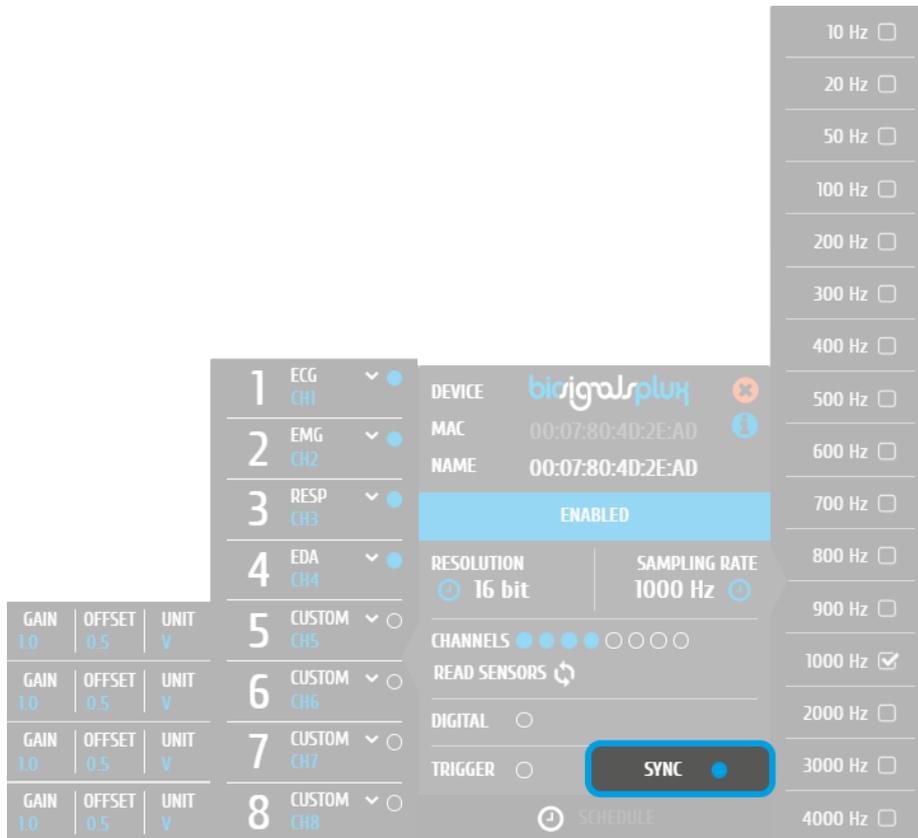


Figure 131: Activated sync-function.

When acquiring from several devices with a correctly set up synchronized acquisition, the sync-function symbol should be displayed in blue in the signal acquisition mode (see Figure 132).

If the sync-function is not active (symbol is displayed in gray), restart the configuration of your devices by following the instructions above.

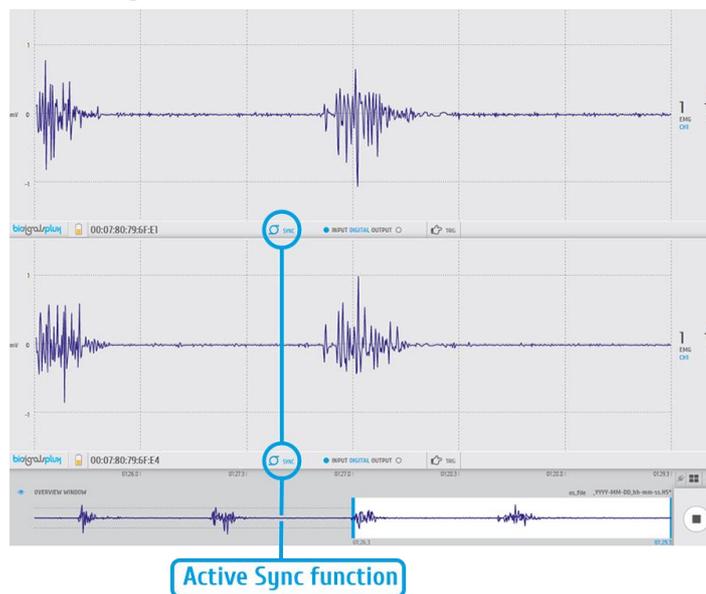


Figure 132: Synchronized acquisition with active sync-function.

3.5 Accessories

OpenSignals (r)evolution is compatible with all PLUX devices and sensors, as well as the following accessories which are available for most devices.

NOTE

biosignalsplux Accessories need to be connected to the digital port of *biosignalsplux* devices. Please check if your device does have a digital port before considering using one of the following accessories (4-channel *biosignalsplux* devices do not have a digital port (can be added additionally); 8-channel *biosignalsplux* devices do have a digital port).

3.5.1 Trigger

NOTE

See also 2.2.4.8 Trigger Options and 3.1.2.3 *Trigger*. For data logging acquisition see also 3.3.1.1 *Control by Trigger*.

3.5.1.1 Handheld Switch

This trigger device is useful to begin acquisition via Bluetooth to *OpenSignals (r)evolution* (or any other custom software) or to manually start offline acquisitions stored in the hub's internal memory when no Bluetooth connection is available.



Figure 133: Handheld switch.

How to connect your handheld switch to your biosignalsplux

The handheld switch needs to be connected to the digital port of your device.

How to configure your handheld switch in OpenSignals (r)evolution to mark events

Input devices, such as the handheld switch, do not require to be configured in *OpenSignals (r)evolution* if their intended use is to mark specific actions during an acquisition. In this case, their input state can be seen in *OpenSignals'* signal acquisition mode in the information bar located right below the acquired signals (see Figure 134; circle to the left of *INPUT DIGITAL*). The digital input circle will turn blue if a digital input has been received (handheld switch pressed) or be kept empty if no digital input has been received (handheld switch not pressed; see Figure 134).

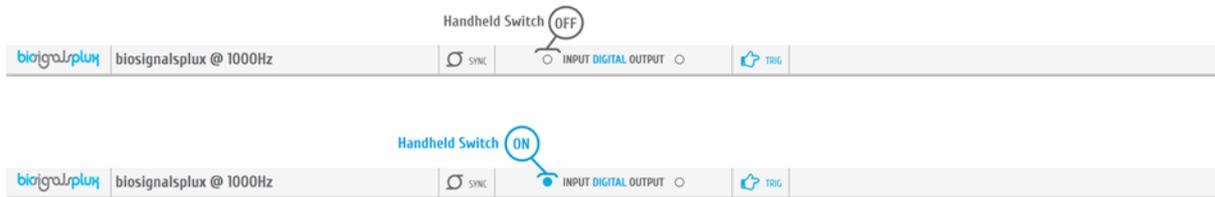


Figure 134: Digital input states.

The received inputs will be visualized in an extra channel named *I01* in the visualization mode.

How to configure your handheld switch in OpenSignals (r)evolution to trigger acquisitions

If the intended use of the handheld switch is supposed to trigger the start of an acquisition, it is needed to turn on this feature in the device manager. For this, open the *OpenSignals (r)evolution* device manager and select the device you have your handheld switch connected to. In the device's panel, click on the circle next to *TRIGGER* to activate the trigger feature (circle must be blue; see Figure 135).



Figure 135: Trigger configuration.

In the information bar of *OpenSignals'* real-time acquisition mode (located right below the acquired signals), the state of the trigger feature is displayed (see Figure 136).



Figure 136: Trigger state.

Accessory specifications

Please read the datasheet of handheld switch carefully before using it the first time. The datasheet can be downloaded here:

<https://support.pluxbiosignals.com/wp-content/uploads/2021/10/biosignalsplux-Handheld-Switch-Trigger-Button-Datasheet.pdf>

3.5.1.2 Foot Switch

This trigger device is useful to begin acquisition via Bluetooth to *OpenSignals (r)evolution* (or any other custom software) or to manually start offline acquisitions stored in the hub's internal memory when no Bluetooth connection is available.



Figure 137: Foot switch.

How to connect your foot switch to your biosignalsplux

The foot switch needs to be connected to the digital port of your device.

How to configure your foot switch in OpenSignals (r)evolution to mark events

Input devices, such as the foot switch, do not require to be configured in *OpenSignals (r)evolution* if their intended use is to mark specific actions during an acquisition. In this case, their input state can be seen in *OpenSignals'* real-time acquisition mode in the information bar located right below the acquired signals (circle to the left of *INPUT DIGITAL*). The digital input circle will turn blue if a digital input has been received (foot switch pressed) or be kept empty if no digital input has been received (foot switch not pressed; see Figure 138).

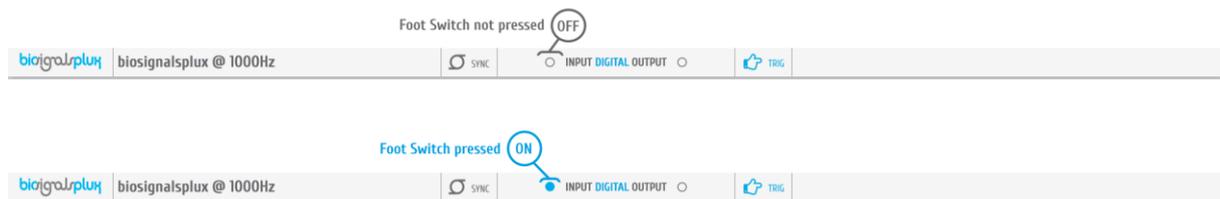


Figure 138: Digital input states.

The received inputs will be visualized in an extra channel named *IO1* in the visualization mode.

How to configure your foot switch in OpenSignals (r)evolution to trigger acquisitions

If the intended use of the foot switch is supposed to trigger the start of an acquisition, it is needed to turn on this feature in the device manager. For this, open the *OpenSignals (r)evolution* device manager and select the device you have your foot switch connected to. In the device's panel, click on the circle next to *TRIGGER* to activate the trigger feature (circle must be blue; see Figure 139).

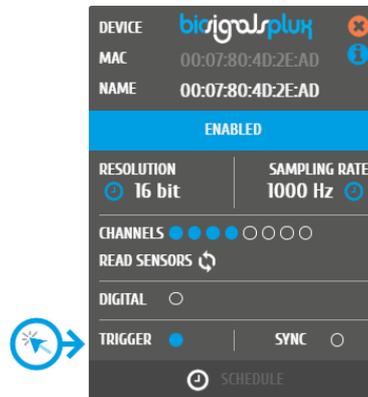


Figure 139: Trigger configuration.

In the information bar of *OpenSignals*' real-time acquisition mode (located right below the acquired signals), the state of the trigger feature is displayed (see Figure 140).



Figure 140: Trigger state.

3.5.2 Light-Emitting Diode (LED)

Note

See also [2.2.4.7 Digital Output](#) and [3.1.2.4 Digital Inputs & Outputs](#).

The LED is typically used to provide visual feedback to the user. A common need when working with biosignals is the synchronization of the recorded data with external recording devices (e.g. a video camera). If applied to a camera lens, the LED sensor can be used to introduce common markers in the recording, hence providing a synchronization source. The LED sensor can also be useful for optical synchronization with third-party devices (provided that the third-party device has a photo detector), in applications where it is important to have electrical decoupling between devices.



Figure 141: LED.

How to connect your accessory to your biosignalsplx

The LED needs to be connected to the digital port of your device.

How to control your LED in OpenSignals (r)evolution

Output devices, such as the LED, do not require to be configured in *OpenSignals (r)evolution*. However, it is possible to select their initial state which can later be changed in the information bar of *OpenSignals*' real-time acquisition mode which is located right below the acquired signals (see Figure 143; circle to the right of *DIGITAL OUTPUT*).

To select an initial state, open the *OpenSignals (r)evolution* device manager and select the device you have your LED connected to. In the device's panel, click on the circle next to *DIGITAL* to turn on (blue circle) or off (empty circle) the LED for the start of your signal acquisition (see Figure 142).

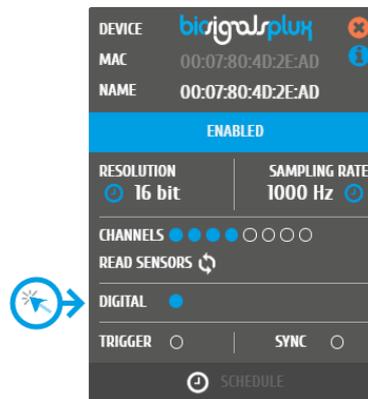


Figure 142: Initial state of the digital output (here: on).

To change the output state of the LED during the acquisition, click on the circle next to *DIGITAL OUTPUT* in the information bar of the real-time acquisition mode. Here, again, the LED will be turned on if the circle is blue and off is the circle is empty (see Figure 143).

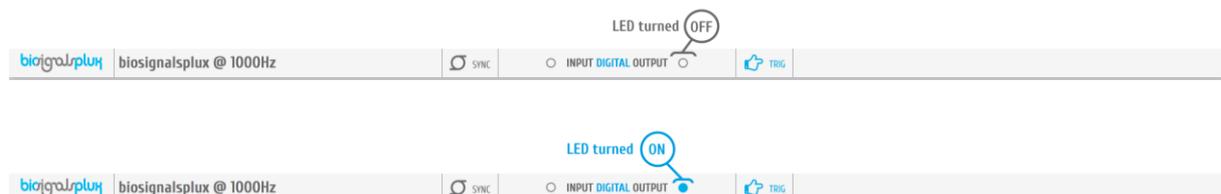


Figure 143: Digital output states for LED.

OpenSignals (r)evolution Add-On: Video Synchronization

The *Video Synchronization* add-on has been designed for research applications which involve recording data from sources other than the biosignal acquisition hardware devices (e.g. video camera). Given that the biosignal hardware and the camera are independent recording sources, a common problem when replaying the recording session is the synchronization of both. This plugin was created to provide an easy way to replay biosignal data synchronously with video using this LED.

For more information see section 4.17 *Video Synchronization*.

Accessory specifications

Please read the datasheet of LED carefully before using it the first time. The datasheet can be downloaded here:

https://support.pluxbiosignals.com/wp-content/uploads/2021/11/LED_Datasheet.pdf

3.5.3 Digital Synchronization Cable

Note

See also *3.4 Synchronization*.

The multi sync splitter accessory meets your multi-device acquisition needs. By using digital synchronization cable(s), this accessory allows you to connect *biosignalsplux* devices with digital ports together to collect up to 24 channels at the same time (by connecting up to 3 HUBs together).



Figure 144: Sync cable & kit.

How to connect your accessory to your biosignalsplux

The digital synchronization cable needs to be connected to the digital ports of the devices that are being used for the multi-device acquisition (see *3.4 Synchronization*).

How to configure synchronized acquisitions in OpenSignals (r)evolution

Open the *OpenSignals (r)evolution* device manager and select one of the devices you want to use for the multi-device acquisition. In the device's panel, click on the circle next to *SYNC* to turn on (blue circle) or off (empty circle) the synchronization function to synchronize the acquisition of your devices (see Figure 145). This function must be active to ensure synchronized multi-device acquisitions without signal drifts. Repeat this step for the second device that is being used for the multi-device acquisition.



Figure 145: Activate synchronization feature.

3.5.4 Multi Sync Splitter

Note

See also [3.4 Synchronization](#).

The multi sync splitter accessory meets your multi-device acquisition needs. By using digital synchronization cable(s), this accessory allows you to connect *biosignalsplex* devices with digital ports together to collect up to 24 channels at the same time (by connecting up to 3 HUBs together).



Figure 146: Multi sync splitter.

How to connect your accessory to your biosignalsplex

The multi sync splitter needs to be connected to the digital port of one of the devices that are being used for the multi-device acquisition (see [3.4 Synchronization](#)). When done, link the other device(s) with the multi sync splitter by connecting digital synchronization cable(s) (see [3.5.3](#)

Digital Synchronization Cable) to the digital port(s) of the other *biosignalsplex* devices and to the one of the three available inputs of the multi sync splitter.

How to configure synchronized acquisitions in OpenSignals (r)evolution

Open the *OpenSignals (r)evolution* device manager and select one of the devices you want to use for the multi-device acquisition. In the device's panel, click on the circle next to *SYNC* to turn on (blue circle) or off (empty circle) the synchronization function to synchronize the acquisition of your devices (see [Figure 147](#)). This function must be active to ensure synchronized multi-device acquisitions without signal drifts. Repeat this step for all devices that are used for signal acquisition.



Figure 147: Activate synchronization feature.

4 Add-Ons

OpenSignals (r)evolution add-ons are signal processing software add-ons that extract meaningful information from your acquired signals. All the available add-ons are split into two types of add-ons which process data during signal acquisitions or after signal acquisitions.

Real-time Add-ons:

Add-ons that process signals during real-time signal acquisitions. These add-ons are available in the real-time acquisition mode. At the time being, the following are the available add-ons of this type:

	Event Emulation		Event Annotation
	Signal Statistics		Goniometer
	Oximetry		Electromyography Analysis (EMG)
	Muscle Load		Force Platform
	Heart Rate Variability (HRV) <small>[New resource]</small>		fNIRS <small>[New resource]</small>
	Load Cell Force Analysis		Manual Measurement Input <small>[New resource]</small>
	Export analysis reports		

Post-processing Add-on:

Add-ons process signals after a finalized acquisition session. These add-ons are available in the signal visualization mode. At the time being, the following are the available add-ons of this type:

	Electrodermal Activity Events (EDA)		Event Annotation
	Electromyography Analysis (EMG)		Goniometer
	Heart Rate Variability (HRV)		Force Platform
	Respiration Analysis (PZT & RIP)		Signal Statistics
	Video Synchronization		Muscle Load
	Manual Measurement Input <small>[New resource]</small>		
	Export analysis reports		

Add-ons are not part of the free *OpenSignals(r)evolution software* and are either already included in some [biosignalsplux kits/wearables](#) or need to be purchased separately, with some exceptions such as the *Event Emulation* and the *Signal Statistics* add-ons, as they come free with *OpenSignals (r)evolution*. However, it is possible to test add-ons on time limited signals before purchasing them.

4.1 Add-on Purchase & Activation

Purchasable add-ons can be bought in the PLUX webstore which can be accessed via the following link:

<https://www.pluxbiosignals.com/collections/software-add-ons>

After purchasing your add-ons and receiving the order confirmation, restart *OpenSignals* and you will now be able to use the add-on(s) in the live acquisition screen (real-time add-ons) and/or playback mode (post-processing add-ons).

NOTE

The **add-on activation process requires a stable internet connection**, as *OpenSignals (r)evolution* needs to communicate with the PLUX servers. Add-ons will work without an internet connection after a successful activation.

NOTE

Add-ons are linked to your PLUX device. Therefore, it is possible to use them on any computer your device is connected to. For this, connect your device to the computer you would like to use and configure it in *OpenSignals (r)evolution* accordingly to the instructions in section *2.2.4 Device Configuration*.

Add-ons which are included in kits are already activated by default and ready-to-use when you receive them; if this is not the case please contact our team (see *9.3 Contact & Support*).

4.2 Listing & Using Add-ons

Normally, all add-ons are activated and listed by default after a successful purchase. However, if your add-on is not listed in the signal visualization mode you have to activate the add-on the help interface (see Figure 148 and section 1.5.4 *Activating & Deactivating Add-ons*).

? Enable add-ons in the help interface.

🔌 Allows you to access the available add-ons on the signal acquisition or visualization mode.

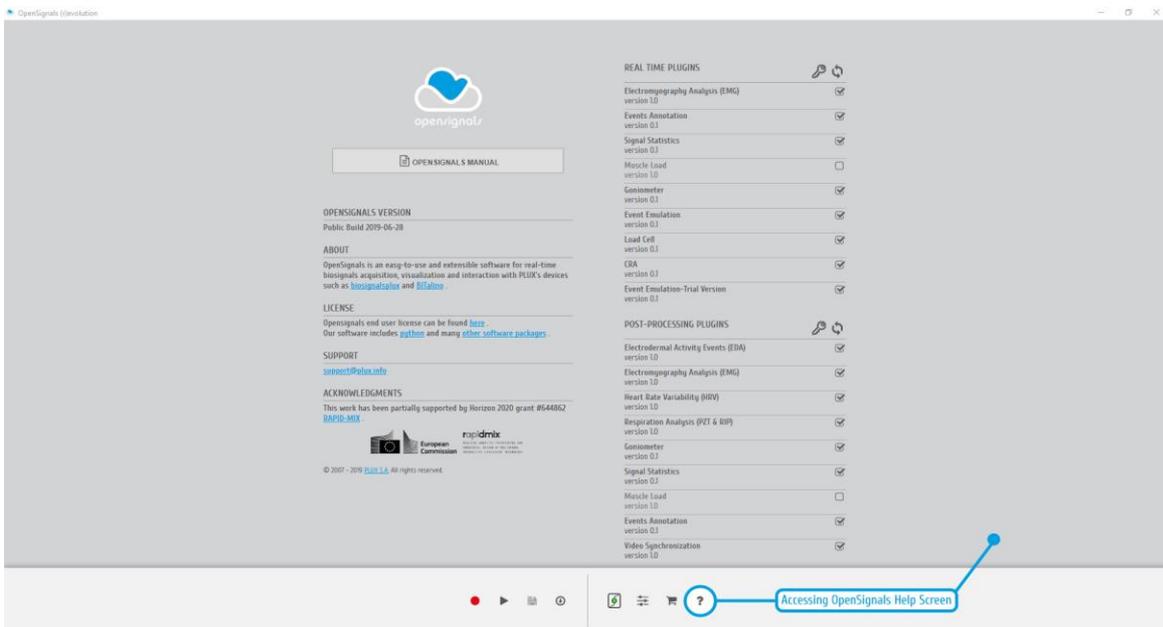


Figure 148: Activate add-ons in the help interface

Activated add-ons can then be selected in the respective modes. Real-time add-ons can be used in the signal acquisition mode (see 3.1 *Signal Acquisition Mode*), while visualization add-ons can be used in the signal visualization mode (see 3.2 *Signal Visualization Mode*). Click on the add-on of your choice to start using it.

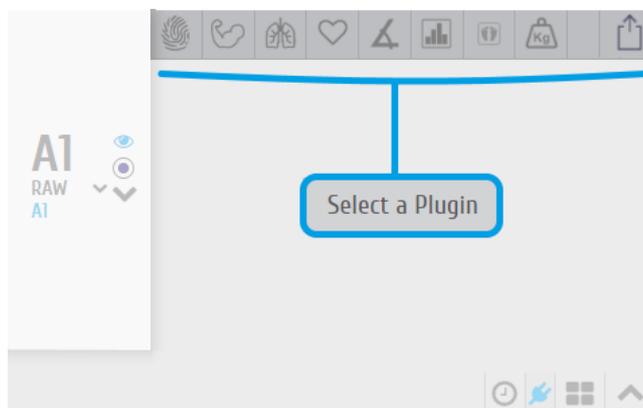


Figure 149: Add-on selection in the visualization mode.

4.3 Add-on Reports



Export generated add-on results into PDF or CSV file.

The results generated by *OpenSignals (r)evolution* add-ons can be exported to a PDF or a CSV file containing all the generated data. While PDF files are useful to display all the acquired signal processing results including diagrams, CSV files allow you to export all your files into a readable format which can then be imported into 3rd party software.

The screenshot shows a dialog box titled "EXPORT OPTIONS". It contains several input fields and a list of plugins. The "TITLE" field is set to "Report", "AUTHOR" to "PLUX", and "DATE/TIME" to "Thu Jul 11 2019 11:48:29 GMT". There is a checked checkbox for "CURRENT". The "FORMAT" dropdown is set to "PDF". Under "PLUGINS", "Electrodermal Activity Events (EDA)" is unchecked, "Electromyography Analysis (EMG)" is checked, "Respiration Analysis (PZT & RIP)" is unchecked, and "Heart Rate Variability (HRV)" is unchecked. The "FILENAME" field is set to "OpenSignals (r)evolution Report". At the bottom are "CANCEL" and "OK" buttons.

Figure 150: PDF report window.

FILE PROPERTIES

Insert the title & author information of the current acquisition and processing session into the *TITLE* and *AUTHOR* fields. This information will be displayed on the first page of the PDF file.

The first page will also contain general information about the selected *OpenSignals* file, information about the devices used during the acquisition and information about the acquisition configuration containing the sampling rate, sampling resolution and active channels.

ADD-ONS

Select the add-ons for which the generated results should be exported into the PDF/CSV file. Make sure you have used the selected add-ons before processing your signals, otherwise the generated PDF/CSV files will list no information about the selected and non-used plugins.

FORMAT

The export function offers the export of files into PDF or CSV file. The PDF file will contain all the add-on generated results and diagrams as well as general information about the acquisition settings (e.g., *OpenSignals (r)evolution* version, date time, duration, used devices, etc.). The CSV file will only contain the generated data without the diagrams and all the general information. In this case, the CSV contains headers which are placed within curved parenthesis ({...}) providing general information about processed file/signal and about the extracted values. These values are always listed right below a given header ({...}).

FILENAME & PATH

Select the preferred filename of the new file and navigate until the destination directory.

4.4 Add-On Processable Devices and Channels

All *OpenSignals (r)evolution* add-ons provide a way to specify the device and channels to be processed, as demonstrated on the following topics.

Additionally, the researcher can limit the processing stage to the desired time-window.

DEVICE & CHANNEL

Select the MAC-address of the device that was used for the signal acquisition. If only one device has been used for signal acquisition, the MAC-address of the used device will automatically be selected (several MAC-addresses will be displayed if multiple devices were used during the acquisition session). When done, select one of the available channel options (“PROCESS APPLICABLE CHANNELS” or “PROCESS ALL VISIBLE CHANNELS”) in order to start the processing stage on the pretended channels.

Active channels are all channels which have not been hidden and are therefore visualized in the signal visualization mode (see sections 3.2 *Signal Visualization Mode*).

PROCESSING INTERVAL

The processing interval selects the time interval that is to be processed. It can be changed by manually writing the start and end of the acquisition session using the time fields in the configuration window or by changing the size of the visualization interval in the overview window.



Figure 151: Configurable options regarding the device and channels to process on HRV analysis add-on.

4.5 Event Emulation



Event emulation

DESCRIPTION

The event emulation add-on allows you to train and test a classifier. During the testing phase, the classification model returns, in real-time, the class to each the current acquired data should belong (prediction). Each class can also trigger a previously configured keyboard/mouse event.

This add-on is an online add-on. It is available in the signal acquisition mode (see [3.1 Signal Acquisition Mode](#)).

This add-on is included in *OpenSignals (r)evolution* and does not have to be purchased additionally.

4.5.1 Hardware Configuration

This add-on works with every kind of signal being acquired with PLUX devices. It is not restricted to any specific sensor.

4.5.2 Add-on Configuration

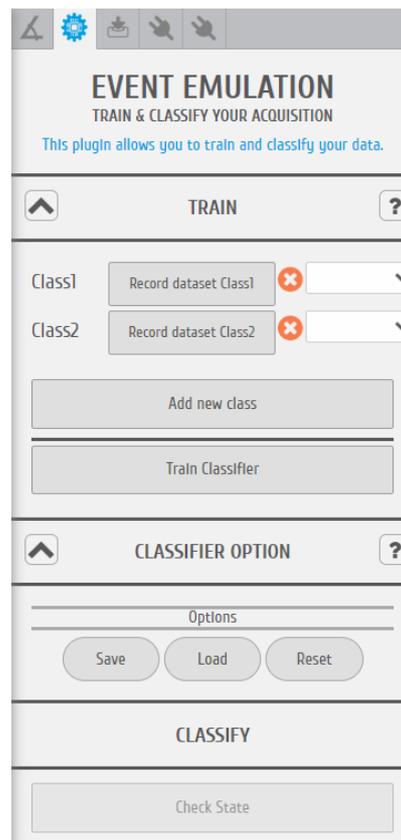


Figure 152: Event emulation add-on.

DEVICES & CHANNELS

The classification methodology will be applied to all active channels.

COLLECTING DATA FROM EACH CLASS

Accordingly to the user application, multiple classes can be defined, in our practical example a binary classifier will be trained, using data from a triaxial accelerometer.

The classifier should be able to separate two classes: **a)** up and down movements and **b)** lateral movements.

Transmitting knowledge to this system is an essential stage. To achieve this purpose, training data of each class should be provided to the machine-learning algorithm, which can easily be done by clicking on the respective “Record dataset Class<x>” button.

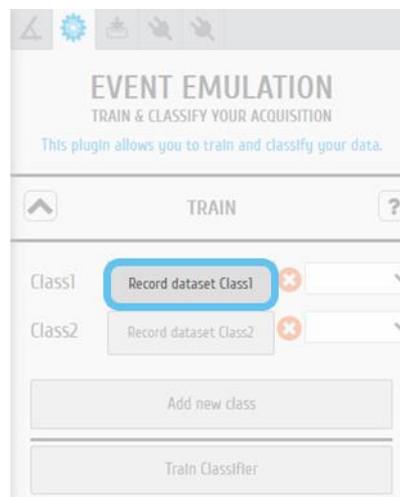


Figure 153: Event emulation add-on | button that triggers the training stage.

After pressing the button, during a few seconds the algorithm is collecting training data from the active channels, in our specific case, during these seconds we should execute “up and down” movements for class 1 and “lateral movements” for class 2.

EMULATING EVENTS

When the training stage is completed the event emulation can be configured, linking a keyboard/mouse event to each class.

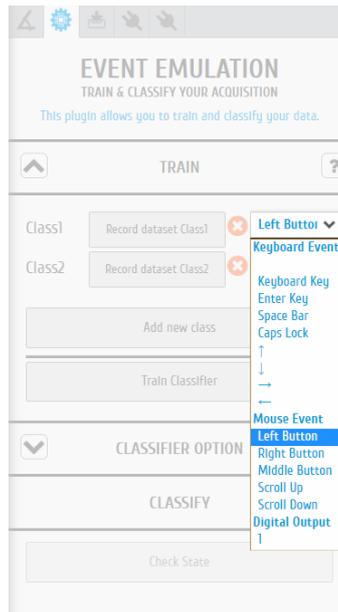


Figure 154: Event emulation add-on | definition of keyboard/mouse events.

ADD NEW TRAINING CLASSES

New classes can be added to the classification system, operation that is achievable through the “Add new class” button.

CLASSIFIER TRAINING STAGE

After collecting training data, the classification model needs to fit to these data.

In order to officially start the fitting procedure, the user should click on “Train Classifier” button at the bottom zone of the add-on graphical interface.

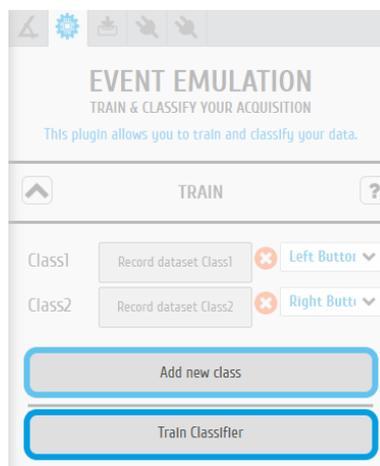


Figure 155: Event emulation add-on | highlighting Add new class and Train Classifier buttons.

CLASSIFICATION OUTPUTS

With the entire configuration concluded the testing phase can begin through a click on “Check State” button, action that will trigger the generation of classification results (predictions).

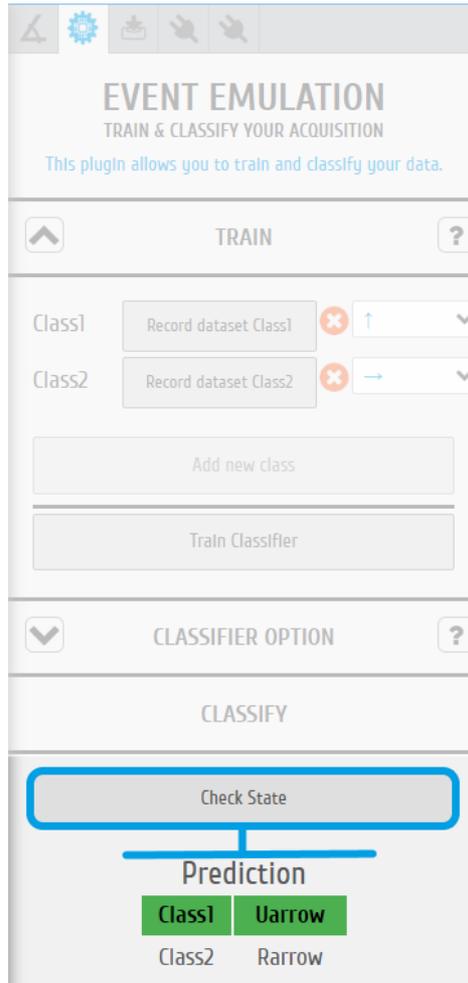


Figure 156: Event emulation add-on | classification results/predictions.

4.6 Event Annotation



Event Annotation

DESCRIPTION

While acquiring/collecting physiological data it becomes extremely relevant to annotate specific events that occur during signal acquisition.

As a practical example, researches where EDA data is collected require a precise annotation of the stimulus time instant, i.e., the event responsible for the electrodermal response.

Using the Event Annotation add-on, the researcher can annotate this type of events with a simple button click, through the computer keyboard, something essential during the post-acquisition analysis stage.

This add-on is both an online and a post-acquisition add-on. It is available in the signal visualization mode (see 3.2 *Signal Visualization Mode*) and in the signal acquisition mode (see 3.1 *Signal Acquisition Mode*).

4.6.1 Hardware Configuration

This add-on works with every kind of signal being acquired with PLUX devices. It is not restricted to any specific sensor.

4.6.2 Add-on Configuration

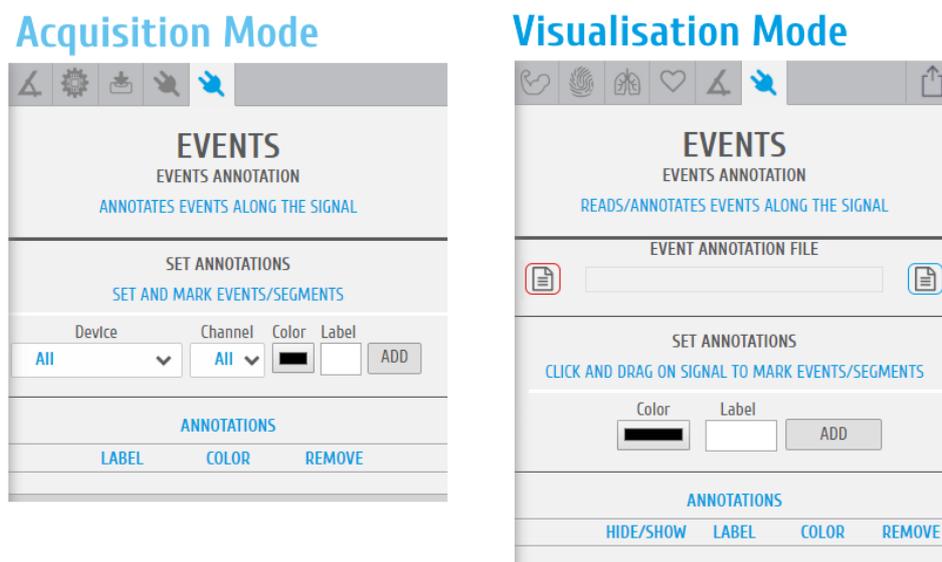


Figure 157: Event Annotation add-on (acquisition mode @left and visualization mode @right).

ACQUISITION MODE

DEVICES & CHANNELS

The selection of the devices and channels where the processing algorithms will be executed, is made on the add-on interface (Figure 158).

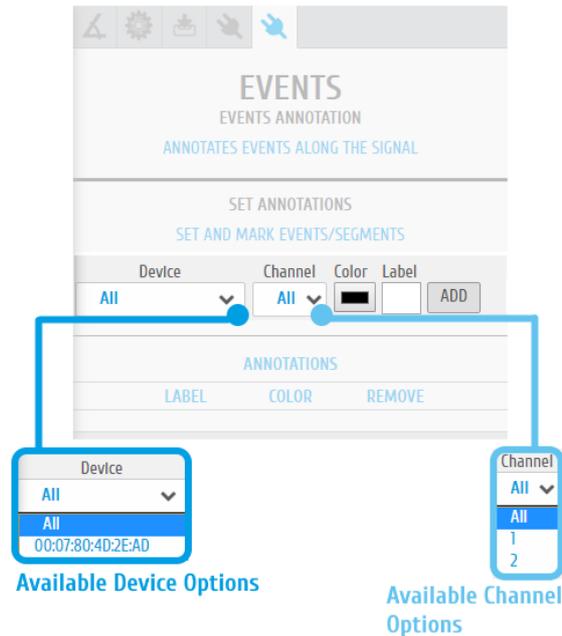


Figure 158: Event Annotation add-on | device and channel selection [illustrative example]

It is possible to apply signal annotation events to **all devices + all channels** or only in a single channel.

EVENT COLOR AND LABEL

After defining the device and channels to which the current event will be applied a **color** and **label/keyboard key** can be specified.

Regarding the **label/keyboard key**, this will be keyboard key that when pressed it triggers the event annotation, while the color defines the graphical appearance of the event bar, drawn at the signal acquisition screen when the event is triggered (keyboard button click).

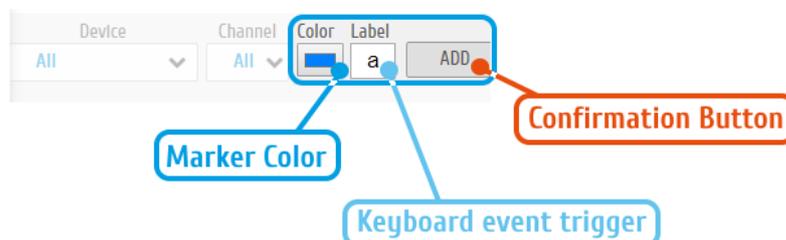


Figure 159: Event Annotation add-on | configuration panel

To apply the previous “annotation” the configuration button **ADD** should be clicked.

With the last configuration (Figure 159) every time the keyboard “a” button is clicked a blue marker will be drawn in “All” devices and “All” channels (Figure 160).

Additional events can be programmed. In order to achieve this multiplicity, the enumerated steps should be repeated, choosing a different keyboard trigger.



Figure 160: Event Annotation add-on | graphical representation of the annotated event

Stopping the real-time acquisition will cause the generation of an additional .txt file where the registered events are stored (this file is in the same location where the acquired file is stored or in the predefined *OpenSignals (r)evolution* folder of the user computer).

VISUALISATION MODE

The real-time annotated events can be observed inside the Visualisation Mode of *OpenSignals (r)evolution*.

The user only needs to open the annotation file, as an example, on Windows systems it should be available at:

"C:\Users

After this loading stage the annotation data can be visualised by simply clicking on "DRAW ANNOTATIONS".

On the other hand, to make a backup of the Annotation file and start creating a new one from scratch (containing only the annotations added in the Visualization Mode), the "Reset Annotation File" button should be pressed.

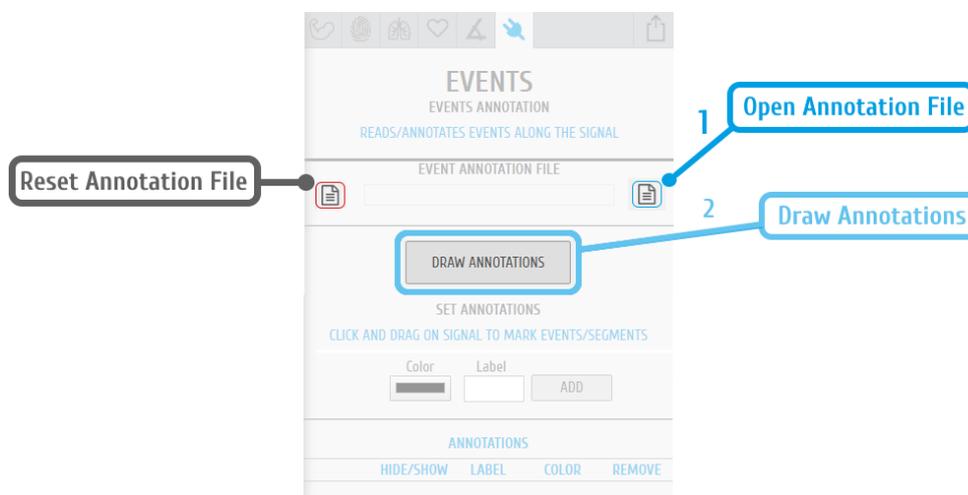


Figure 161: Event Annotation add-on | graphical representation of the annotated event [visualisation mode]

An offline event can also be added to the acquired data (steps highlighted on Figure 162).

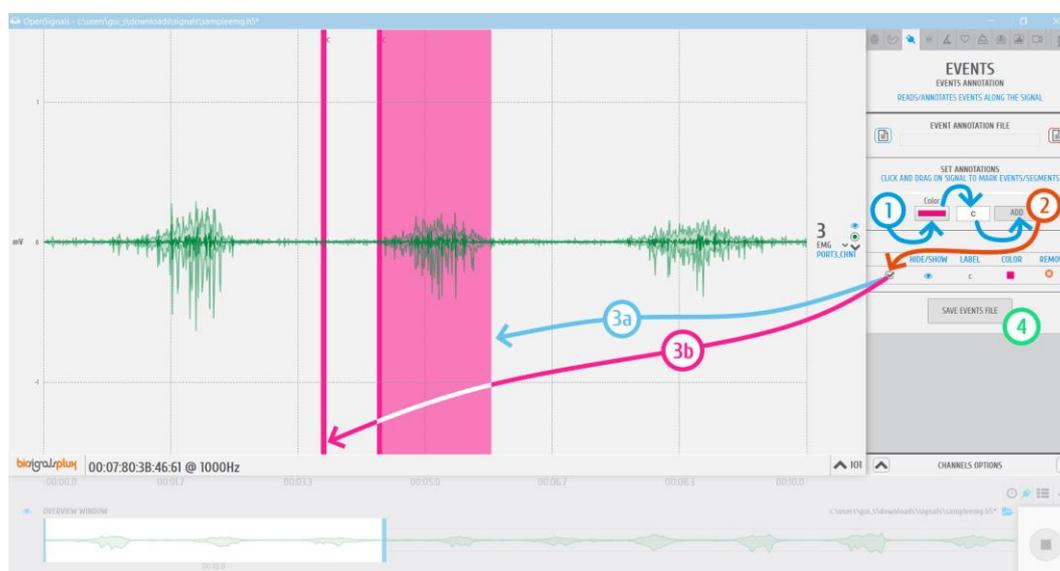


Figure 162: Event Annotation add-on | adding offline events [visualisation mode] by 1) configuring an annotation; 2) checking/selecting the added annotation in the annotation list; 3) draw an event through 3a) mouse drag or 3b) mouse-click and finally 4) saving the new events into the annotations file.

4.7 Goniometer



Goniometer

DESCRIPTION

PLUX's Goniometer (<https://www.pluxbiosignals.com/products/goniometer-gon>) requires a proper calibration procedure in order to produce precise results adapted to the experimental methodology of the user research.

To facilitate this procedure, an auxiliary add-on is available to the users, providing an easy way to calibrate the Goniometer.

This add-on is both an online and a post-acquisition add-on. It is available in the signal visualization mode (see 3.2 *Signal Visualization Mode*) and in the signal acquisition mode (see 3.1 *Signal Acquisition Mode*).

4.7.1 Hardware Configuration

The goniometer add-on only processes sensor data acquired with Goniometer sensor:

biosignalsplux Goniometer sensor:

<https://www.pluxbiosignals.com/products/goniometer-gon>

How to configure your sensor in OpenSignals (r)evolution

Open the *OpenSignals (r)evolution* device manager and select the device to which you have your Goniometer sensor connected to.

In the device's panel, activate the two channels where the Goniometer cables are attached (in our case we choose channels 1 and 2).

After the activation procedure (ensured with a click on the "channel activation" button which should become blue) it is necessary to define the type of sensor as "GONIO", accordingly with the illustration of Figure 163.

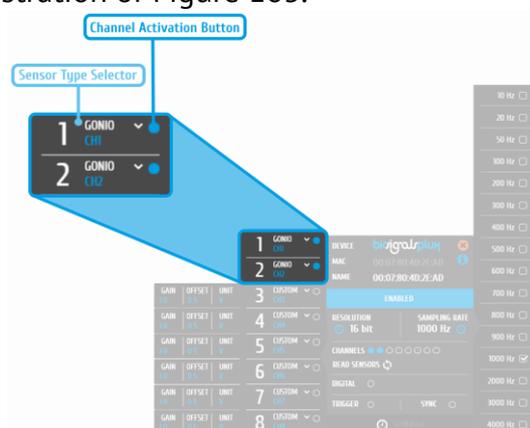


Figure 163: Channel activation and sensor type selection.

4.7.2 Add-on Configuration

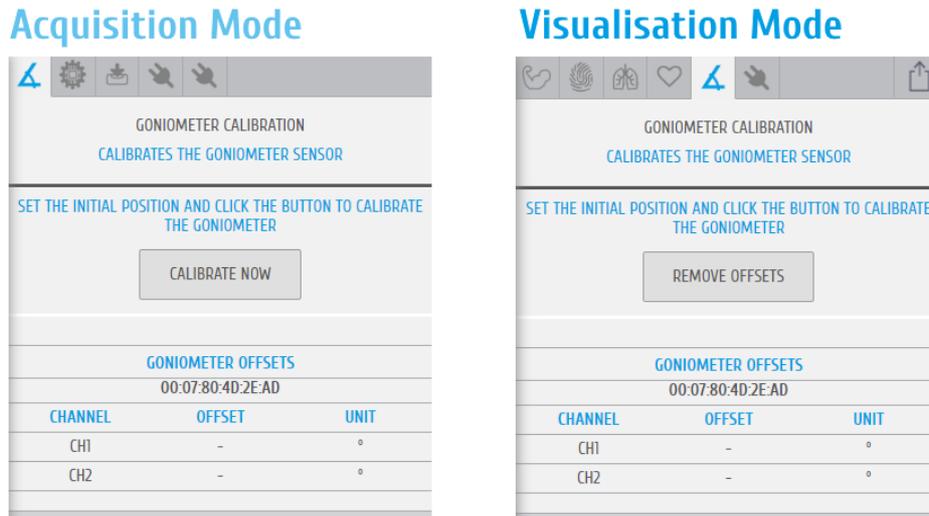


Figure 164: Goniometer Add-on (acquisition mode @left and visualization mode @right).

ACQUISITION MODE

DEVICES & CHANNELS

The selection of the devices and channels is automatically executed by the add-on, accordingly to the sensor label, configured prior to the acquisition, as demonstrated on Figure 163.

CALIBRATION PROCEDURE

In order to ensure an effective calibration procedure, the goniometer cells should be oriented in the **0 degrees configuration** (generally the 0 degrees configuration will be similar to the one presented at Figure 165, but the researcher can customize it accordingly to the desired application).



Figure 165: Goniometer | common 0 degrees orientation

After the previous experimental setup, it will only be necessary to click on "Calibrate Now" and the offset values will automatically appear at the add-on interface and they will also be stored inside the metadata of .h5 file.

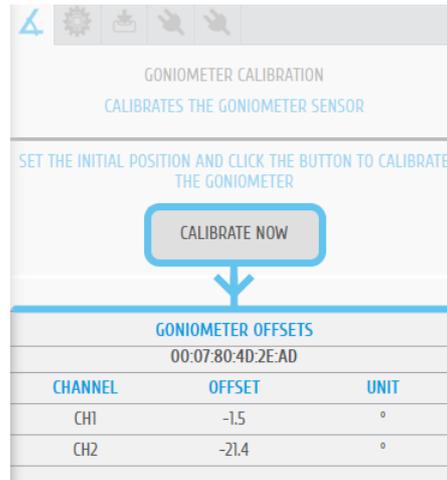


Figure 166: Goniometer add-on | calibration button

With these steps the Goniometer is calibrated and ready to use.

VISUALISATION MODE

The extracted calibration offsets, during the real-time acquisition stage, are automatically stored inside the .h5 file metadata.

So, when a .h5 file containing previously calibrated goniometer data is loaded in the visualisation mode of *OpenSignals (r)evolution* the offsets can easily be applied to the signals pressing the "REMOVE OFFSETS" button in the add-on interface.

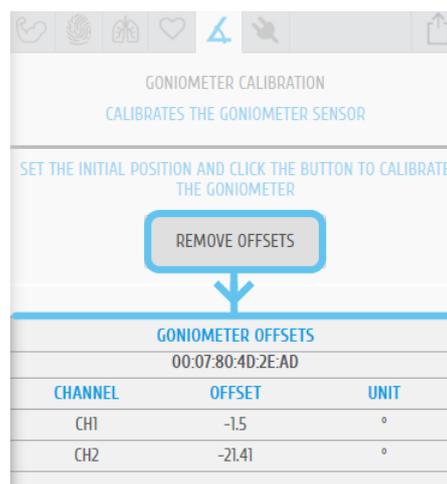


Figure 167: Goniometer add-on | get offset values

With this procedure the researcher can visualise the acquired data in a calibrated format and start the execution of offline processing algorithms.

4.8 Signal Statistics



Signal statistics extracts statistical time-based information from all active channels.

DESCRIPTION

This add-on extracts time-based features from all active acquiring channels in real-time. It provides useful information such as the minimum and maximum values of each channel as also the mean, median, root mean square values and the residual sum of squares.

This add-on was originally a real-time add-on, available in the signal acquisition mode (see 3.1 *Signal Acquisition Mode*).

However, with a recent *OpenSignals (r)evolution* update, the **Signal Statistics** add-on functionalities are also available on the signal visualization mode (3.2 *Signal Visualization Mode*).

4.8.1 Hardware Configuration

This add-on works with every kind of signal being acquired with PLUX devices. It is not restricted to specific sensors.

4.8.2 Add-on Configuration

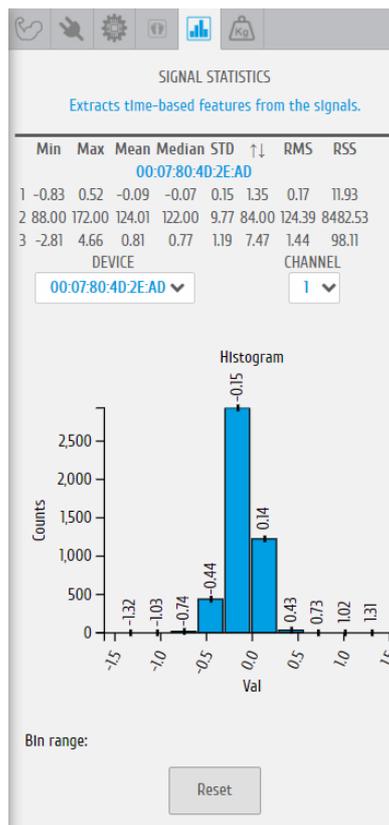


Figure 168: Signal statistics add-on.

This add-on does not require any configuration for the signal acquisition mode version. It will automatically provide statistical information in a table, as soon as it has been selected in the signal acquisition mode for all actively acquiring channels/devices. The respective channel number is displayed in the first column of the table (here: 1,2,3).

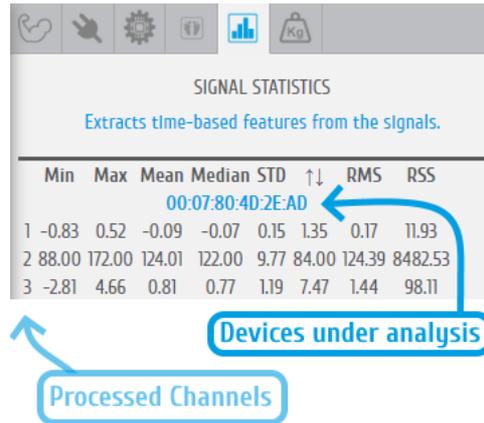


Figure 169: Signal statistics add-on | devices and channels under analysis.

This add-on provides the minimum and maximum values of each channel as also the mean, median, root mean square values and the residual sum of squares.

Additionally, a histogram with the distribution of sample values can be visualized for a device/channel configuration at once.

Regarding the signal visualization version, in order to get results from the **Signal Statistics** plugin, it should be selected the desired time segment to be processed and the user should trigger the statistical algorithm by clicking on “PROCESS” button.

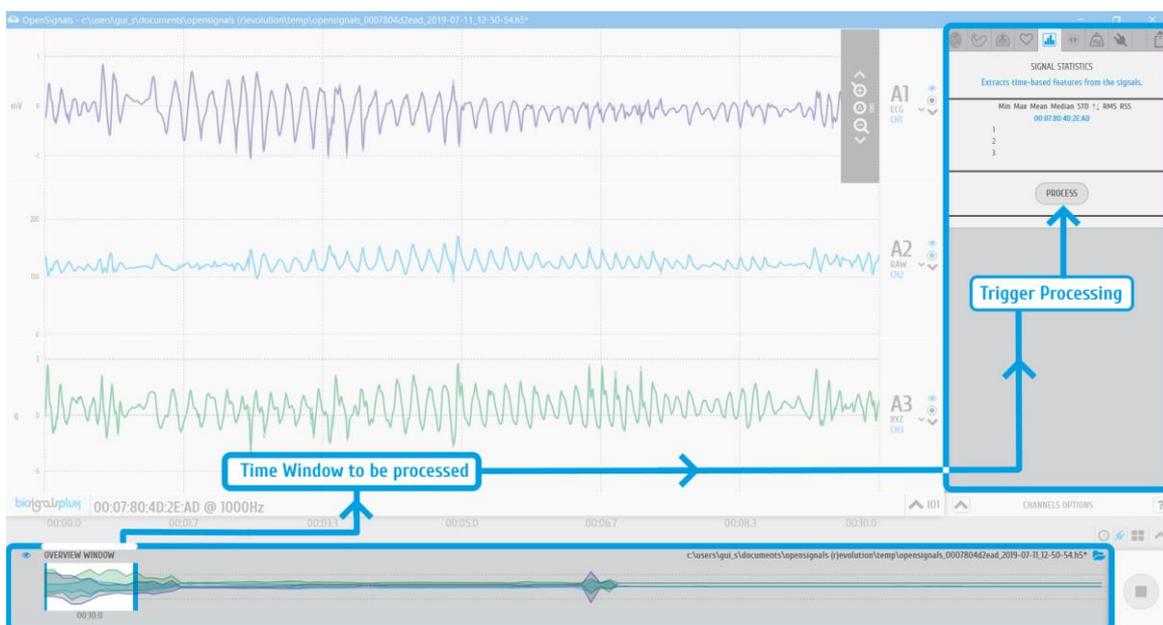


Figure 170: Signal statistics add-on | triggering signal processing through the interface clickable button.

4.9 Oximeter

 Computes and displays the oxygen saturation and beats per minutes.

DESCRIPTION

This add-on uses SpO2 sensor data from the *biosignalsplux* SpO2 sensor to estimate the saturation level in the blood with +/- 2% accuracy compared to a medical sensor. Additionally, this add-on analyses all the acquired sensor data and provides BPM values in real-time.

This add-on is a real-time add-on. It is only available in the signal acquisition mode (see *3.1 Signal Acquisition Mode*).

This add-on is not included in *OpenSignals (r)evolution* and must be purchased additionally.

4.9.1 Hardware Configuration

The oximeter add-on processes sensor data acquired with one of the following sensors:

biosignalsplux SpO2 sensor:

<https://www.pluxbiosignals.com/products/blood-oxygen-saturation-spo2-versatile-sensor>

biosignalsplux SpO2 sensor with finger clip:

<https://www.pluxbiosignals.com/products/blood-oxygen-saturation-spo2-sensor>

Firmware requirements

The SpO2 sensor requires a specific *biosignalsplux* firmware version (v.3.7) to work. Please follow the instructions in chapter 7.2 to learn how to check the current firmware version of your device and to upgrade the firmware to version 3.7 if another version is currently being used on your device.

You can download the correct firmware here (v.3.7 for SpO2 & fNIRS) or directly through *OpenSignals (r)evolution* as described in *2.2.4.11 Device Info*:

https://www.downloads.plux.info/OpenSignals/misc/user-manual/biosignalsplux_FW_3.7_Installer.zip

How to connect your sensor to your biosignalsplux

The SpO2 sensor has to be connected to the **ground port** of your device. Connecting this sensor to an analog input of your device will not work.

How to configure your sensor in OpenSignals (r)evolution

Open the *OpenSignals (r)evolution* device manager and select the device to which you have your SpO2 sensor connected to.

In the device's panel, click on **READ SENSORS** button and wait until two extra channels appear, typically they will be labelled as **9A** and **9B**, however, in the following figure they are presented as channel **9** and **10**.

Then set up channel 9 for signal acquisition, set the port (P) to 9 and the channel (C) to 1 (Figure 171) and select one of the available SpO2 options in the channel's dropdown menu (if the correct option was not assigned with the **READ SENSORS** command).

The different options are designed for different placement areas of the SpO2 sensor. We recommend selecting the available configuration that fits best your needs. The following configurations are available:

- *SPO2.ARM* for applications where the sensor is placed on the subject's arm
- *SPO2.FING* for applications where the sensor is placed on the subject's finger
- *SPO2.HEAD* for applications where the sensor is placed on the subject's head

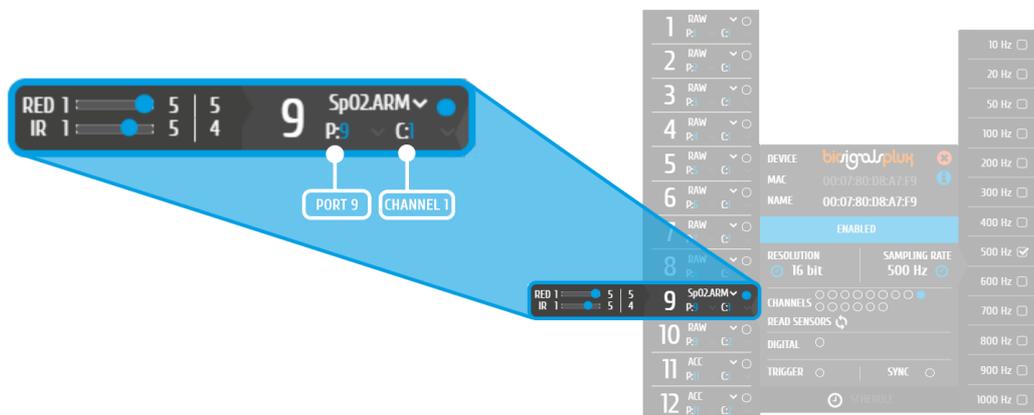


Figure 171: Port and channel number for SpO2 sensor.

4.9.2 Add-on Configuration

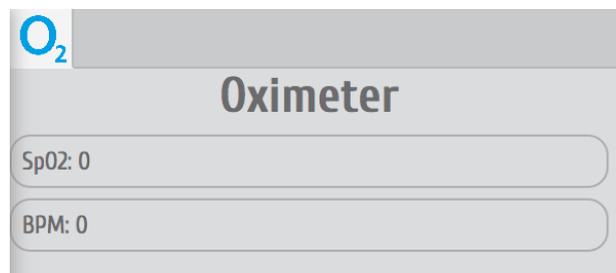


Figure 172: Oximeter add-on.

This add-on does not require any configuration. It will automatically provide the estimated oxygen saturation level in the *SpO2* field and the calculated BPM in the *BPM* field in real-time based on sensor data of the SpO2 sensor.

4.10 Muscle Load



Online and offline muscular load analysis add-on for ergonomic applications

DESCRIPTION

This add-on evaluates the muscular load that the muscles are subjected during a normal work day in real-time and after signal acquisition sessions. It measures the static, median and high intensity levels and displays the amplitude probability density function which helps detecting whether the worker is working within ranges that are safe or risky.

This add-on works both offline and online. It is available in the signal visualization mode (see *3.2 Signal Visualization Mode*) and in the signal acquisition mode (see *3.1 Signal Acquisition Mode*).

This add-on is included in the [biosignalsplux Professional](#), [biosignalsplux ergoExplorer](#) and [biosignalsplux ergoResearcher](#) kits. It can also be bought in addition to all *biosignalsplux* and *BITalino* devices.

4.10.1 Hardware Configuration

The Muscle Load add-on processes EMG sensor data acquired with one of the following sensors:

biosignalsplux EMG Sensor:

<https://www.pluxbiosignals.com/products/electromyography-emg>

biosignalsplux MuscleBAN:

<https://www.pluxbiosignals.com/products/muscleban-kit>

BITalino EMG Sensor:

<https://www.pluxbiosignals.com/products/electromyography-emg-sensor>

The channel(s) you have one of these sensors connected to have to be configured properly before starting the acquisition. To do so, follow the device configuration instructions as described in section *2.2.4 Device Configuration*. During the configuration process make sure to select the channel your EMG sensor is connected to and to set the channel configuration to *EMG* (see *2.2.4.6 Channel Configuration*).

4.10.2 Add-on Configuration

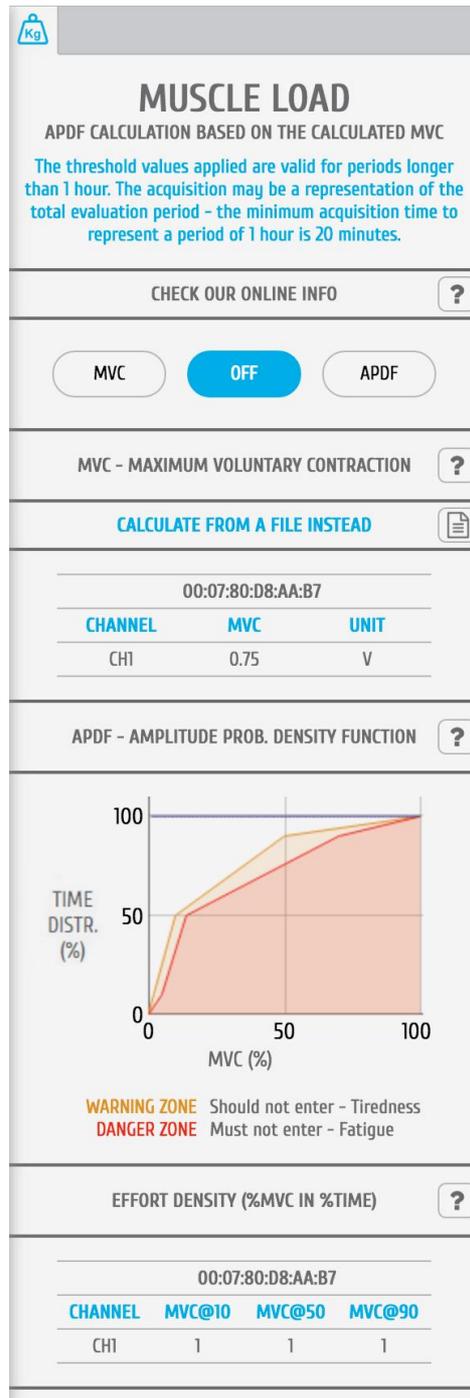


Figure 173: Muscle Load add-on.

DEVICES & CHANNELS

For selecting the devices and channels where the processing algorithms will be executed, we recommend the reading of section 4.4

Make sure you have followed the instructions on 4.10.1 Hardware Configuration for the add-on to be able to recognize your EMG channels.

MVC – MAXIMUM VOLUNTARY CONTRACTION

At the beginning of each acquisition it is necessary to calibrate the add-on, ensuring that it provides reliable data. To do so, click on the *MVC* button (maximum voluntary contraction) and let the subject contract the muscle which is being monitored to its maximum (**real-time approach**). Make sure the muscle can be contracted without any restrictions for the calibration to be done properly. This contraction will provide the value of the maximum possible contraction of the muscle. This value will be the reference point for the signal acquisition and processing. Each channel's MVC value are listed in the table below *MVC – MAXIMUM VOLUNTARY CONTRACTION*.

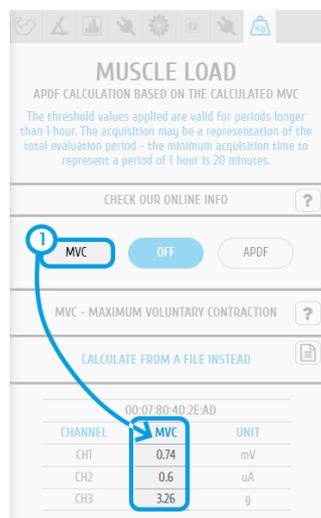


Figure 174: Real-time MVC value.

The maximum voluntary contraction (MVC) will be calculated for each channel based on the EMG maxima of your selected EMG signals.

While using the **Muscle Load** plugin in the visualization mode, for estimating the MVC value you should select a time-window where the calibration protocol was executed in the overview window (for more details, please, check section 3.1.3.5). After this specification the “MVC” button should be clicked and the calibration is completed.

The MVC acquisition can also be done in a separate signal acquisition session/file. This can be useful to use when doing several signal acquisitions to ensure that the analysis of all files and signals are done with the same reference values – the MVC values.

For this, start an acquisition and acquire your data which are being used as MVC values for later analysis and save your acquisition. To use the MVC data from this acquisition in another acquisition, click on the document icon in the *CALCULATE FROM A FILE INSTEAD* field and select the *OpenSignals (r)evolution* file which contains the MVC signal data. *OpenSignals (r)evolution* will then calculate the MVC value from the selected file.



Select *OpenSignals (r)evolution* file to calculate MVC values from file

APDF – AMPLITUDE PROBABILITY DENSITY FUNCTION

After acquiring the MVC, click on the APDF button to start the amplitude probability density function. This function calculates the cumulative percentage histogram of the enveloped data in terms (%) of the MVC and displays the muscle load that the muscle(s) are subjected during the current task (represented by the blue lines in the graph), identifying if this task is causing muscle tiredness (see *WARNING ZONE* area in Figure 173) or muscle fatigue (see *DANGER ZONE* area in in Figure 173). Low to no muscle load will shift the blue line to the top left corner of the graph, increasing muscle load and muscle load over duration will shift the graph into the warning and danger zones.

EFFORT DENSITY (%MVC IN %TIME)

This function displays the cumulated effort in terms (%) of the MVC at 10, 50 and at 90% of the time (blue lines on chart) and displays all values of all devices and sensors in a table.



Figure 175: APDF function and Effort Density results.

4.11 Electromyography (EMG) Analysis



Electromyography Analysis

DESCRIPTION

Muscle activity is usually assessed using temporal and spectral features. With this plugin, you will be able to extract useful statistical information from Electromyography (EMG) sensor data. Its automatic onset detection algorithm enables the analysis of each individual muscle activation event, in addition to the overall analysis of the recording session. Timings analysis can also be done for each activation relative to a reference muscle activation.

Originally, EMG analysis was achieved exclusively on an offline add-on for post processing purposes.

However, since a recent release of *OpenSignals (r)evolution* (August of 2019), it is also available in the signal acquisition mode (see 3.1 Signal Acquisition Mode).

This add-on is included in the [biosignalsplux Professional](#) and [biosignalsplux ergoResearcher](#). It can also be bought in addition to all *biosignalsplux* and *BITalino* devices.

4.11.1 Hardware Configuration

The EMG analysis add-on processes sensor data acquired with one of the following sensors:

biosignalsplux EMG Sensor:

<https://www.pluxbiosignals.com/products/electromyography-emg>

biosignalsplux MuscleBAN:

<https://www.pluxbiosignals.com/products/muscleban-kit>

BITalino EMG Sensor:

<https://www.pluxbiosignals.com/products/electromyography-emg-sensor>

The channel(s) to which you have one of these sensors connected to have to be configured properly before starting the acquisition. To do so, follow the device configuration instructions as described in 2.2.4 *Device Configuration*. During the configuration process make sure to select the channel your EMG sensor is connected to and to set the channel configuration to *EMG* (see 2.2.4.6 *Channel Configuration*).

4.11.2 Add-on Configuration

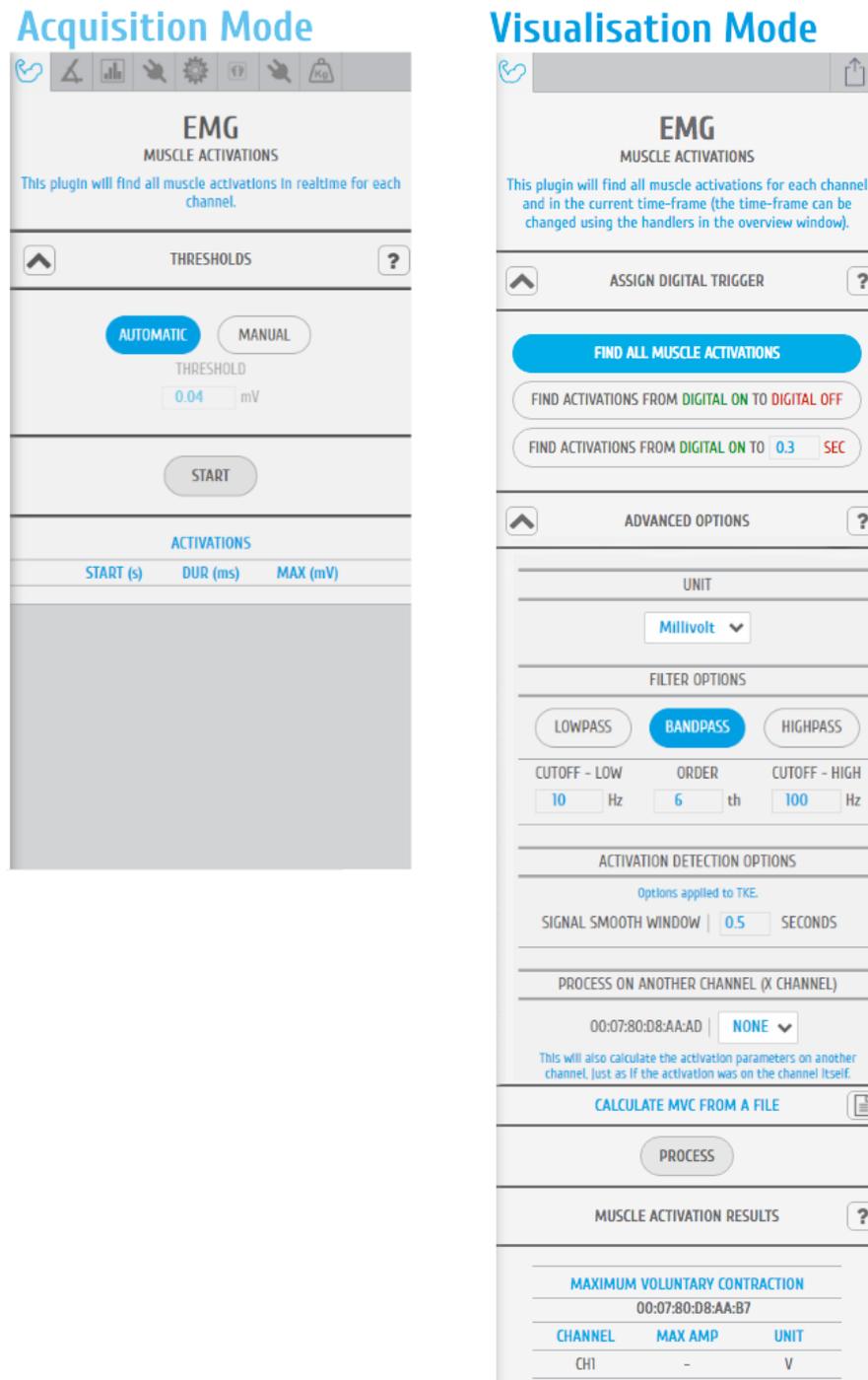


Figure 176: EMG add-on (acquisition mode @left and visualization mode @right).

ACQUISITION MODE DEVICES & CHANNELS

For selecting the devices and channels where the processing algorithms will be executed, we recommend the reading of section 4.4

This specification can be defined at the bottom tab of the plugin interface, before clicking on the “START” button.

THRESHOLDS

After clicking on the “START” button the algorithm responsible for detecting muscular activations starts to run.

There are two available modes of function, i.e., it can execute their internal code without any interference/input from the user (AUTOMATIC threshold option) or it can act accordingly to the user specification/fixed threshold (MANUAL threshold option), where all sample values above the defined threshold will be considered as a potential start of a muscular activation period.

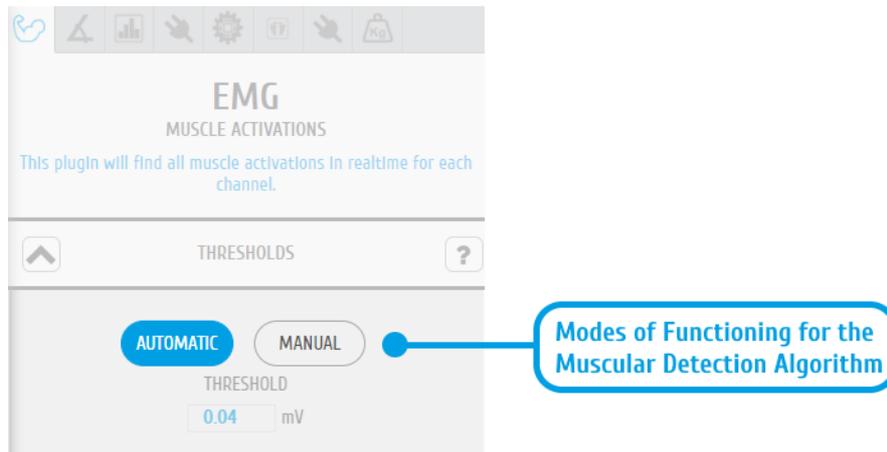


Figure 177: Section containing the inputs that will define the mode of functioning of the muscular detection algorithm.

MUSCLE ACTIVATION RESULTS

The results produced by the EMG plugin consists in the duration and maximum amplitude of each detected muscular activation, presented into a textual format, through a dynamic table and also in a graphical way with the muscular activation being highlighted by the application through a translucent rectangle in the signal acquisition section of *OpenSignals (r)evolution* interface.

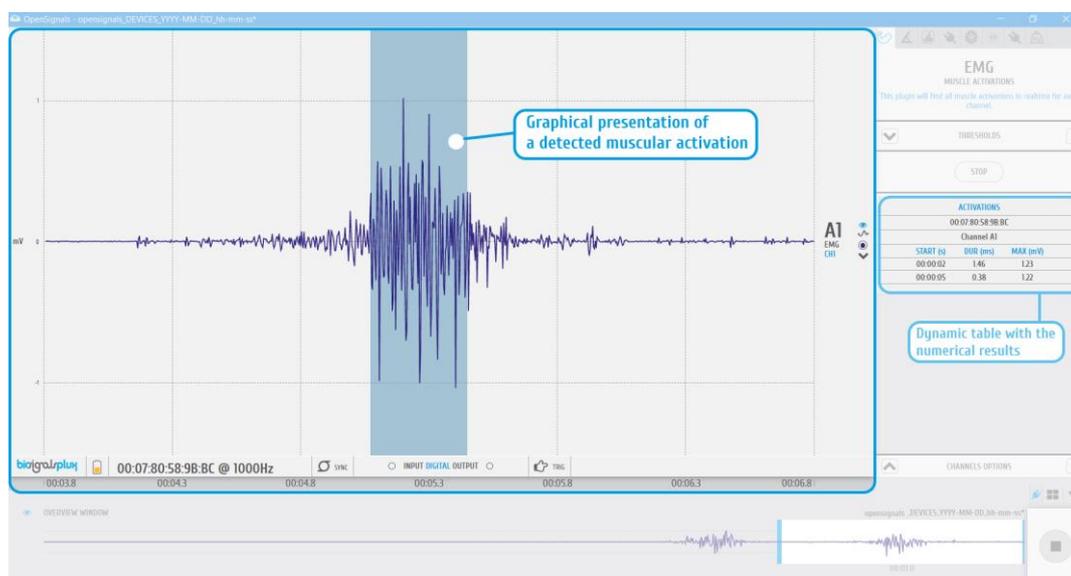


Figure 178: EMG plugin | numerical and graphical results

VISUALISATION MODE

DEVICES & CHANNELS

The EMG analysis add-on offers the possibility to process all available EMG channels (hidden and visible channels) or all visible channels of the acquisition. Note that processing all visible channels will also process non-EMG sensor data, however, this can be useful to process RAW EMG channels.

Additionally, it can be selected when the channels units should be set to microvolt or millivolt in the *UNIT* dropdown menu (Figure 179).

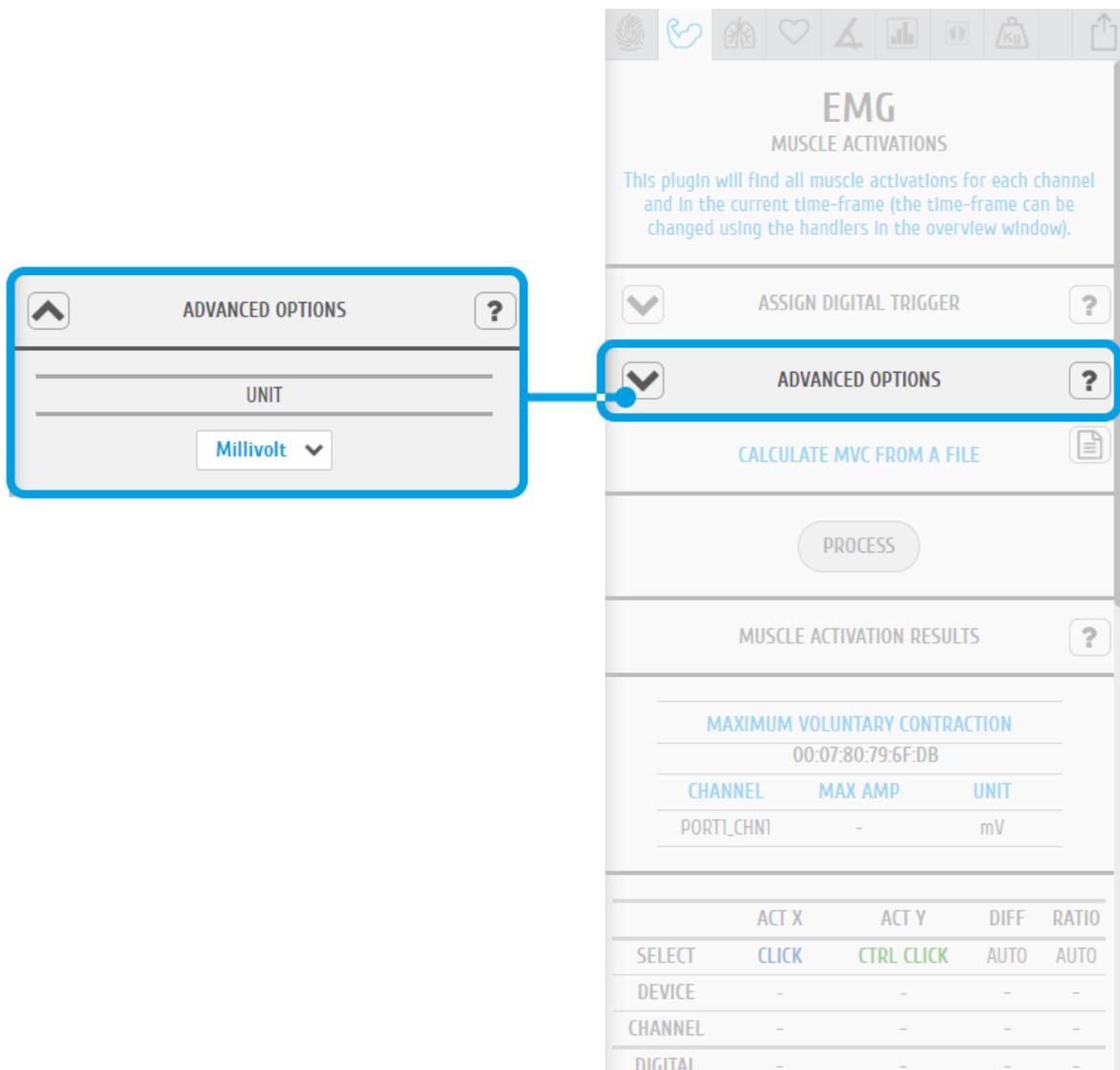


Figure 179: EMG plugin | unit specification

PROCESSING INTERVAL

In order to specify the processing time-window/interval, we recommend the reading of section 4.4

ASSIGN DIGITAL TRIGGER

In the *ASSIGN DIGITAL TRIGGER* option it is possible to select if the add-on should find all available muscle activations or if it should find activations depending on digital input and output signals, i.e. trigger signals.

The *FIND ACTIVATIONS FROM DIGITAL ON TO DIGITAL OFF* option will find all activations within the time frames where digital inputs have been received, for example, during the time frame a pushbutton has been being pressed (see also *3.5.1 Trigger*).

The *FIND ACTIVATIONS FROM DIGITAL ON TO ... SEC* will find all activations within the specified duration after the first digital inputs received, for example, beginning at the moment a pushbutton has been pressed (see also *3.5.1 Trigger*).

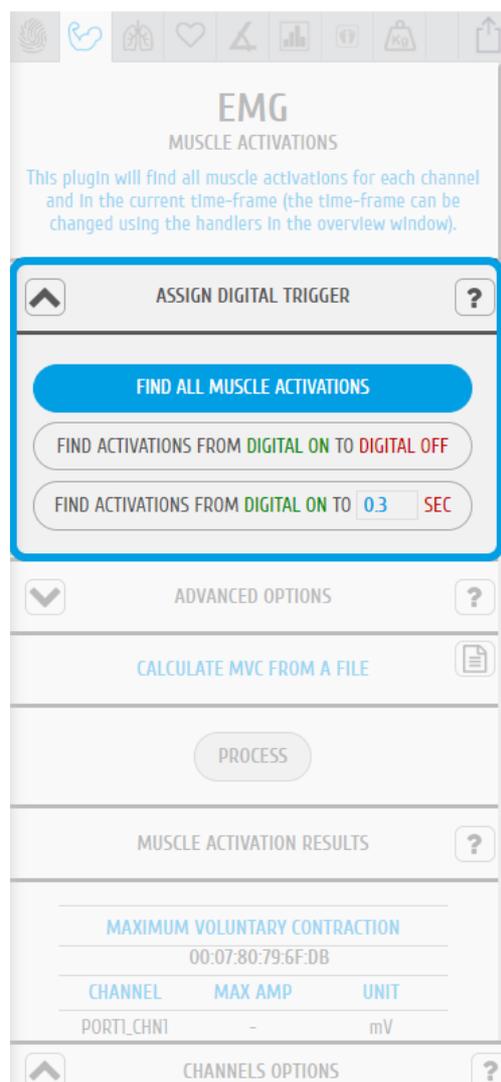


Figure 180: EMG plugin | digital trigger options

ADVANCED OPTIONS - Filters

The advanced options allows you to change the parameters which are being used in the activation detection algorithm to adjust the processing stage of your acquisition. In this option, you can select if you want to use a lowpass, highpass or a bandpass filter to be applied on your signal and set the respective cutoff frequencies for the selected filter.

To use a lowpass filter, click on the *LOWPASS* button and enter the preferred cutoff frequency for this filter in the *CUTOFF - LOW* field and enter the order of the filter in the *ORDER* field.

To use a highpass filter, click on the *HIGHPASS* button and enter the preferred cutoff frequency for this filter in the *CUTOFF - HIGH* field and enter the order of the filter in the *ORDER* field.

To use a bandpass filter, click on the *BANDPASS* button and enter the preferred cutoff frequency for the lowpass component of this filter in the *CUTOFF - LOW* field and enter the preferred cutoff frequency for the highpass component of this filter in the *CUTOFF - HIGH*. Enter the order of the filter in the *ORDER* field.

ADVANCED OPTIONS – Signal Smooth Window

In order to ensure an effective detection of muscular activations, the EMG signals needs to be smoothed, using a moving average window, taking into consideration that the signal envelope is morphologically simpler, and the inexistence of abrupt transitions is ideal to apply a single-threshold detection algorithm.

The size of the moving window can be configured on the “ADVANCED OPTIONS” panel.

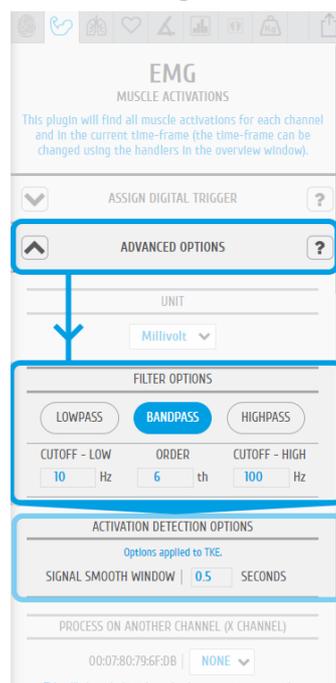


Figure 181: EMG plugin | signal filtering options

ADVANCED OPTIONS – Process X Channel

This option allows you to apply the statistical algorithms of the add-on to a selected channel X within the same time intervals of another channel without any activation dependency of channel X. The add-on will search for activations on a selected EMG channel (click on the channel to select) and analyze the selected channel X at the same time intervals where activations have been found in the selected EMG channel even if there are no activations or activation triggering activities within these time intervals in channel X (e.g. muscle is relaxed during this interval).

With this option it is possible to compare two EMG signals within the same time intervals without any activation dependencies.

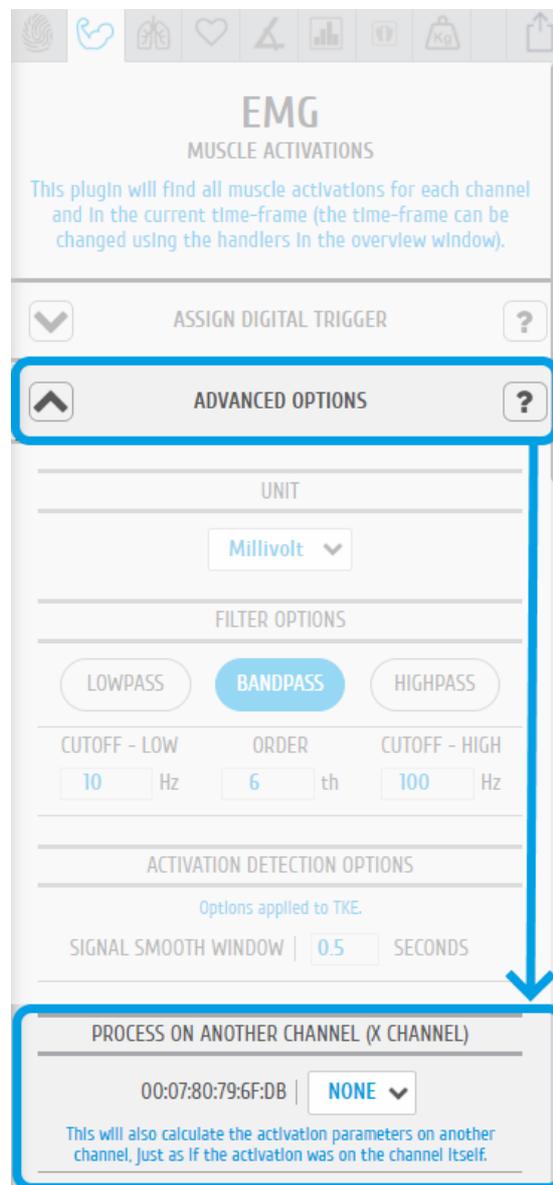


Figure 182: EMG plugin | definition of an additional signal to process

MVC – MAXIMUM VOLUNTARY CONTRACTION

The maximum voluntary contraction (MVC) will be calculated for each channel based on the EMG maxima of your selected EMG signals.

The MVC acquisition can also be done in a separate signal acquisition session/file. This can be useful to use when doing several signal acquisitions to ensure that the analysis of all files and signals are done with the same reference values – the MVC values.

For this, start an acquisition and acquire your data which are being used as MVC values for later analysis and save your acquisition. To use the MVC data from this acquisition in another acquisition, click on the document icon in the *CALCULATE FROM A FILE INSTEAD* field and select the *OpenSignals (r)evolution* file which contains the MVC signal data. *OpenSignals (r)evolution* will then calculate the MVC value from the selected file.



Select *OpenSignals (r)evolution* file to calculate MVC values from file

Each channel's MVC value are listed in the table below *MAXIMUM VOLUNTARY CONTRACTION*.

MUSCLE ACTIVATION RESULTS

Click on the *PROCESS* button to process all signals according to the set configuration and all found activations will be visualized in the processed channels (white frames).

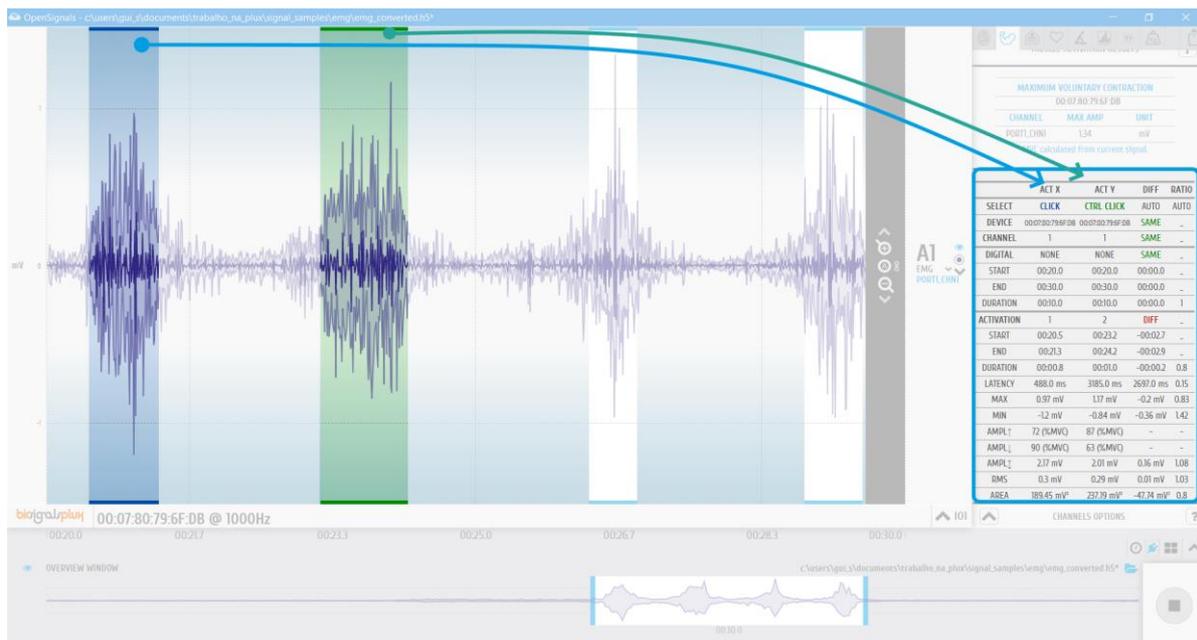


Figure 183: EMG plugin | muscular activation results

To display all statistical information about one or two specific activations, click on the first activation (left mouse button). The white frame turns into a blue frame and all statistical information about this activation will be listed in the *ACT X* column of the analysis results. When done, select the second activation of your choice by holding the *CTRL* key of your keyboard and click on the activation (left mouse button). The white frame turns into a green frame and all statistical information about this activation will be listed in the *ACT Y* column of the analysis results.

The results table displays general information about the source of this signal (device & channel) as well as general information about the duration and times of your acquisition. All activation specific data is displayed below the given general information (e.g., activation number, duration).

4.12 Electrodermal Activity (EDA) Events



Computes overall statistics and basic spectral analysis from EDA sensor data

DESCRIPTION

Electrodermal responses are characterized as phasic changes in the skin conductance, associated with the sympathetic nervous system activation. This plugin has been designed to compute overall statistics, basic spectral analysis and extract typical event-related phasic features from Electrodermal Activity (EDA) sensor data.

This is an offline add-on for post processing purposes. It is only available in the signal visualization mode (see [3.2 Signal Visualization Mode](#)).

This add-on is included in the [biosignalsplux Professional](#). It can also be bought in addition to all *biosignalsplux* and *BITalino* devices.

4.12.1 Hardware Configuration

The EDA analysis add-on processes sensor data acquired with one of the following sensors:

biosignalsplux EDA Sensor:

<https://www.pluxbiosignals.com/products/electrodermal-activity-eda-sensor-1>

BITalino EDA Sensor:

<https://www.pluxbiosignals.com/products/electrodermal-activity-eda-sensor>

The channel(s) you have one of these sensors connected to have to be configured properly before starting the acquisition. To do so, follow the device configuration instructions as described in [2.2.4 Device Configuration](#). During the configuration process make sure to select the channel your EDA sensor is connected to and to set the channel configuration to *EDA* (see [2.2.4.6 Channel Configuration](#)).

4.12.2 Add-on Configuration

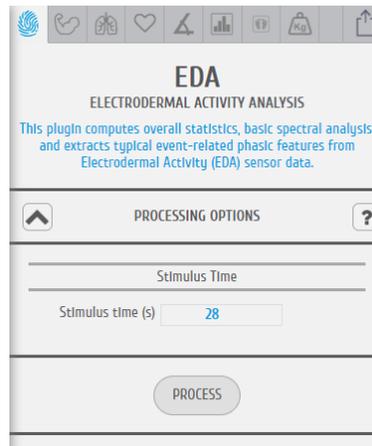


Figure 184: EDA analysis add-on configuration.

DEVICES & CHANNELS

In order to specify the processing time-window/interval, we recommend the reading of section 4.4

Make sure you have followed the instructions on [4.12.1 Hardware Configuration](#) for the add-on to be able to recognize your EDA channels.

STIMULUS TIME

Input the time at which the test person has experienced any external stimulus during your experiment. The input time in this field will mark the stimulus in the acquired signal.

RESPONSE TIME

The response time interval will set the time span of the EDA signal data to be processed after the external stimulus. This add-on will therefore process all the data that falls within the response time interval after the set stimulus time.

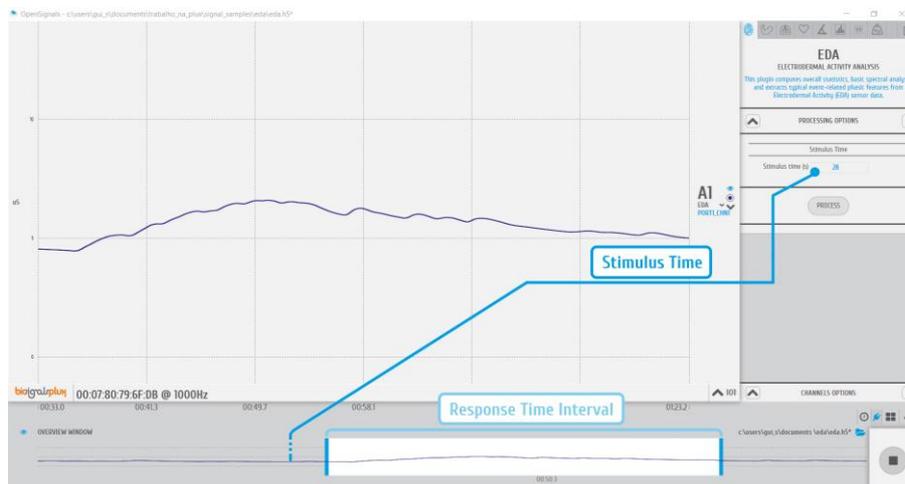


Figure 185: EDA add-on | stimulus time and response interval.

EDA PROCESSING RESULTS

The processing results returned by the EDA add-on are divided in “Phasic Parameters”, “Signal Parameters”, “Histogram” and “PSD”.

For the “Phasic Parameters” section, are presented metrics such as *Electrodermal Response* (time between the stimulus and the start of the electrodermal reaction) and *Electrodermal Amplitude* (intensity of the electrodermal response).

Regarding the “Signal Parameters” are included statistical parameters, namely the maximum, minimum and mean values of the registered skin conductance.

In terms of “Histogram” and “PSD”, these two sections present graphical objects intended to describe the statistical distribution of the sampled values and the power spectral density function, respectively.

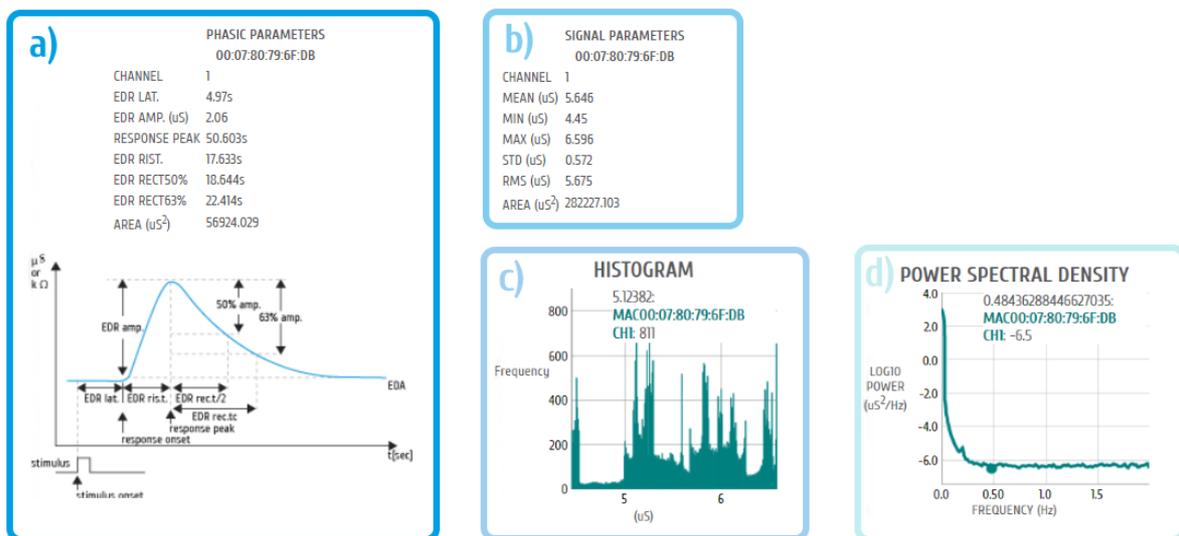


Figure 186: EDA add-on | processing results: a) Phasic Parameter; b) Signal Parameters; c) Histogram; d) PSD.

4.13 Heart Rate Variability (HRV) Analysis



HRV analysis add-on

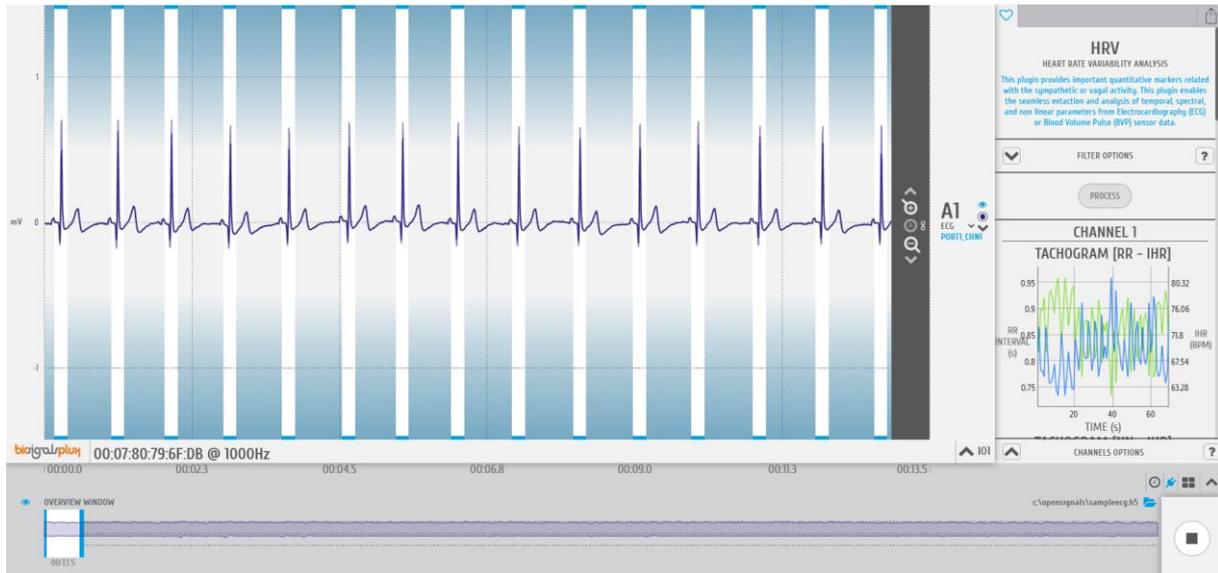


Figure 187: HRV analysis add-on | highlighting the detected R peaks .

DESCRIPTION

Heart Rate Variability (HRV) provides important quantitative markers related with the sympathetic or vagal activity. This plugin enables the seamless extraction and analysis of temporal, spectral, and non-linear parameters from Electrocardiography (ECG) or Blood Volume Pulse (BVP) sensor data. All the algorithms were implemented according to the "Standards of Measurement, Physiological Interpretation, and Clinical Use" devised by the joint European Society of Cardiology and North American Society of Pacing Electrophysiology task force.

This is both an offline and a real-time add-on, available in the signal visualization (see [3.2 Signal Visualization Mode](#)) and in the signal acquisition (see [3.1 Signal Acquisition Mode](#)) modes.

This add-on is included in the [biosignalsplux Professional Kit](#) and the [CardioBAN](#). It can also be bought in addition to all [biosignalsplux](#) and [BITalino](#) devices.

4.13.1 Hardware Configuration

The HRV analysis add-on processes sensor data acquired with one of the following sensors:

biosignalsplux ECG Sensor:

<https://www.pluxbiosignals.com/products/electrocardiography-ecg-sensor-1>

biosignalsplux BVP Sensor:

<https://www.pluxbiosignals.com/products/blood-volume-pulse-bvp-sensor>

biosignalsplux SpO₂ Versatile Sensor:

<https://www.pluxbiosignals.com/products/blood-oxygen-saturation-spo2-versatile-sensor>

biosignalsplux SpO₂ Finger Sensor:

<https://www.pluxbiosignals.com/products/blood-oxygen-saturation-spo2-sensor>

biosignalsplux fNIRS Sensor:

<https://www.pluxbiosignals.com/products/functional-near-infrared-spectroscopy-fnirs-sensor>

BITalino ECG Sensor:

<https://www.pluxbiosignals.com/products/electrocardiography-ecg-sensor>

fNIRS Explorer

<https://www.pluxbiosignals.com/products/fnirs-explorer-kit>

We recommend configuring, before starting the acquisition, properly the channel(s) where you have one of these sensors connected to. To do so, follow the device configuration instruction as described in *2.2.4 Device Configuration*. During the configuration process make sure to select the channel your ECG sensor is connected to and to set the channel configuration to *ECG* or, if using a BVP, SpO₂ or fNIRS sensor, set the channel configuration to *BVP*, *SpO₂.ARM* or *SpO₂.FING* and *fNIRS*, respectively (see *2.2.4.6 Channel Configuration*).

4.13.2 Add-on Configuration

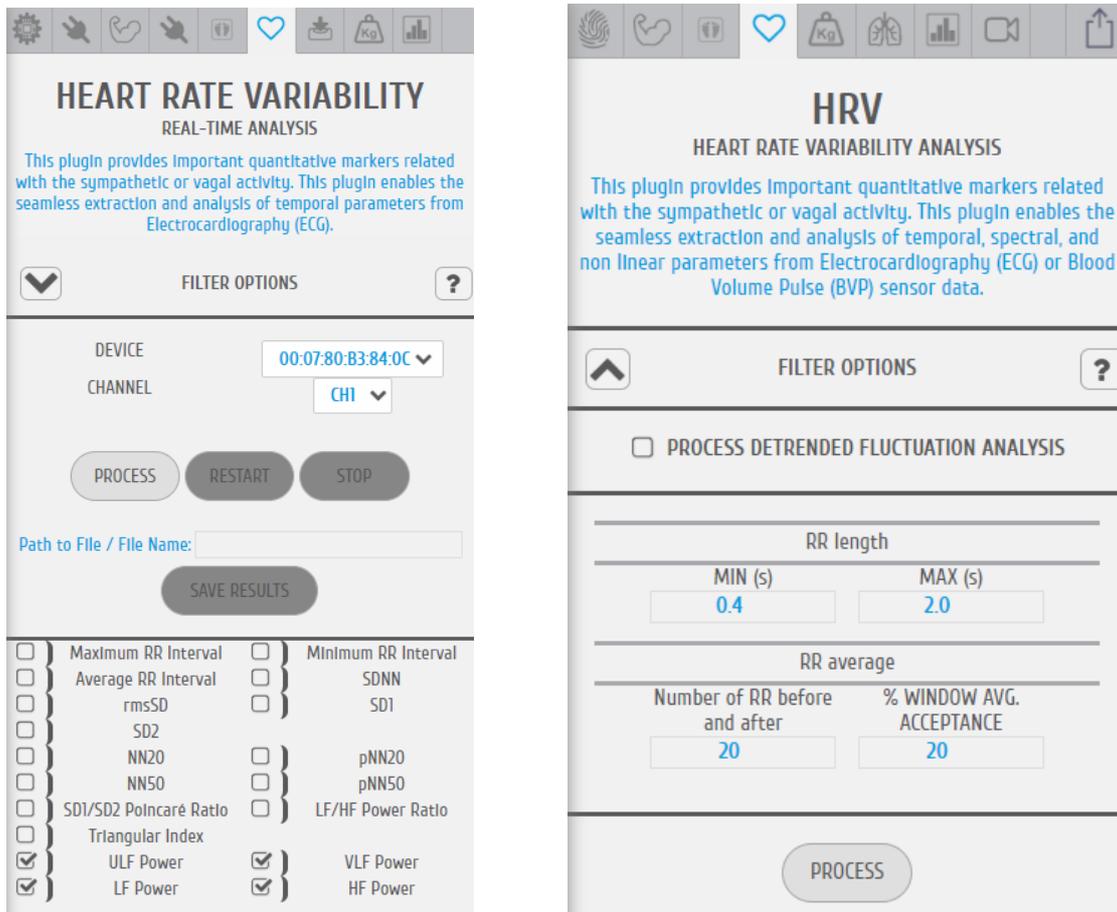


Figure 188: HRV analysis add-on (acquisition mode @left and visualization mode @right).

ACQUISITION MODE DEVICES & CHANNELS

The current add-on has two dedicated fields (dropdown menus) where the user can select the **1)** device under analysis and **2)** the channel containing the data to be processed.

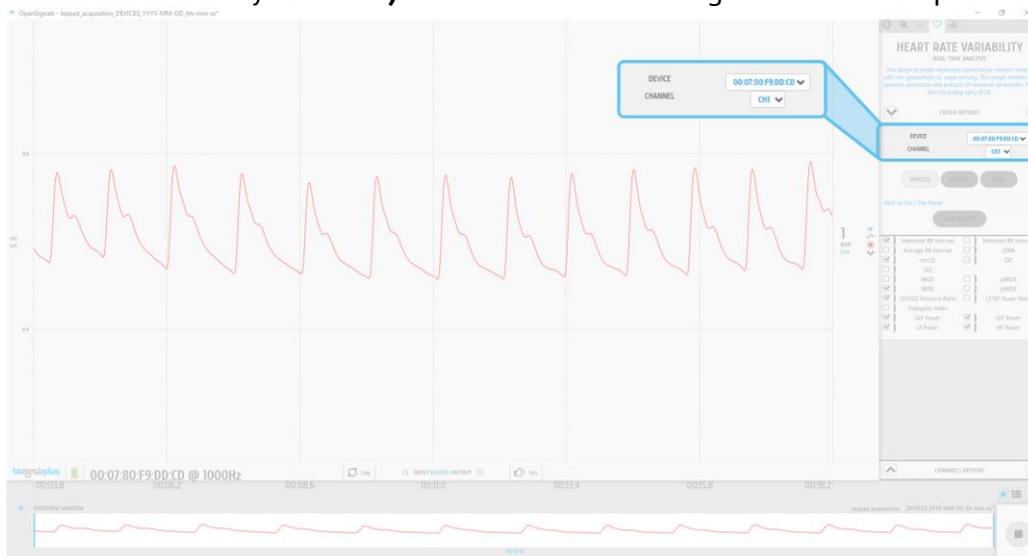


Figure 189: Device and Channels options in the HRV real-time add-on

However, it should be noted that only channels with the **ECG**, **SpO2.ARM**, **SpO2.FING**, **fNIRS** and **BVP** labels will appear in the list, considering that are the only ones where the processing algorithms can be properly applied.

FILTER OPTIONS | FILTER RR [RR LENGTH]

This RR filter improves the R-peak detection by analyzing the R-peaks within a set time interval, detecting the minimum RR interval value and the maximum RR interval value. This filter can be useful when analyzing ECG signals which have been acquired under heavy movement, where muscle activity can cause variations in R-peak heights, generating displaced R peak detections, which conditionate the RR interval values.

In order to apply this type of filtering stage, enter the time of both the start and ending point of the preferred interval into the *RR. MIN (S)* and *RR. MAX (S)* fields.

With this filtering methodology we are excluding RR intervals with duration outside $[RR. MIN (S); RR. MAX (S)]$, i.e., outliers that may not be related with the conditions under study.

FILTER OPTIONS | FILTER RR [RR AVERAGE]

This RR filter improves the R-peak detection and signal processing by analyzing the noise within a set number of RR-intervals before and after each detected R-peak.

Two inputs must be given to this filter: **Number of RR before and after** and **% Window Average Acceptance (WAA)**.

The basic logic behind this filtering stage consists in the usage of an average sliding window, that is, each sample of the tachogram (time-series that shows how RR interval duration changes along time) is scanned and the mean value of the tachogram samples contained in the user-defined window is determined. If the sample of present iteration has a duration exceeding the defined interval $[\overline{RR}_{interval} - WAA; \overline{RR}_{interval} + WAA]$ it is excluded from tachogram.

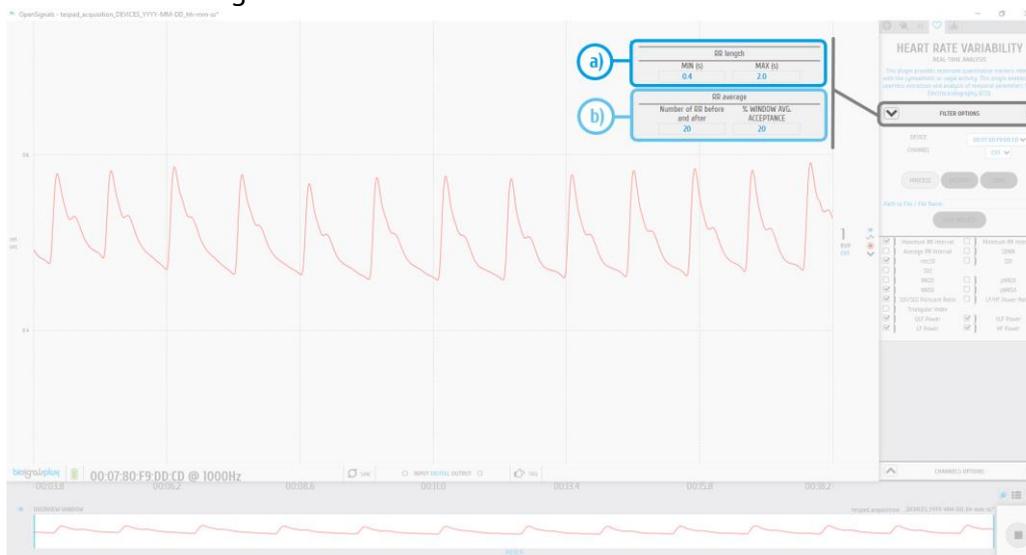


Figure 190: Filtering options: a) FILTER RR [RR LENGTH] b) FILTER RR [RR AVERAGE]

HRV AVAILABLE ACTIONS

The user has the possibility of taking four types of actions in the add-on graphical user interface (GUI):

a) PROCESS

Used to trigger the start of the HRV real-time processing algorithms.

b) RESTART

Button only activated after the **PROCESS** button has been clicked. Through this button the user can restart the processing algorithms and discard all previous results.

c) STOP

When clicked the execution of the processing algorithms is immediately stopped and the **PROCESS** button becomes again enabled while the **RESTART** and **STOP** buttons return to the disabled state.

d) PATH TO FILE / FILENAME

In this field the user can write the path where the processing results should be stored (after the **SAVE RESULTS** button has been pressed) and the respective filename.

e) SAVE RESULTS

Through this action the user can store the processing results in the folder specified in the **Path to File / Filename** field.

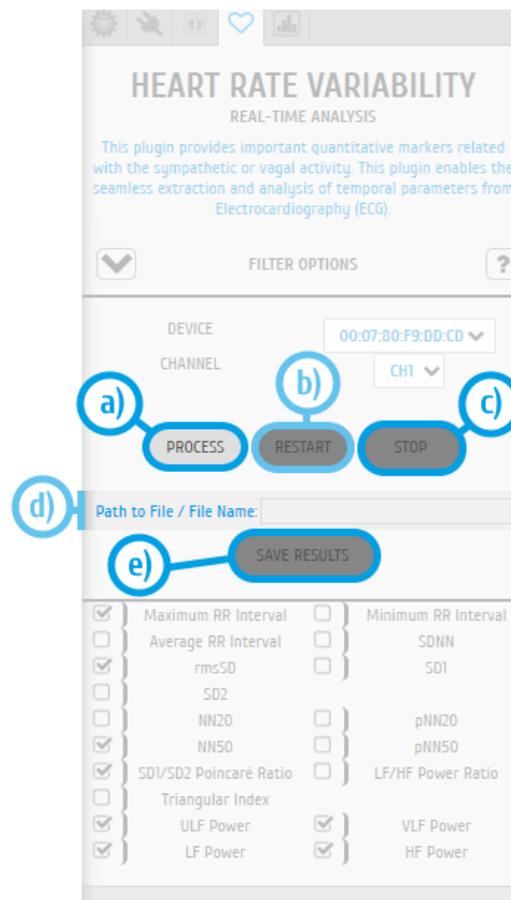


Figure 191: Available actions in the HRV Real-Time add-on.

HRV PROCESSING RESULTS

The add-on processing results are presented in a real-time chart format. The generated charts will contain the evolution of the parameters selected by the user during the acquisition:

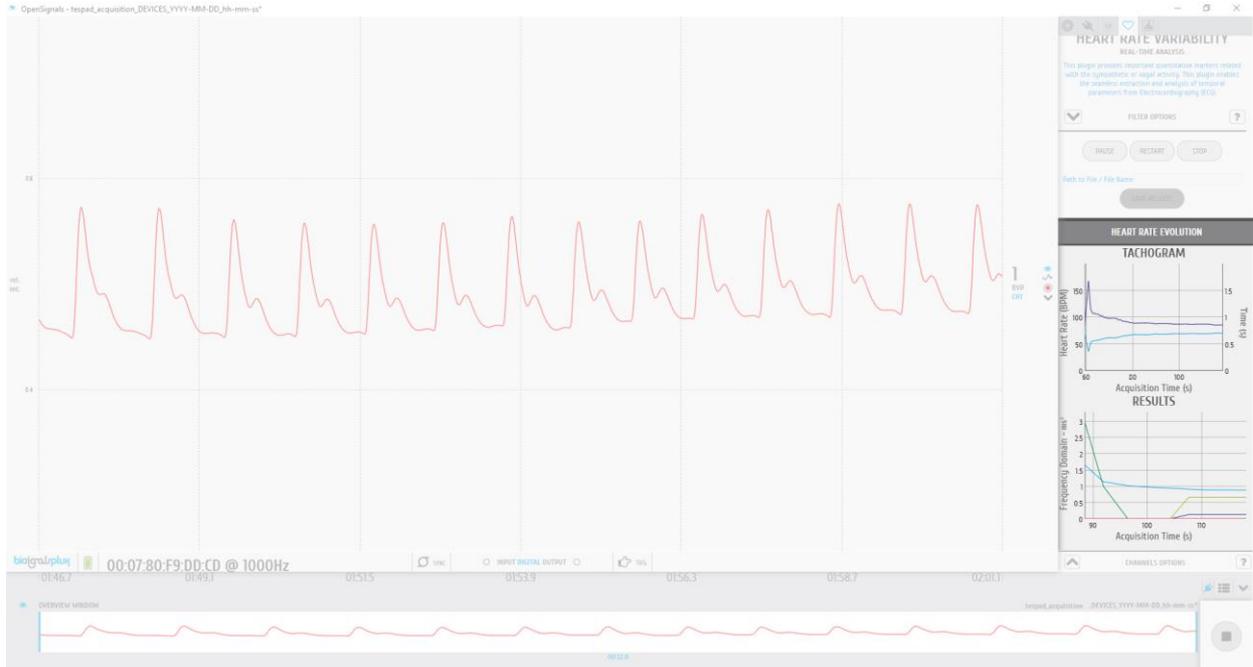


Figure 192: HRV Real-Time add-on with the processing results in a chart format.

NOTE

A brief description about the meaning of each extracted parameter can be found in the following Position Paper which contains some primordial steps followed towards the creation of the **HRV Real-Time Add-On**:

- a) [Real-time approach to HRV analysis](#)

VISUALISATION MODE

DEVICES & CHANNELS

For selecting the devices and channels where the processing algorithms will be executed, we recommend the reading of section 4.4

PROCESSING INTERVAL

In order to specify the processing interval, the reading of section 4.4 is recommended, note, that short intervals may not provide enough data to conduct a deep analysis. In this case, *OpenSignals (r)evolution* will issue a warning (“Not enough NN Peaks for deeper analysis.”; not enough beat-to-beat intervals found).

FILTER OPTIONS

The filtering options available in the signal visualization mode add-on are identical to the ones described in the previous pages for the signals acquisition mode add-on.

PROCESS DETRENDED FLUCTUATION ANALYSIS

If this option is active then, two extra charts will be drawn containing the interpretable results for a Detrended Fluctuation Analysis (DFA), which is particularly useful in detecting long-range correlations in nonstationary data and also in distinguishing data extracted from healthy subjects and from individuals with cardiac dysfunctionalities.

HRV PROCESSING RESULTS

The processing results generated by the HRV add-on includes both graphical and numerical elements, namely:

- Tachogram (structural element on the entire HRV analysis);
- Histogram (with the distribution of the RR interval durations);
- Power Spectral Density function (ensuring a frequency analysis of tachogram);
- Poincaré Plot and DFA charts (for extraction of non-linear parameters);
- Sequence of time parameters, such as the minimum, maximum and average duration of RR intervals.

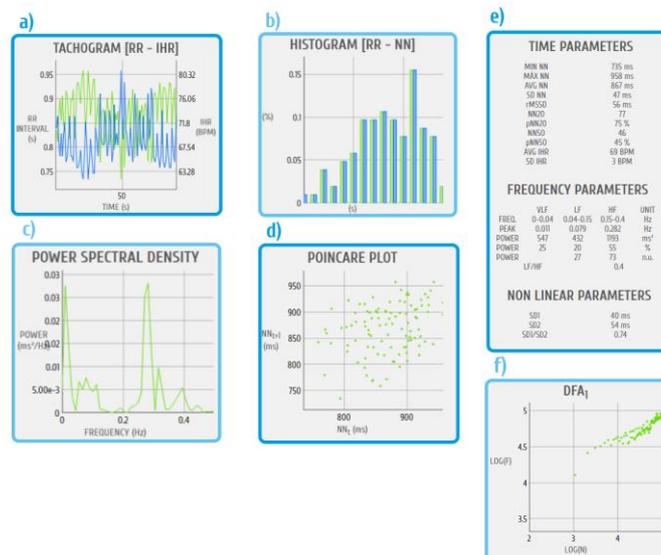


Figure 193: HRV Add-on results | a) tachogram, b) histogram, c) Power Spectral Density, d) Poincaré Plot e) Numerical Parameters and f) Detrended Fluctuation Analysis.

4.14 Respiration Analysis (PZT & RIP)



Respiration analysis add-on

DESCRIPTION

Respiratory data provide useful information about the breathing dynamics. Designed to work with Piezoelectric Respiration (PZT) and Inductive Respiration (RIP) sensor data, this plugin is a convenient way to determine respiratory rate and other useful temporal and statistical parameters associated with the respiratory cycles.

This is an offline add-on for post processing purposes. It is only available in the signal visualization mode (see *3.2 Signal Visualization Mode*).

This add-on is included in the [biosignalsplux Professional kit](#) and the [RespiBAN Professional](#) wearable. It can also be bought in addition to all *biosignalsplux* and *BITalino* devices.

4.14.1 Hardware Configuration

The Respiration analysis add-on processes sensor data acquired with one of the following sensors:

biosignalsplux RIP Sensor:

<https://www.pluxbiosignals.com/products/inductive-respiration-rip-sensor>

biosignalsplux PZT Sensor:

<https://www.pluxbiosignals.com/products/respiration-pzt>

We recommend to properly configure, before starting the acquisition, the channel(s) you have one of these sensors connected to. To do so, follow the device configuration instruction as described in *2.2.4 Device Configuration*. During the configuration process make sure to select the channel your PZT or RIP sensor is connected to and to set the channel configuration to *RESP* (see *2.2.4.6 Channel Configuration*).

4.15 Add-on Configuration

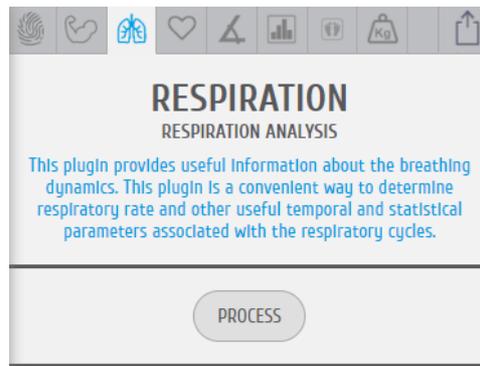


Figure 194: Respiration analysis add-on.

DEVICES & CHANNELS

In order to specify the devices and channels to be processed, we recommend the reading of section 4.4

PROCESSING INTERVAL

Following the previous recommendation (for DEVICES & CHANNELS), a simple illustrative example of how to access the processing interval options is available at section 4.4

RESPIRATION ANALYSIS PROCESSING RESULTS

Respiration analysis included in this add-on encompasses a diversified set of parameters, both in quantity and diversity.

The extracted parameters are divided into six sections:

- a) Amplitude Parameters;
- b) Signal Parameters;
- c) Histogram;
- d) Frequency Parameters;
- e) Power Spectral Density (PSD);
- f) Time Parameters.

Amplitude Parameters refers to a set of statistical parameters (maximum, mean, standard deviation and root-mean-square) extracted directly from the processing window under analysis, but taking only into consideration the respiratory peaks (i.e. the maximum value of each respiratory cycle).

Signal Parameters are similar to the ones included on *Amplitude Parameters* section, but in this case all samples inside the processing window are taken into consideration (and not only the respiratory peaks like in the previous section).

Regarding *Histogram* and *Power Spectral Density*, these are two graphical outputs which describes, respectively, the distribution of the respiratory signal sample values and the relative weight of each frequency component during the signal reconstruction by sinusoidal wave superposition.

Time Parameters include a set of parameters intended to demonstrate the maximum, minimum and average duration of respiratory cycles, while *Frequency Parameters* make use of *Time Parameters* to estimate the number of breaths per minute.

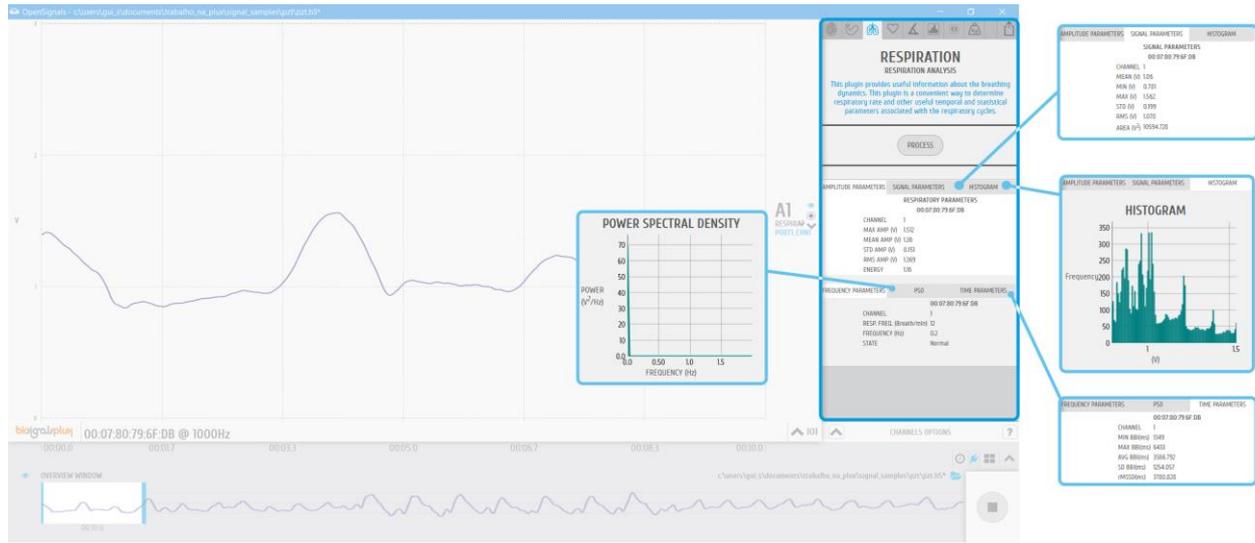


Figure 195: Respiration Add-on results

4.16 Load Cell Force Analysis



Load Cell Force Analysis add-on

DESCRIPTION

The *Load Cell Force Analysis* add-on extracts traction (kg) and compression (kg) maxima from forces applied on *biosignalsplux* Load Cells in real-time. It supports multiple Load Cells where baseline values can be individually set for each Load Cell (e.g. to remove offsets).

This add-on is an online add-on. It is available in the signal acquisition mode (see [3.1 Signal Acquisition Mode](#)).

This add-on is included in the [biosignalsplux Professional kit](#).

4.16.1 Hardware Configuration

NOTE

This add-on requires an active internet access to download Load Cell parameters using Load Cell IDs which you can find on your Load Cell. These parameters are used to compute Load Cell values and have to be downloaded by *OpenSignals (r)evolution* when configuring a Load Cell for the very first time on a computer. Afterwards, these parameters will be stored on your computer and no further internet access is needed when using this add-on.

The Load Cell add-on processes sensor data acquired with *biosignalsplux* Load Cell:

biosignalsplux Load Cell:

<https://www.pluxbiosignals.com/products/load-cell>

The channels to which a Load Cell is connected have to be properly configured for this add-on. To do so, follow the device configuration instruction as described in [2.2.4 Device Configuration](#). During the configuration process make sure to select the channel your Load Cell is connected to and to set the channel configuration to LOADCELL (see [2.2.4.6 Channel Configuration](#)). When done, the device panel should look similar to the device panel in Figure 196, with a LOADCELL being used on channel 1 and the Load Cell parameters being set to X.

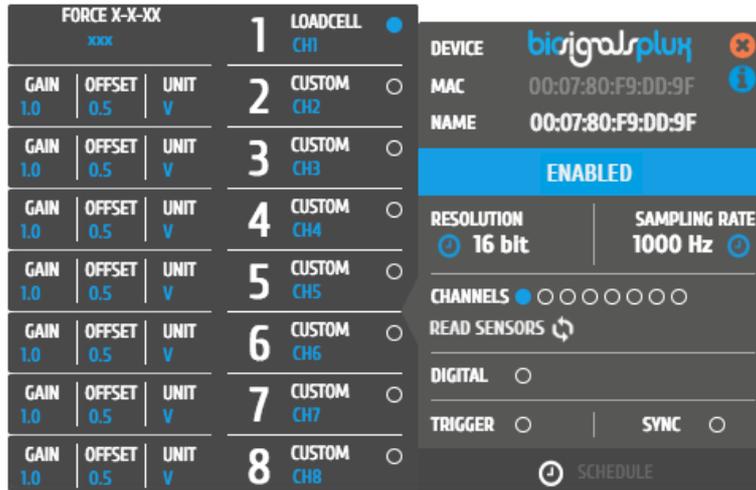


Figure 196: Unconfigured Load Cell channel.

Click on the blue **xxx** below the **FORCE X-X-XXX** label and enter the Load Cell ID of your Load Cell which you can find on your Load Cell. The ID is a 5 digit number in X-X-XXX format. Please contact our support team (contact info and further support steps available at <https://support.pluxbiosignals.com/>) if you cannot find any ID or if no ID has been provided to you with the Load Cell. When done, the device panel should look similar to the device panel seen in Figure 197 (ID being used here: 1-5-360).

NOTE

It is important to use the correct Load Cell ID of the Load Cell being used. *OpenSignals (r)evolution* downloads Load Cell specific parameters which are taken during the manufacturing process and vary with every Load Cell, i.e. every Load Cell has its own specific set of parameters.

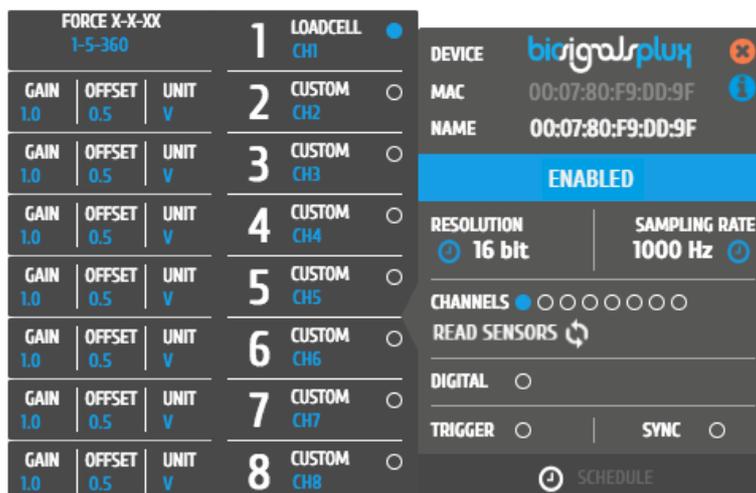


Figure 197: Configured Load Cell with a Load Cell ID (here: 1-5-360).

4.16.2 Add-on Configuration

Start the signal acquisition using a configured Load Cell and open the Load Cell add-on (see 4.2 Listing & Using Add-ons).

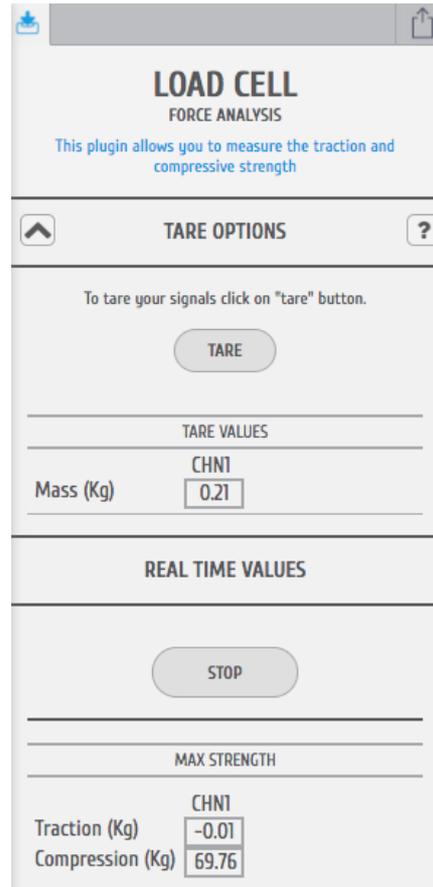


Figure 198: Load Cell Force Analysis add-on.

TARE OPTIONS

The *Tare Options* allow you to set baseline values for the Load Cells being used, i.e. the current force being applied will be set as a calibrated 0kg when clicked on the *TARE* button.

This can be useful when using additional accessories on the Load Cell, which would create an offset in the signal by default and where this offset should not be measured to avoid distortions of the measured signals. In this case, place the accessory on the Load Cell and use the *TARE* option to set the current weight to the 0kg reference.

The baseline values being used will then be listed in the *TARE VALUES* section of the add-on, where each Load Cells' tared value will be listed.

REAL TIME VALUES

The *Real Time Values* section will show the maximum traction (kg) and compression (kg) of the Load Cell in the *MAX STRENGTH*. To acquire these values, click on the *START* values and apply forces as required on the Load Cell. The add-on will save the maximum values and update the *MAX STRENGTH* whenever new maxima have been detected. If a reset of the detected values is required, click on the *STOP* button and restart the acquisition of the maxima.

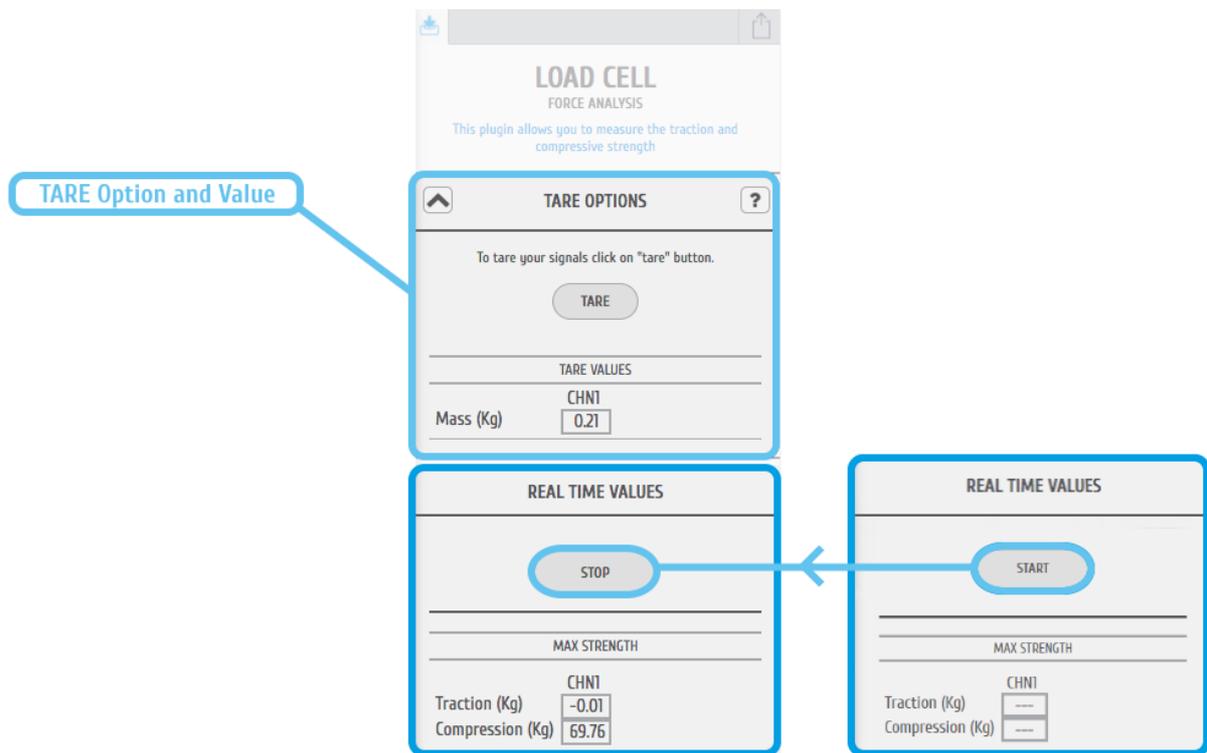


Figure 199: Load Cell Force Analysis TARE Option and Real-Time Values.

EXPORTING REPORTS

Report exportation of the acquired data can be done by following the instructions in section 4.3 *Add-on Reports*. Note, that the current version does only support report exportation in CSV format.

4.17 Video Synchronization

Video synchronization add-on

DESCRIPTION

Multimodal data acquisition in human studies usually involves recording data from sources other than the biosignal acquisition hardware devices (e.g. video camera). Given that the biosignals hardware and the camera are independent recording sources, a common problem when replaying the recording session is the synchronization of both. This plugin was created to provide an easy way to replay biosignal data synchronously with video using for example our LED accessory to provide a common event to both devices.

This is an offline add-on for post processing purposes. It is only available in the signal visualization mode (see *3.2 Signal Visualization Mode*).

This add-on is included in the [biosignalsplux Researcher](#), [biosignalsplux Professional](#), [biosignalsplux ergoExplorer](#) and [biosignalsplux ergoResearcher](#). It can also be bought in addition to all *biosignalsplux* and *BITalino* devices.

4.17.1 Hardware Configuration

The add-on does not specifically require any actuators or accessories, however, the use of one of the following is recommended to facilitate the synchronization of your signal and video recordings using light sources and light sensors.

LIGHT SENSORS

When using light sensors to record external light signals coming from other devices, it is recommended to configure the channel your sensor is connected to a suitable channel type (see *2.2.4.6 Channel Configuration* and *5 Advanced Device Configuration* of your device).

biosignalsplux light sensor:

<https://www.pluxbiosignals.com/products/light-lux-sensor-1>

BITalino light sensor:

<https://www.pluxbiosignals.com/products/light-lux-sensor>

LEDs

When using LEDs to emit light signals to synchronize the device of your setup, connect the LED to your device and follow to instruction in section 3.1.2.4 *Digital Inputs & Outputs* on how to control your LED during the acquisition.

biosignalsplux LED:

<https://www.pluxbiosignals.com/products/light-emitting-diode-led-1>

BITalino LED:

<https://www.pluxbiosignals.com/products/light-emitting-diode-led>

4.17.2 Add-on Configuration

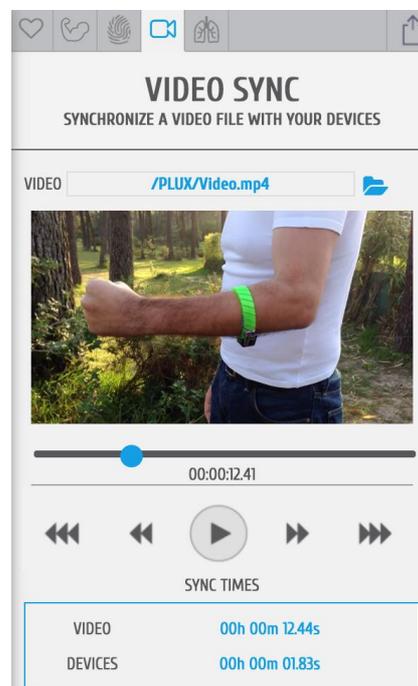


Figure 200: Video Synchronization add-on.

DEVICES & CHANNELS

The video synchronization add-on does not have to be configured to a specific channel or device. The video mark that synchronizes the video will match all available signals, i.e., it affects all acquisition channels equally.

VIDEO PATH

Click on the folder next to the video field and select the acquisition video.

SYNC TIMES

Sync times displays the information of the video mark which synchronizes the video at the indicated time with a selected point in the acquired signals (see detailed information below).

Video: Locked synchronization point in the video timeline

Devices: Locked synchronization point in the acquired signals



Locked synchronization point

The lock marks the synchronization point between the video and the acquired signals. Stop the video at the time point where the synchronization input (e.g. visual input via LED) was set in the video. Move the lock to the time point in your acquisition where the acquired signal overlap with your recorded video and click on the “lock” to fix this point as the synchronization point between your video and your acquisitions.



Current position

The “aim” icon uses the locked synchronization point as a reference point and displays the current position of your acquired signals compared to the acquisition video. It will only be displayed in the signal window if the displayed signals lie within the synchronized video interval.

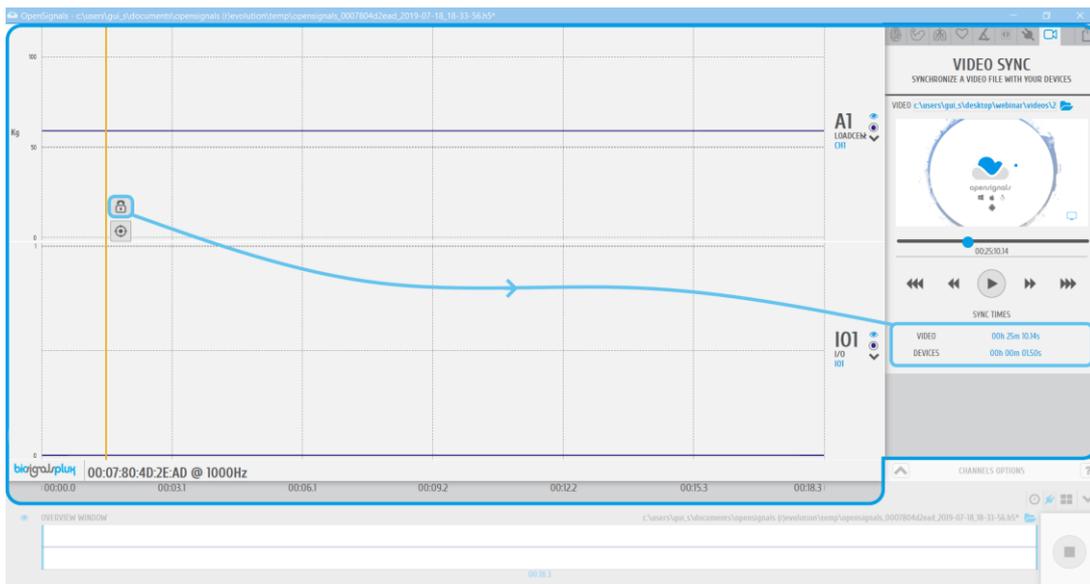


Figure 201: Synchronization button and respective sync time instants.

4.18 Force Platform



Force Platform add-on

DESCRIPTION

Force Platform add-on is both an online and post-processing add-on. It is available in the signal acquisition mode (see *3.1 Signal Acquisition Mode*) and in signal visualization mode (see *3.2 Signal Visualization Mode*).

The Force Platform is composed by 4 load cells responsible for the detection of compression forces acting in the Force Platform surface, due to the weight of the volunteer standing or jumping in the area embraced by the platform.

The four load-cells are positioned in a standardized and precise way, equidistant from the center of the platform, ensuring that a punctual/conceptual body located at the center of the platform will originate identical measurements in all load-cell channels, i.e, the weight is equally distributed in the measurement surface.

With the *Force Platform* add-on it is possible to determine the exact mass on the platform, the respective exerted force and the position of the center of pressure (real-time functionalities), while in the post-processing version parameters such as maximum force or flight time/height can be determined.

This add-on is included in the [biosignalsplux Professional Kit](#) and in the [biosignalsplux Force kit](#).

4.18.1 Hardware Configuration

The *Force Platform* add-on processes sensor data acquired with *biosignalsplux* Force Platform:

biosignalsplux Force Platform:

<https://www.pluxbiosignals.com/products/force-platform>

As previously mentioned, the *Force Platform* is composed by 4 load-cells, so, in order to use this resource in an appropriate way it is necessary to activated 4 channels containing the “PLATFORM” label. To do so, follow the device configuration instructions as described in *2.2.4 Device Configuration*. When done, the device panel should look similar to the device panel in Figure 202, with the four active channels with the label “PLATFORM” and the Vfs parameter equal to 0.

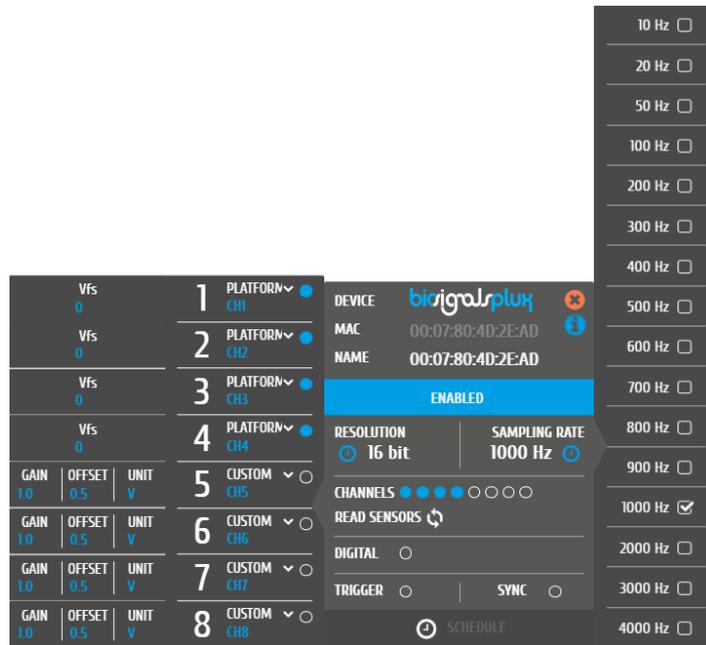


Figure 202: Unconfigured Force Platform channels.

Click on the blue **0** below the **Vfs** label, here it is possible to define the output voltage in mV/V @ 200kgf, which is specific from each platform/load-cell, being essential to ensure the presentation of calibrated values. After finishing this procedure, the device panel should look similar to the device panel seen in Figure 203.



Figure 203: Configured Force Platform channels.

4.18.2 Add-on Configuration

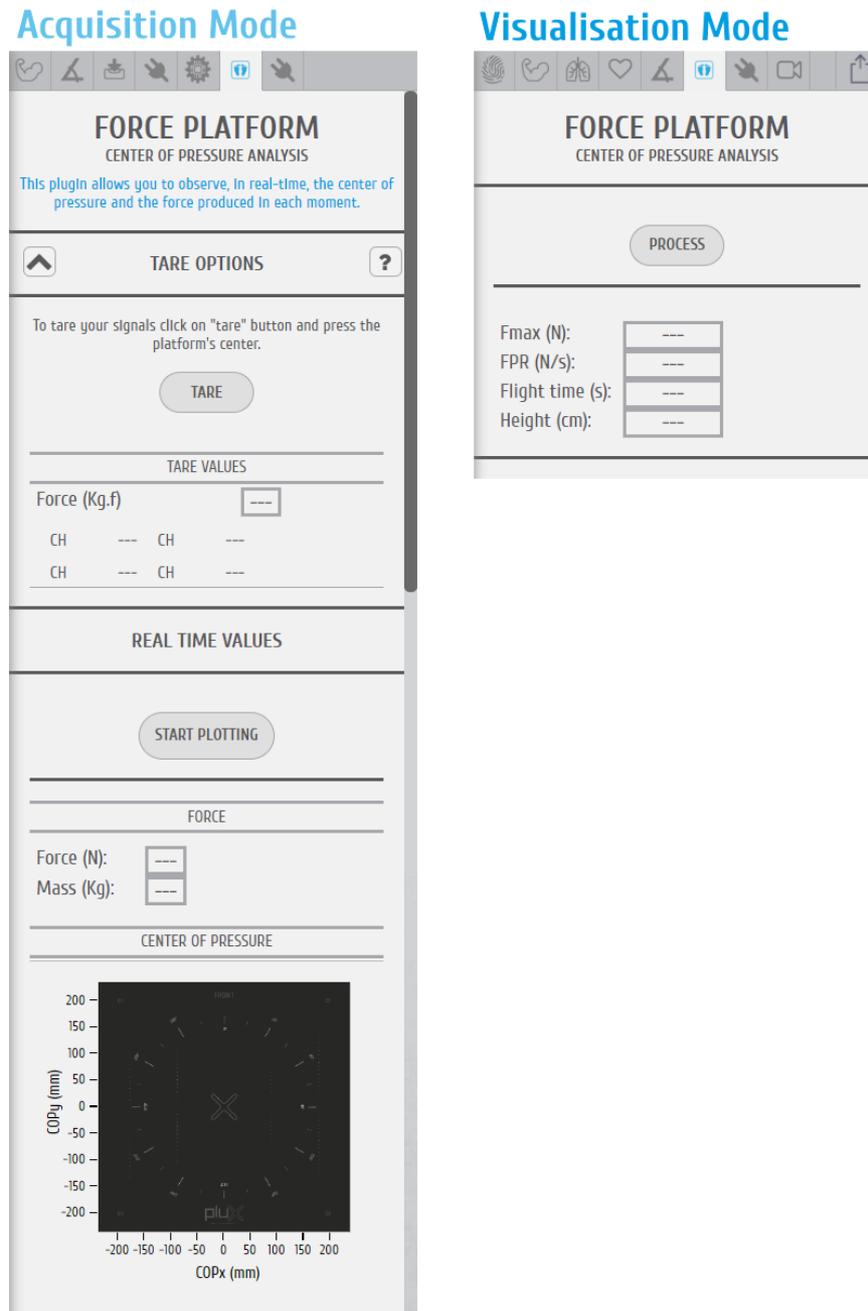


Figure 204: Load Cell Force Analysis add-on.

ACQUISITION MODE

TARE OPTIONS

The *Tare Options* allow you to set baseline values for the Load Cells that compose the Force Platform, i.e. the current force being applied will be set as a calibrated 0kg when clicked on the *TARE* button.

The baseline values being used will then be listed in the *TARE VALUES* section of the add-on.

REAL TIME VALUES

The *Real Time Values* section will show the total force (N) being exerted into the platform, together with the respective mass estimate (kg). These two values are instantaneous parameters.

Additionally, the center of pressure position is represented on the graphical object included in the add-on interface.

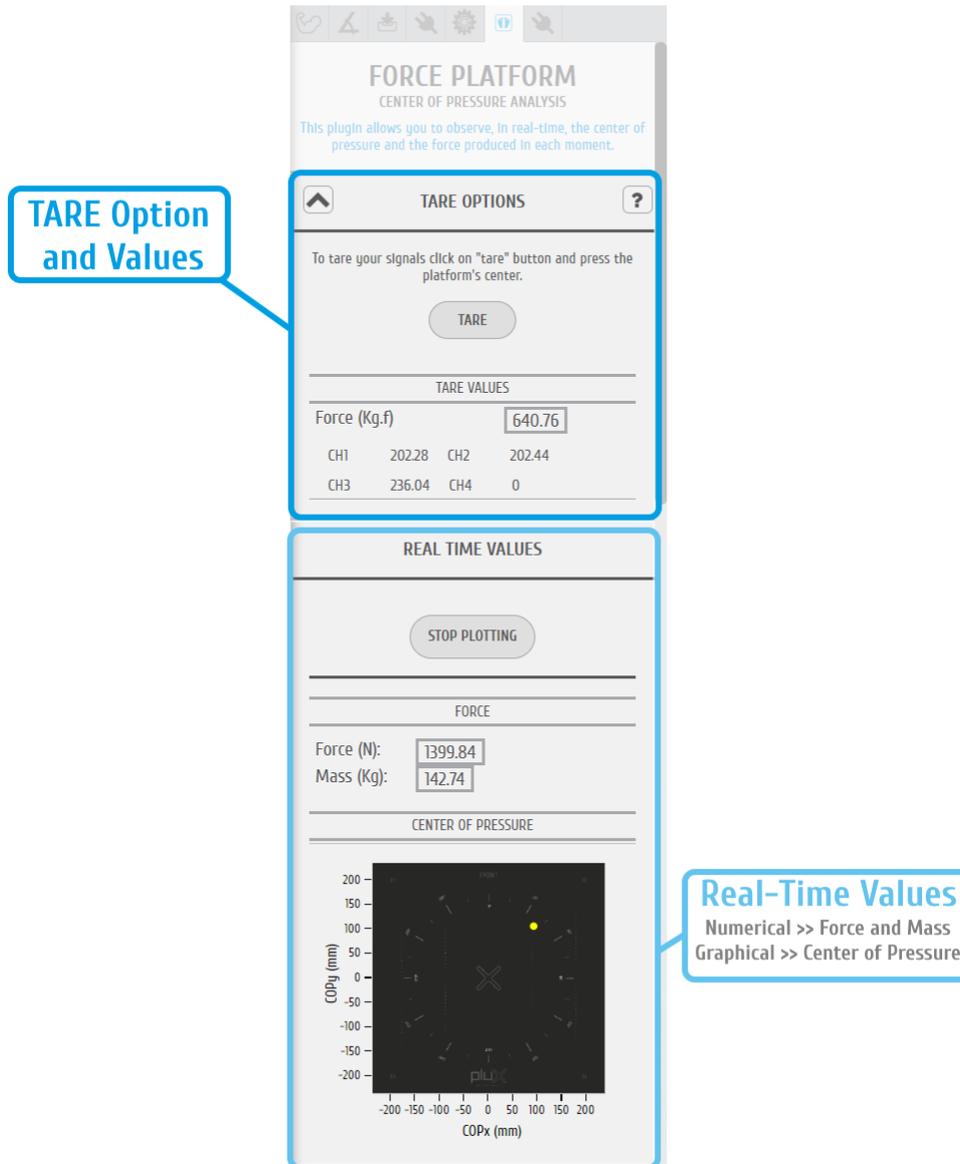


Figure 205: Force Platform Add-on with TARE Option and Real-Time Values.

VISUALISATION MODE

PROCESSING RESULTS

The Visualization version of the *Force Platform* add-on does not include configurable options, so, after pressing the “PROCESS” button the algorithm starts running autonomously and parameters such as maximum force (Fmax), force per second (FPS), flight time and jump height are presented.

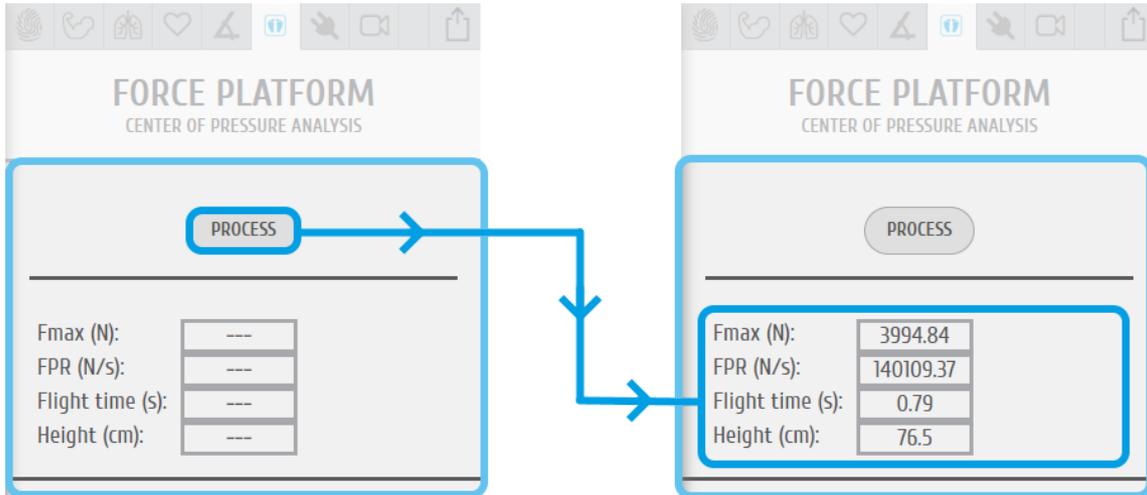


Figure 206: Force Platform Add-on with post-processing results.

4.19 fNIRS Analysis



fNIRS Analysis add-on

DESCRIPTION

The *fNIRS Analysis add-on* provides a way to process the RAW information gathered from fNIRS Explorer wearable device or from an independent fNIRS sensor.

The fNIRS technology is able to evaluate changes in the relative concentration of the oxygenated (**HbO**) and deoxygenated (**HbR**) hemoglobin, which, ultimately, contribute to the calculation of the **%SpO2** metric, i.e., the blood oxygen saturation.

Estimating the relative concentrations of **HbO** and **HbR** and the **%SpO2** is possible due a very specific property of the hemoglobin. The two forms (**HbO** and **HbR**) of this protein interact with **red** and **infrared** light in a characteristic way, presenting a distinct absorption spectrum.

This phenomenon is the one that supports the fNIRS technology, so, based on pair of **LEDs** (1 x **red** and 1 x **infrared**) and a light receptor (photodiode) the sensor is capable of detecting variations on the **red** and **infrared** light absorption/reflection which, as previously mentioned, are strongly related with the **HbO** and **HbR** concentrations.

Considering these variations, the real-time add-on, available in the signal acquisition mode (see *3.1 Signal Acquisition Mode*), processes and combines the two channels of fNIRS data (1/LED) and estimates the relative concentrations of **HbO** and **HbR**.

Then, based on the user choice (through the dropdown box in the graphical user interface) a virtual channel showing the variations of **HbO** or **HbR** is created and updated in real-time.

On the other hand, when the add-on verifies that data is also being acquired from an Accelerometer sensor, it applies an additional processing stage to exclude artefact-corrupted data avoiding an undesirable effect in the final processing results.

4.19.1 Hardware Configuration

The *fNIRS Analysis* add-on processes sensor data acquired with *biosignalsplux* fNIRS and Accelerometer sensors and also from the *fNIRS Explorer* wearable device:

biosignalsplux fNIRS Sensor:

<https://www.pluxbiosignals.com/products/functional-near-infrared-spectroscopy-fnirs-sensor>

biosignalsplux Accelerometer Sensor:

<https://www.pluxbiosignals.com/products/accelerometer-acc>

fNIRS Explorer Wearable:

<https://www.pluxbiosignals.com/products/fnirs-explorer-kit>

4.19.2 Add-on Configuration

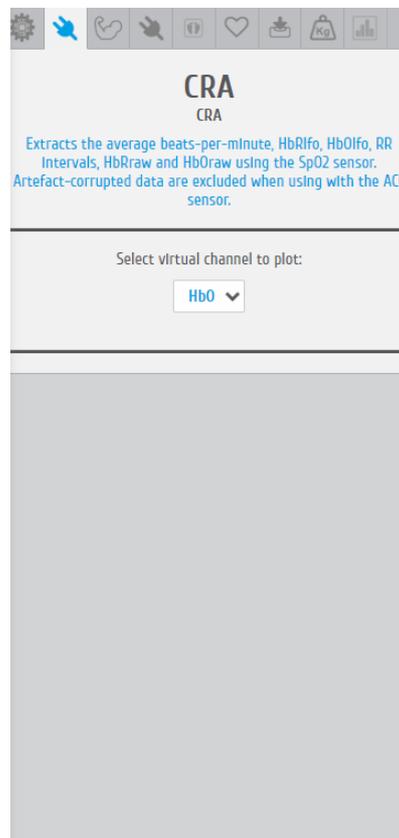


Figure 207: fNIRS add-on graphical user interface (GUI).

The current add-on doesn't need any special configuration, essentially, the user only needs to select the virtual channel that should be presented through the available dropdown input field.

Then, when the acquisition starts, the virtual channel is immediately generated and updated in real-time.

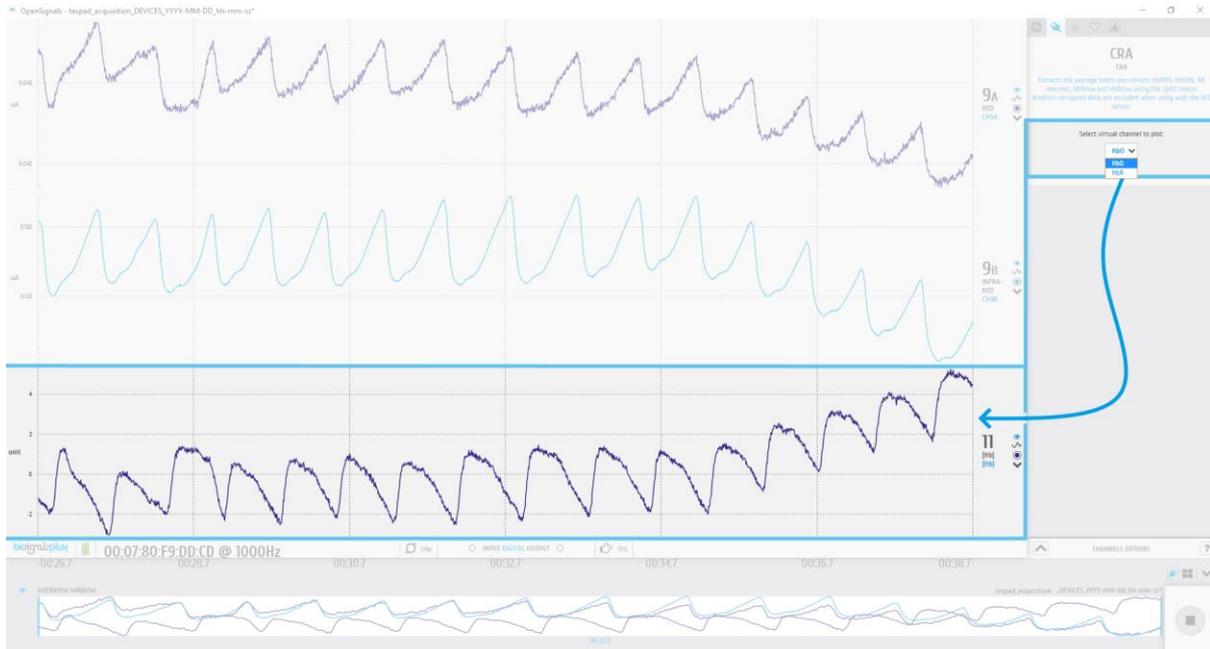


Figure 208: fNIRS add-on virtual channel.

4.20 Manual Measurement Input

Manual Measurement Input

DESCRIPTION

The annotation features provided by the *Event Annotation* add-on (see 4.6 *Event Annotation*) are extremely helpful when the researcher needs to register (through a keyboard press action) specific events that compose the experimental protocol.

With the *Event Annotation* add-on, each time the researcher flags an event, its time instant and label are stored in a file, in order to be accessed during the signal analysis stage.

The *Manual Measurement Input* add-on inherits the same principles and logic of the *Event Annotation* add-on, i.e., during a real-time data acquisition (see 3.1 *Signal Acquisition Mode*), the researcher can create an annotation that is stored with its corresponding time instant and label in a file.

However, in addition to the time instant and label information, the annotation created by the *Manual Measurement Input* add-on also contains the value of some parameters acquired with third-party devices.

With this approach, external device readings can be easily “integrated” in the signal analysis stage, considering that all real-time manual input annotations are synchronized with the signal collected by *OpenSignals (r)evolution* and accessible in the signal visualization mode (see 3.2 *Signal Visualization Mode*).

4.20.1 Hardware Configuration

This add-on works with every kind of signal being acquired with PLUX devices. It is not restricted to any specific sensor.

The annotated parameter values should be taken from third-party Oximeters, Blood Pressure Monitors or Glucometers, but being manual inputs provided by the researcher, no configuration steps are required in *OpenSignals (r)evolution*.

4.20.2 Add-on Configuration

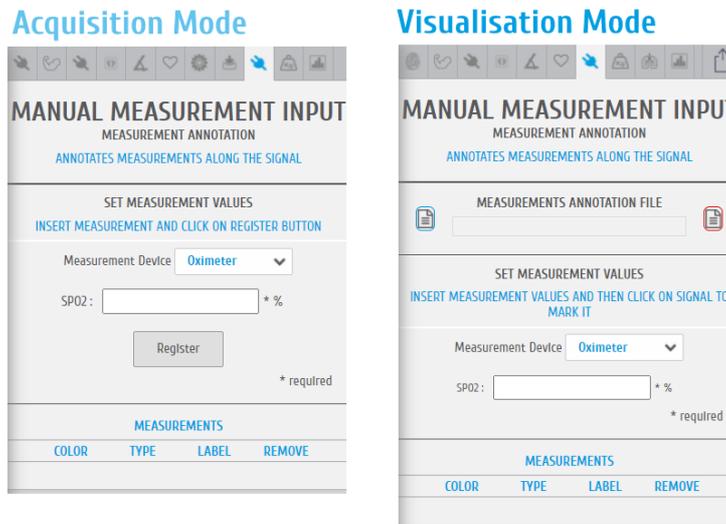


Figure 209: Manual Measurement Input add-on (acquisition mode @left and visualization mode @right).

ACQUISITION MODE MEASUREMENT DEVICE

In the drop-down input (Figure 210), it is provided a list containing all types of devices supported by the add-on, which will determinate the type and content of the annotation:

- a) **Oximeter**
 - %SpO2 | *Valid Range* >>> [50, 100] %
- b) **Blood Pressure Monitor**
 - Systolic Blood Pressure (mmHg) | *Valid Range* >>> [0, 300] mmHg
 - Diastolic Blood Pressure (mmHg) | *Valid Range* >>> [0, 300] mmHg
 - Heart Rate (BPM) | *Valid Range* >>> [0, 220] BPM
- c) **Glucometer**
 - Glucose Level (mg/dL) | *Valid Range* >>> [0, 2000] mg/dL

As highlighted on Figure 210, each option will be linked to different configurable/editable fields.

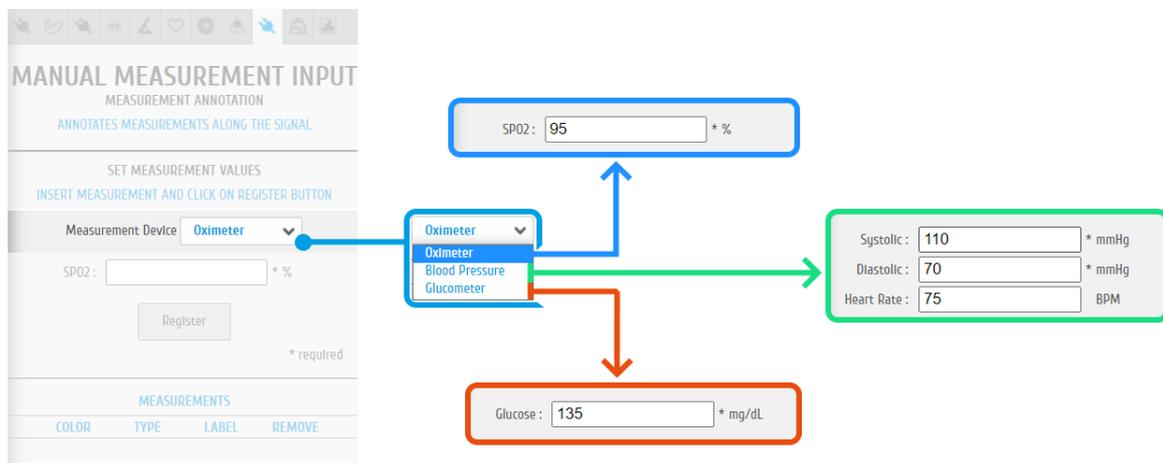


Figure 210: Manual Measurement Input add-on | measurement device options and editable fields [illustrative example]

REGISTER BUTTON

Once a new set of parameters from the third-party device is available, the researcher only needs to insert them in the add-on and click on the **Register** button, action that will immediately create a visual marker in the chart and stores the annotation time-instant, device type and parameters in a file.

As highlighted in the figure below, **1)** each device type as a specific color linked to the annotation (**blue >>> Oximeter | red >>> Blood Pressure Monitor | green >>> Glucometer**); **2)** annotations of the same type (belonging to the same device type) are distinguished by a unique incremental number and **3)** a previously created annotation can be removed/deleted.

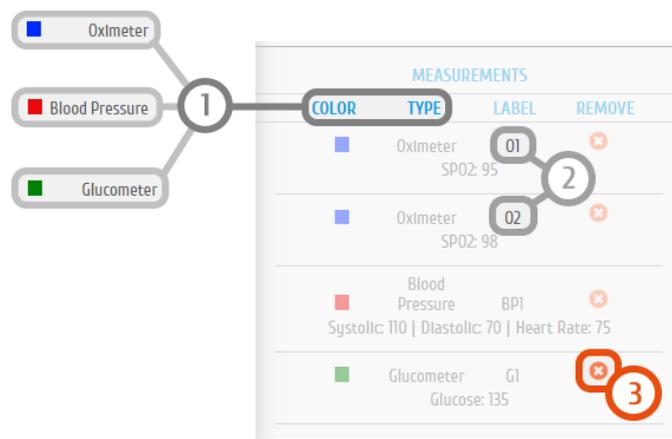


Figure 211: Manual Measurement Input add-on | measurements panel

Stopping the real-time acquisition will cause the generation of an additional .txt file where the registered annotations are stored (this file is in the same location where the acquired file is stored or in the predefined *OpenSignals (r)evolution* folder of the user computer).

It is also possible to manually trigger the storage of the annotations by clicking in the **SAVE MEASUREMENTS FILE** button.

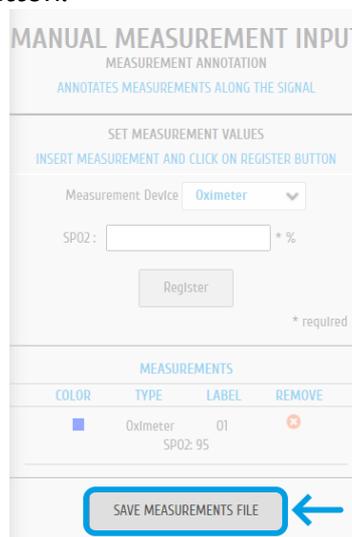


Figure 212: Manual Measurement Input add-on | save measurements button

VISUALISATION MODE

The real-time manual input annotations can be observed inside the Visualisation Mode of *OpenSignals (r)evolution*.

The user only needs to open the annotation file, as an example, on Windows systems it should be available at:

"C:\Users

After this loading stage the annotation data can be visualised by simply clicking on "DRAW MEASUREMENTS".

On the other hand, to make a backup of the Annotation file and start creating a new one from scratch (containing only the annotations added in the Visualization Mode), the "Reset Annotation File" button should be pressed.

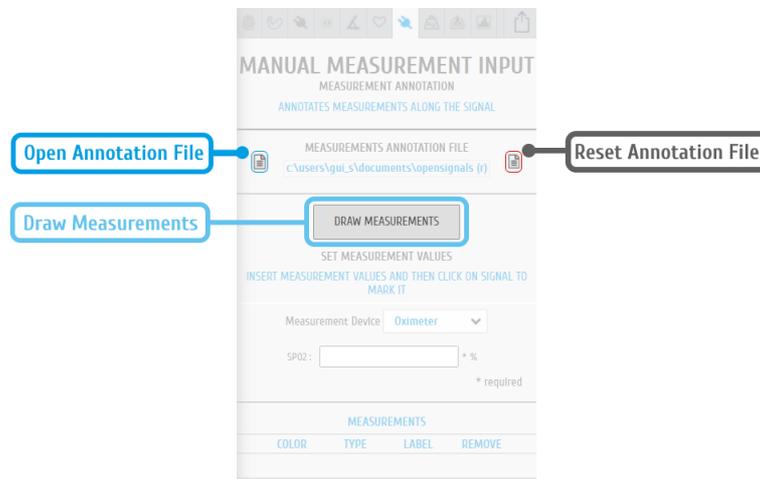


Figure 213: Manual Measurement Input add-on | graphical representation of the annotated events [visualisation mode]

An offline manual input parameter annotation can also be added to the acquired data (steps highlighted on Figure 214).

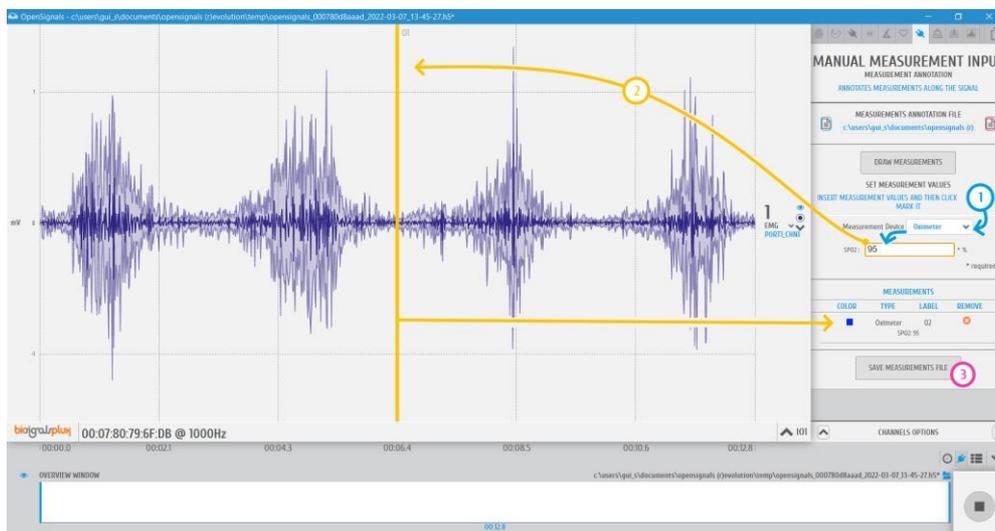


Figure 214: Manual Measurement Input add-on | adding offline measurements [visualisation mode] by 1) configuring an annotation; 2) draw an event through a mouse-click and finally 3) saving the new measurements into the annotations file.

5 Advanced Device Configuration

OpenSignals (r)evolution is compatible with all PLUX devices. It is recommended to review the following device specific configuration to obtain the best acquisition results.

This section provides you more information for the correct configuration of your PLUX device in *OpenSignals (r)evolution* to allow optimal signal acquisition quality. General and basic information on how to setup your device in *OpenSignals (r)evolution* (e.g. de/activating channels) is described in [2.2.4 Device Configuration](#).

5.1 BITalino

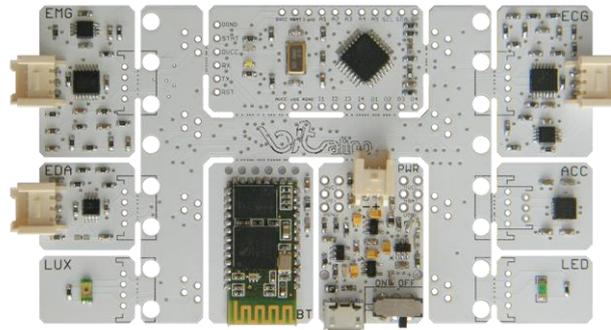


Figure 215: BITalino Board

NOTE

The following *BITalino* configuration is described for the *BITalino Board* kit. Keep in mind that, the channel configurations might be different if you're using a *BITalino Freestyle* or *Plugged* kit and that you may need to adjust them to your current setup.

CHANNEL CONFIGURATION

The following channel configuration is the standard configuration for *BITalino Boards* (see Table 1). Keep in mind, that this configuration might not apply to *BITalino Freestyle* or *Plugged* devices. The configuration of these devices has to be adjusted to your current device setup.

Table 1: Standard *BITalino Board* configuration.

CHANNEL	SENSOR	CHANNEL TYPE
A1	Electromyography	EMG
A2	Electrocardiography	ECG
A3	Electrodermal activity	EDA
A4	Accelerometer	ACC
A5	Light sensor	LUX
A6	Battery level	-
I1	-	-
I2	-	-
O1	LED	-
O2	-	-

All the available channel types for *BITalino* are listed in Table 2.

Table 2: BITalino channel types.

TYPE	SENSOR
RAW	Compatible with every sensor
ACC	Accelerometer
ECG	Electrocardiography
EDA	Electrodermal Activity
EMG	Electromyography
LUX	Light
CUSTOM	Customizable channel configuration

LED STATES

When using *BITalino*, the onboard LED provides visual information about the current device status (see Table 3).

Table 3: BITalino LED behavior.

LED	BEHAVIOR	DEVICE STATUS
White LED	Fading	Idle
White LED	Blinking	Acquiring signals
Red LED (MCU)	On	Low battery/Channel A6 is not connected (<i>BITalino Freestyle</i> and <i>BITalino Plugged</i>)
Red LED (PWR Block)	On	Charging battery (<i>BITalino</i> needs to be turned off)

Keep in mind that the red LED on the MCU board might be permanently turned on when using a *BITalino Freestyle* or *Plugged* devices as the channel A6 is not connected by default on these devices.

RESOLUTION

The resolution is pre-set for *BITalino* devices. When 5 or 6 channels are actively acquiring signals, channels 1,2,3, and 4 are acquiring with 10-bit resolution while channels 5 and 6 are acquiring with 6-bit resolution.

When 4 or less channels are actively acquiring signals, all active channels are acquiring with 10-bit resolution including channels 5 and 6.

Digital inputs and outputs are acquiring with 1-bit resolution.

SAMPLING RATE

BITalino supports sampling rates of 1, 10, 100 and 1000Hz. However, in *OpenSignals (r)evolution* only 10, 100 and 1000Hz can be selected.

NOTE

Higher resolutions & sampling rates lead to higher quality signal acquisitions but are linked to higher consumption of system resources & higher probability of Bluetooth transmissions issues which may lead to signal delays in the real-time acquisition mode.

We recommend the usage of a PLUX proven [Bluetooth dongle](#) to prevent Bluetooth connection related acquisition issues.

5.2 BITalino (r)evolution BT (Bluetooth)

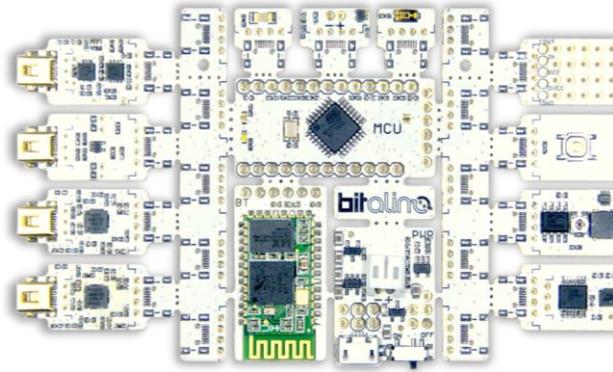


Figure 216: BITalino (r)evolution Board BT

NOTE

The following *BITalino (r)evolution* configuration is described for the *BITalino (r)evolution Board BT* kit. Keep in mind that, the channel configurations might be different if you're using a *BITalino (r)evolution Freestyle BT* or *Plugged BT* kit and that you may need to adjust them to your current setup.

CHANNEL CONFIGURATION

The following channel configuration is the standard configuration for *BITalino (r)evolution Boards* (see Table 4). Please keep in mind, that this configuration might not apply to *BITalino (r)evolution Freestyle* or *Plugged* devices. The configuration of these devices has to be adjusted to your current device setup.

Table 4: Standard BITalino (r)evolution Board BT configuration.

CHANNEL	SENSOR	CHANNEL TYPE
A1	Electromyography	EMG
A2	Electrocardiography	ECG
A3	Electrodermal activity	EDA
A4	Electroencephalography	EEG
A5	Accelerometer	ACC
A6	Light sensor	LUX
I1	Button	-
I2	-	-
O1	LED	-
O2	Buzzer	-

All the available channel types for *BITalino (r)evolution* are listed in Table 5.

Table 5: *BITalino (r)evolution* channel types.

TYPE	SENSOR
RAW	Compatible with every sensor
ACC	Accelerometer
ECG	Electrocardiography
EDA	Electrodermal Activity
EEG	Electroencephalography
EMG	Electromyography
LUX	Light Sensor
TEMP	Temperature
CO2	CO ₂ Evaluation
DIA	Blood pressure reader (BPR) - diastolic sensor values
EKG	Electrogastrography
EOG	Electrooculography
GLUC	Glucose Meter Reader
HR	SpO ₂ reader (OSL) – heart rate sensor values
OXI	Oximeter
PULSEOXI	SpO ₂ reader (OSL) – oxygen saturation sensor values
RESP	Respiration sensor (RIP & PZT)
SYS	Blood pressure reader (BPR) – systolic sensor values
TEMP	Temperature sensor (TMP)
TEMPw	High-Definition Temperature
CUSTOM	Customizable channel configuration

LED STATES

When using *BITalino (r)evolution*, the onboard led provides visual information about the current device status (see Table 6).

Table 6: *BITalino (r)evolution* LED behavior.

LED	BEHAVIOR	DEVICE STATUS
White LED	Fading	Idle
White LED	Blinking	Acquiring signals
Red LED (MCU)	On	Low battery
Red LED (PWR Block)	On	Charging battery (<i>BITalino (r)evolution</i> needs to be turned off)

RESOLUTION

Like described for *BITalino*, the resolution is pre-set for *BITalino (r)evolution* devices. When 5 or 6 channels are actively acquiring signals, channels 1,2,3, and 4 are acquiring with 10-bit resolution while channels 5 and 6 are acquiring with 6-bit resolution.

When 4 or less channels are actively acquiring signals, all active channels are acquiring with 10-bit resolution including channels 5 and 6.

Digital inputs and outputs are acquiring with 1-bit resolution.

SAMPLING RATE

BITalino supports sampling rates of 1, 10, 100 and 1000Hz. However, in *OpenSignals (r)evolution* only 10, 100 and 1000Hz can be selected.

NOTE

Higher resolutions & sampling rates lead to higher quality signal acquisitions but are linked to higher consumption of system resources & higher probability of Bluetooth transmissions issues which may lead to signal delays in the real-time acquisition mode.

We recommend the usage of a PLUX proven [Bluetooth dongle](#) to prevent Bluetooth connection related acquisition issues.

DATASHEETS

BITalino Board:

<https://support.pluxbiosignals.com/wp-content/uploads/2021/11/revolution-bitalino-board-kit-datasheet.pdf>

BITalino Plugged:

<https://support.pluxbiosignals.com/wp-content/uploads/2021/11/revolution-bitalino-plugged-kit-datasheet.pdf>

5.3 BITalino (r)evolution BLE (Bluetooth Low Energy)

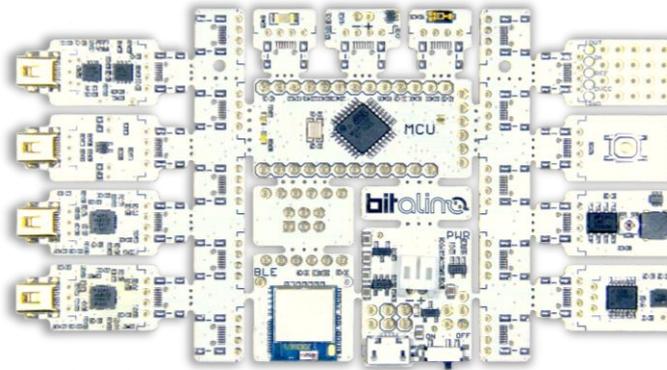


Figure 217: BITalino (r)evolution Board BLE

NOTE

The following *BITalino (r)evolution* configuration is described for the *BITalino (r)evolution Board BLE* kit. Keep in mind that, the channel configurations might be different if you're using a *BITalino (r)evolution Freestyle BLE* or *Plugged BLE* kit and that you may need to adjust them to your current setup.

NOTE

The *BITalino (r)evolution BLE* (Bluetooth Low Energy) devices are currently not supported on Mac OS and Linux systems.

CHANNEL CONFIGURATION

The previously defined channel configuration for *BITalino (r)evolution BT* (Table 4) is replicated by *BITalino (r)evolution BLE*. Please keep in mind, that this configuration might not apply to *BITalino (r)evolution Freestyle* or *Plugged* devices. The configuration of these devices has to be adjusted to your current device setup.

All the available channel types for *BITalino (r)evolution* are listed in Table 5

LED STATES

When using *BITalino (r)evolution*, the onboard led provides visual information about the current device status (see Table 6).

RESOLUTION

The resolution is pre-set for *BITalino (r)evolution* devices. When 5 or 6 channels are actively acquiring signals, channels 1,2,3, and 4 are acquiring with 10-bit resolution while channels 5 and 6 are acquiring with 6-bit resolution.

When 4 or less channels are actively acquiring signals, all active channels are acquiring with 10-bit resolution including channels 5 and 6.

Digital inputs and outputs are acquiring with 1-bit resolution.

SAMPLING RATE

BITalino supports sampling rates of 1, 10, 100 and 1000Hz. However, in *OpenSignals (r)evolution* only 10, 100 and 1000Hz can be selected.

NOTE

Higher resolutions & sampling rates lead to higher quality signal acquisitions but are linked to higher consumption of system resources & higher probability of Bluetooth transmissions issues which may lead to signal delays in the real-time acquisition mode.

We recommend the usage of a PLUX proven [Bluetooth Low Energy dongle](#) to prevent Bluetooth connection related acquisition issues.

Due to limitations associated with the current Bluetooth Low Energy capabilities, please keep in mind the following aspects when using a BITalino (r)evolution BLE device for data acquisition:

- Only a single channel can be acquired at the maximum sampling rate (1000Hz)
- To acquire data from all channels simultaneously a sampling rate of 100Hz or lower needs to be selected
- BITalino (r)evolution BLE devices currently are only supported on Windows
- A Bluegiga BLED112 BLE dongle is required ([BLE dongle](#))

DATASHEETS

BITalino Board:

<https://support.pluxbiosignals.com/wp-content/uploads/2021/11/revolution-bitalino-board-kit-datasheet.pdf>

BITalino Plugged:

<https://support.pluxbiosignals.com/wp-content/uploads/2021/11/revolution-bitalino-plugged-kit-datasheet.pdf>

5.4 biosignalsplx



Figure 218: 4 channel (left) and 8 channel (right) biosignalsplx HUBs.

CHANNEL CONFIGURATION

biosignalsplx devices have no specific channel configuration by default. The channel configuration should be done according to your setup. All available channel types for all compatible sensors are listed in section 5.14.1 *biosignalsplx*.

RESOLUTION

biosignalsplx devices support 8-bit, 12-bit (in some models) and 16-bit resolutions.

SAMPLING RATE

If using *OpenSignals (r)evolution* versions from December 2017 (and newer) and *biosignalsplx* devices with firmware version 3.5 or higher, it is even possible to select higher sampling rates of up to 4000Hz.

However, note that high sampling rate for all channels are only supported for sampling rates of up to 3000Hz, while 4000Hz is limited to a maximum of 3 channels only.

NOTE

Higher resolutions & sampling rates lead to higher quality signal acquisitions but are linked to higher consumption of system resources & higher probability of Bluetooth transmissions issues which may lead to signal delays in the real-time acquisition mode.

We recommend the usage of a PLUX proven [Bluetooth dongle](#) to prevent Bluetooth connection related acquisition issues.

INTERNAL MEMORY

biosignalsplux hubs with internal memory can be used for offline acquisitions which do not require an active Bluetooth connection during data acquisition. For detailed instructions on how to use the internal memory of your device, please check section 3.3 *Data Logging*.

DOCUMENTATION

Datasheets and user manuals for your *biosignalsplux* device can be found on the following website:

<https://www.pluxbiosignals.com/collections/biosignalsplux>

5.5 biosignalsplux Hybrid-8



Figure 219: biosignalsplux Hybrid-8 HUB.

CHANNEL CONFIGURATION

biosignalsplux Hybrid-8 device has no specific channel configuration by default. The channel configuration should be done according to your setup. All available channel types for all compatible sensors are listed in section 5.14.1 *biosignalsplux*.

RESOLUTION

biosignalsplux devices support up to 16-bit (analog sensors) and 24-bit (digital sensors) resolutions.

SAMPLING RATE

The first commercialized version of the *biosignalsplux Hybrid-8* system supports real-time acquisitions with 8 analog sensors (16 bits) streaming data simultaneously up to 3000Hz.

On the other hand, while dealing with 8 digital sensors (24 bits) the maximum sampling rate with this configuration will be 500Hz.

Higher sampling rates, for example, 4000Hz for the analog sensors will be possible if fewer channels (3 or less active channels) are used simultaneously.

NOTE

Higher resolutions & sampling rates lead to higher quality signal acquisitions but are linked to higher consumption of system resources & higher probability of Bluetooth transmissions issues which may lead to signal delays in the real-time acquisition mode.

We recommend the usage of a PLUX proven [Bluetooth dongle](#) to prevent Bluetooth connection related acquisition issues.

INTERNAL MEMORY

biosignalsplux Hybrid-8 hub has an internal memory with 32GB, which can be used for offline acquisitions that won't require an active Bluetooth connection during data acquisition. For detailed instructions on how to use the internal memory of your device, please check section *3.3 Data Logging*.

USB DATA TRANSFER

biosignalsplux Hybrid-8 hub also supports the download of data acquisition sessions, stored in the internal memory of the device, through a dedicated **USB Adapter** (for further details about this adapter, please, contact the team through the communication channels available at <https://support.pluxbiosignals.com/>).

The download procedure is identical to the one described on section *3.3.2.2 Download via Fast USB Data Transfer Cable*.

In order to guarantee the recognition of the *biosignalsplux Hybrid-8* device by a Windows computer, some drivers need to be installed.

The driver installer is available in the following page:

https://downloads.plux.info/OpenSignals/misc/drivers/Hybrid-8_usb_driver_installer.exe

DOCUMENTATION

Datasheets and user manuals for your *biosignalsplux Hybrid-8* device will be available on the following website:

<https://www.pluxbiosignals.com/collections/biosignalsplux>

5.6 biosignalsplux Force



Figure 220: biosignalsplux Force

NOTE

The *biosignalsplux Force* is a force platform with real-time streaming and data logging capabilities which uses a 4-channel *biosignalsplux* device. It is recommended to also check the following sections to take full advantage of all the force platform features:

- Data logging (see 3.3 Data Logging)
- Force platform add-on (see 4.18 Force Platform)
- *biosignalsplux* configuration (see 5.4 *biosignalsplux*)

CHANNEL CONFIGURATION

The available channel types for the *biosignalsplux* device being used with the force platform should be set to the *PLATFORM* sensor type.

RESOLUTION

biosignalsplux devices support 8-bit, 12-bit (in some models) and 16-bit resolutions.

SAMPLING RATE

If using *OpenSignals (r)evolution* versions from December 2017 (and newer) and *biosignalsplux* devices with firmware version 3.5 or higher, it is even possible to select higher sampling rates of up to 4000Hz.

However, note that high sampling rate for all channels are only supported for sampling rates of up to 3000Hz, while 4000Hz is limited to a maximum of 3 channels only.

NOTE

Higher resolutions & sampling rates lead to higher quality signal acquisitions but are linked to higher consumption of system resources & higher probability of Bluetooth transmissions issues which may lead to signal delays in the real-time acquisition mode.

We recommend the usage of a PLUX proven [Bluetooth dongle](#) to prevent Bluetooth connection related acquisition issues.

INTERNAL MEMORY

biosignalsplux hubs with internal memory can be used for offline acquisitions which do not require an active Bluetooth connection during data acquisition. For detailed instructions on how to use the internal memory of your device, please check section 3.3 *Data Logging*.

DOCUMENTATION

Datasheets and user manuals for your *biosignalsplux* device can be found on the following website:

<https://www.pluxbiosignals.com/collections/biosignalsplux>

5.7 biosignalsplux Solo [new designation for OpenBAN]



Figure 221: biosignalsplux Solo

NOTE

biosignalsplux Solo is a single channel acquisition device which can be used with any *biosignalsplux* sensor except the SpO2 and fNIRS sensors.

Also, this device is not compatible with any of the accessory devices which require a digital port, taking into consideration that this device does not have a digital port.

NOTE

The usage of some sensors is limited or not possible due to the limited number of inputs of this device, as some sensors require several channels to work properly (e.g., external accelerometer, goniometer).

CHANNEL CONFIGURATION

biosignalsplux Solo devices have a specific channel configuration with the exception of channel 2, which is the channel where *biosignalsplux* sensors are connected to. Therefore, only the configuration of channel 2 has to be adjusted to the connected sensor.

Additionally, the *biosignalsplux Solo* comes with an integrated and pre-connected microphone, triaxial accelerometer, and triaxial magnetometer.

The channel configuration must be done according to the internal setup of the *biosignalsplux Solo* and your additional sensor; see Table 7.

Table 7: Channel configuration of the *biosignalsplux Solo*.

CHANNEL	SENSOR	CHANNEL TYPE
1	Microphone	MICRO
2	biosignalsplux sensor	[DEPENDS ON CONNECTED SENSOR]
3	Accelerometer – X-Axis	ACC_X
4	Accelerometer – Y-Axis	ACC_Y
5	Accelerometer – Z-Axis	ACC_Z
6	Magnetometer – X-Axis	MAG_X
7	Magnetometer – Y-Axis	MAG_Y
8	Magnetometer – Z-Axis	MAG_Z

The available channel types are listed in section 5.14.1 *biosignalsplux*.

NOTE

This sensor has a virtual reference; there is no need to connect an additional reference electrode.

RESOLUTION

biosignalsplux Solo supports 8-bit, 12-bit (in some models) and 16-bit resolutions.

SAMPLING RATE

The sampling rate can be set to 10Hz, 20Hz, 50Hz, 100Hz, 200Hz, 300Hz, ..., 1000Hz.

NOTE

Higher resolutions & sampling rates lead to higher quality signal acquisitions but are linked to higher consumption of system resources & higher probability of Bluetooth transmissions issues which may lead to signal delays in the real-time acquisition mode.

We recommend the usage of a PLUX proven [Bluetooth dongle](#) to prevent Bluetooth connection related acquisition issues.

DOCUMENTATION

Datasheets and user manuals for your device can be found on the following website:

<https://www.pluxbiosignals.com/collections/biosignalsplux>

5.8 MuscleBAN



Figure 222: MuscleBAN.

NOTE

MuscleBAN devices are closed devices with prewired, internal analog ports. Other *biosignalsplux* sensors cannot be connected to this device.

CHANNEL CONFIGURATION

MuscleBAN devices have a specific channel configuration by default. The configuration inside *OpenSignals (r)evolution* should be done as displayed in Table 8.

Table 8: MuscleBAN channel configuration.

CHANNEL	SENSOR	CHANNEL TYPE
1	Electromyography	EMG
2	Accelerometer (x-axis)	XYZ
3	Accelerometer (y-axis)	
4	Accelerometer (z-axis)	
5	Magnetometer (x-axis)	
6	Magnetometer (y-axis)	
7	Magnetometer (z-axis)	

RESOLUTION

MuscleBAN devices support 8-bit and 16-bit resolutions.

SAMPLING RATE

The sampling rate can be set to 50Hz, 80Hz, 160 Hz, 200Hz, 400Hz, ..., 1000Hz. However, note that these devices stream all samples in envelopes to your computer with a streaming rate of up to 50Hz.

NOTE

Higher resolutions & sampling rates lead to higher quality signal acquisitions but are linked to higher consumption of system resources & higher probability of Bluetooth transmissions issues which may lead to signal delays in the real-time acquisition mode.

We recommend the usage of a PLUX proven [Bluetooth Low Energy dongle](#) to prevent Bluetooth connection related acquisition issues.

DOCUMENTATION

Datasheets and user manuals for your device can be found on the following website:

<https://www.pluxbiosignals.com/collections/biosignalsplux>

5.9 MotionBAN



Figure 223 : MotionBAN

NOTE

MotionBAN devices are closed devices with prewired, internal analog ports. Other *biosignalsplux* sensors cannot be connected to this device.

CHANNEL CONFIGURATION

MotionBAN devices have a specific channel configuration by default. The configuration inside *OpenSignals (r)evolution* should be done as displayed in Table 9.

Table 9: *MotionBAN* channel configuration.

CHANNEL	SENSOR	CHANNEL TYPE
1	Accelerometer (x-axis)	XYZ
2	Accelerometer (y-axis)	
3	Accelerometer (z-axis)	
4	Magnetometer (x-axis)	
5	Magnetometer (y-axis)	
6	Magnetometer (z-axis)	

RESOLUTION

MotionBAN supports 8-bit and 16-bit resolutions.

SAMPLING RATE

The sampling rate can be set to 100Hz, 200Hz, 300Hz, ..., 800Hz.

NOTE

Higher resolutions & sampling rates lead to higher quality signal acquisitions but are linked to higher consumption of system resources & higher probability of Bluetooth transmissions issues which may lead to signal delays in the real-time acquisition mode.

We recommend the usage of a PLUX proven [Bluetooth dongle](#) to prevent Bluetooth connection related acquisition issues.

INTERNAL MEMORY

MotionBAN devices have an internal memory which can be used for offline acquisitions without the need of an active Bluetooth connection during data acquisition. For detailed instructions on how to use the internal memory of your device, please check section 3.3 *Data Logging*.

DOCUMENTATION

Datasheets and user manuals for your devices can be found on the following website:

<https://www.pluxbiosignals.com/collections/biosignalsplux>

5.10 CardioBAN



Figure 224: CardioBAN

NOTE

CardioBAN devices are closed devices with prewired, internal analog ports. Other *biosignalsplus* sensors cannot be connected to this device.

CHANNEL CONFIGURATION

CardioBAN devices have a specific channel configuration by default. The configuration inside *OpenSignals (r)evolution* should be done as displayed in Table 10.

Table 10: CardioBAN channel configuration.

CHANNEL	SENSOR	CHANNEL TYPE
1	Electrocardiography	ECG
2	Accelerometer (x-axis)	XYZ
3	Accelerometer (y-axis)	
4	Accelerometer (z-axis)	

RESOLUTION

CardioBAN supports 8-bit and 16-bit resolutions.

SAMPLING RATE

If using *OpenSignals (r)evolution* versions from December 2017 (and newer) and *biosignalsplux* devices with firmware version 3.5 or higher, it is even possible to select higher sampling rates of up to 4000Hz.

However, note that high sampling rate for all channels are only supported for sampling rates of up to 3000Hz, while 4000Hz is limited to a maximum of 3 channels only.

NOTE

Higher resolutions & sampling rates lead to higher quality signal acquisitions but are linked to higher consumption of system resources & higher probability of Bluetooth transmissions issues which may lead to signal delays in the real-time acquisition mode.

We recommend the usage of a PLUX proven [Bluetooth dongle](#) to prevent Bluetooth connection related acquisition issues.

DOCUMENTATION

Datasheets and user manuals for your device can be found on the following website:

<https://www.pluxbiosignals.com/collections/biosignalsplux>

5.11 RespiBAN Explorer



Figure 225: RespiBAN Explorer

NOTE

RespiBAN Explorer devices are closed devices with prewired, internal analog ports. Other *biosignalsflux* sensors cannot be connected to this device.

CHANNEL CONFIGURATION

RespiBAN Explorer devices have a specific channel configuration by default. The configuration inside *OpenSignals (r)evolution*.

Table 11: *RespiBAN Explorer* channel configuration.

CHANNEL	SENSOR	CHANNEL TYPE
1	Respiration	RESP
2	Accelerometer (x-axis)	XYZ
3	Accelerometer (y-axis)	
4	Accelerometer (z-axis)	

RESOLUTION

RespiBAN Explorer supports 8-bit and 16-bit resolutions.

SAMPLING RATE

The sampling rate can be set to 100Hz, 200Hz, 300Hz, ..., 1000Hz.

NOTE

Higher resolutions & sampling rates lead to higher quality signal acquisitions but are linked to higher consumption of system resources & higher probability of Bluetooth transmissions issues which may lead to signal delays in the real-time acquisition mode.

We recommend the usage of a PLUX proven [Bluetooth dongle](#) to prevent Bluetooth connection related acquisition issues.

DOCUMENTATION

Datasheets and user manuals for your device can be found on the following website:

<https://www.pluxbiosignals.com/collections/biosignalsplux>

5.12 RespiBAN Researcher



Figure 226: RespiBAN Researcher

NOTE

RespiBAN Researcher devices are closed devices with prewired, internal analog ports. Other *biosignalsplux* sensors cannot be connected to this device.

CHANNEL CONFIGURATION

RespiBAN Researcher devices have a specific channel configuration by default. The configuration inside *OpenSignals (r)evolution* should be done as displayed in Table 12.

Table 12: *RespiBAN Researcher* channel configuration.

CHANNEL	SENSOR	CHANNEL TYPE
1	Respiration	RESP
2	Accelerometer (x-axis)	XYZ
3	Accelerometer (y-axis)	
4	Accelerometer (z-axis)	

RESOLUTION

RespiBAN Researcher supports 8-bit and 16-bit resolutions.

SAMPLING RATE

If using *OpenSignals (r)evolution* versions from December 2017 (and newer) and *biosignalsplux* devices with firmware version 3.5 or higher, it is even possible to select higher sampling rates of up to 4000Hz.

However, note that high sampling rate for all channels are only supported for sampling rates of up to 3000Hz, while 4000Hz is limited to a maximum of 3 channels only.

NOTE

Higher resolutions & sampling rates lead to higher quality signal acquisitions but are linked to higher consumption of system resources & higher probability of Bluetooth transmissions issues which may lead to signal delays in the real-time acquisition mode.

We recommend the usage of a PLUX proven [Bluetooth dongle](#) to prevent Bluetooth connection related acquisition issues.

DOCUMENTATION

Datasheets and user manuals for your device can be found on the following website:

<https://www.pluxbiosignals.com/collections/biosignalsplux>

5.13 RespiBAN Professional



Figure 227: RespiBAN Professional

CHANNEL CONFIGURATION

RespiBAN Professional devices have 4 prewired channels which are connected to the onboard sensors and 4 generic channels which can be used to connect *biosignalsplux* sensors to this device.

The configuration for the first 4 channels have to be done according to the configuration of the prewired sensors while the last 4 have to be done according to your sensor specific setup using the available channel types for *biosignalsplux* sensors listed in section 5.14.1 *biosignalsplux*.

Table 13: RespiBAN Professional channel configuration.

CHANNEL	SENSOR	CHANNEL TYPE
1	Respiration (inductive)	RESP
2	Accelerometer (x-axis)	XYZ
3	Accelerometer (y-axis)	
4	Accelerometer (z-axis)	
5	Generic	Sensor specific selection needed.
6		
7		
8		

RESOLUTION

RespiBAN Professional supports 8-bit and 16-bit resolutions.

SAMPLING RATE

If using *OpenSignals (r)evolution* versions from December 2017 (and newer) and *biosignalsplux* devices with firmware version 3.5 or higher, it is even possible to select higher sampling rates of up to 4000Hz.

However, note that high sampling rate for all channels are only supported for sampling rates of up to 3000Hz, while 4000Hz is limited to a maximum of 3 channels only.

NOTE

Higher resolutions & sampling rates lead to higher quality signal acquisitions but are linked to higher consumption of system resources & higher probability of Bluetooth transmissions issues which may lead to signal delays in the real-time acquisition mode.

We recommend the usage of a PLUX proven [Bluetooth dongle](#) to prevent Bluetooth connection related acquisition issues.

DOCUMENTATION

Datasheets and user manuals for your device can be found on the following website:

<https://www.pluxbiosignals.com/collections/biosignalsplux>

5.14 Compatible Sensors & Channel Types

OpenSignals (r)evolution is compatible with all *biosignalsplux* & *BITalino* sensors. See below the links to access more information about these sensors and all sensor datasheets.

5.14.1 biosignalsplux

The following *biosignalsplux* sensors are compatible with *OpenSignals (r)evolution*. The available channel types (detailed configuration instructions presented in section 2.2.4.6 *Channel Configuration*) for each sensor can be found in the table below (Table 14), followed by links to each sensor's datasheet and website can be found below this table.

Table 14: Channel types of compatible biosignalsplux sensors.

TYPE	SENSOR
BVP	Blood volume pulse (BVP)
CUSTOM	Customizable channel configuration
DIA	Blood pressure reader (BPR) - diastolic sensor values
ECG	Electrocardiography (ECG)
EDA	Electrodermal Activity (EDA)
EEG	Electroencephalography (EEG)
EMG	Electromyography (EMG)
PLATFORM	Force platform (load cells)
FSR-X	Force (FSR) sensor (X = type of sensor (1, 2, 3 or 4))
GLUC	Glucose Meter Reader
GONIO	Goniometer
HANDGR	Hand load cell
HR	SpO2 reader (OSL) – heart rate sensor values
LOADCELL	Load cell
LUX	Light sensor
MICRO	Built-in microphone of the biosignalsplux Solo device
OXI	SpO2 sensor (with or without clip) ^A
PULSEOXI	SpO2 reader (OSL) – oxygen saturation sensor values
RAW	Raw digital signal; compatible with every sensor
RESP	Respiration sensor (RIP & PZT)
SYS	Blood pressure reader (BPR) – systolic sensor values
SpO2.ARM	SpO2 sensor (with or without clip) for measurements on the arm ^A
TEMP	Temperature sensor (TMP)
EDA.ARM	Electrodermal Activity (EDA) for measurements on the arm
SpO2.FING	SpO2 sensor (with or without clip) for measurements on the finger ^A
fNIRS	Functional Near-Infrared Spectroscopy (fNIRS) ^A
XYZ	Accelerometer (ACC) ^B

A: not compatible with biosignalsplux Solo; B: only one axis when being used with biosignalsplux Solo

The *CUSTOM* channel type has no pre-set channel configuration allowing you to manually select the gain, offset and unit of the channel. This channel configuration can be helpful when using 3rd party or custom sensors.

Accelerometer (ACC):

Website: <https://www.pluxbiosignals.com/products/accelerometer-acc>

Datasheet: https://support.pluxbiosignals.com/wp-content/uploads/2021/11/Accelerometer_ACC_Datasheet.pdf

Blood Pressure Reader (BPR):

Datasheet: [https://www.downloads.plux.info/OpenSignals/misc/user-manual/Blood_Pressure_Reader_\(BPR\)_Datasheet.pdf](https://www.downloads.plux.info/OpenSignals/misc/user-manual/Blood_Pressure_Reader_(BPR)_Datasheet.pdf)

Blood Volume Pulse (BVP):

Website: <https://www.pluxbiosignals.com/products/blood-volume-pulse-bvp-sensor>

Datasheet: https://support.pluxbiosignals.com/wp-content/uploads/2021/11/Blood_Volume_Pulse_BVP_Datasheet.pdf

Electrocardiogram (ECG):

Website: <https://www.pluxbiosignals.com/products/electrocardiography-ecg-sensor-1>

Datasheet: https://support.pluxbiosignals.com/wp-content/uploads/2021/11/Electrocardiography_ECG_Datasheet.pdf

Electrodermal Activity (EDA):

Website: <https://www.pluxbiosignals.com/products/electrodermal-activity-eda-sensor-1>

Datasheet: https://support.pluxbiosignals.com/wp-content/uploads/2021/11/Electrodermal_Activity_EDA_Datasheet.pdf

Electroencephalography (EEG):

Website: <https://www.pluxbiosignals.com/products/electroencephalography-eeeg>

Datasheet: https://support.pluxbiosignals.com/wp-content/uploads/2021/11/Electroencephalography_EEG_Datasheet.pdf

Electromyography (EMG):

Website: <https://www.pluxbiosignals.com/products/electromyography-emg>

Datasheet: <https://support.pluxbiosignals.com/wp-content/uploads/2021/10/biosignalsplux-Electromyography-EMG-Datasheet.pdf>

Functional Near-Infrared Spectroscopy (fNIRS):

Website: <https://www.pluxbiosignals.com/products/functional-near-infrared-spectroscopy-fnirs-sensor>
Datasheet: <https://support.pluxbiosignals.com/wp-content/uploads/2021/10/biosignalsplux-FNIRS-Datasheet.pdf>

Foot Switch (FSW):

Website: <https://www.pluxbiosignals.com/products/foot-switch>
Datasheet: <https://support.pluxbiosignals.com/wp-content/uploads/2021/10/biosignalsplux-Foot-switch-Datasheet.pdf>

Force (FSR):

Website: <https://www.pluxbiosignals.com/products/force-fsr>
Datasheet: https://support.pluxbiosignals.com/wp-content/uploads/2021/11/Pressure_FSR_Datasheet.pdf

Force Platform:

Website: <https://www.pluxbiosignals.com/products/force-platform>
Datasheet: https://support.pluxbiosignals.com/wp-content/uploads/2021/11/Force_Platform_Datasheet.pdf

Glucose Meter Reader (GMR):

Datasheet: [https://www.downloads.plux.info/OpenSignals/misc/user-manual/Glucose_Meter_Reader_\(GMR\)_Datasheet.pdf](https://www.downloads.plux.info/OpenSignals/misc/user-manual/Glucose_Meter_Reader_(GMR)_Datasheet.pdf)

Goniometer (GON):

Website: <https://www.pluxbiosignals.com/products/goniometer-gon>
Datasheet: https://support.pluxbiosignals.com/wp-content/uploads/2021/12/Goniometer_GON_Datasheet.pdf

Light (LUX):

Website: <https://www.pluxbiosignals.com/products/light-lux-sensor-1>
Datasheet: https://support.pluxbiosignals.com/wp-content/uploads/2021/11/Light_LUX_Datasheet.pdf

Load Cell:

Website: <https://www.pluxbiosignals.com/products/load-cell>
Datasheet: https://support.pluxbiosignals.com/wp-content/uploads/2021/11/Load_Cell_Datasheet.pdf

Pushbutton (BTN):

Website: <https://www.pluxbiosignals.com/products/handheld-switch>
Datasheet: <https://support.pluxbiosignals.com/wp-content/uploads/2021/10/biosignalsplux-Handheld-Switch-Trigger-Button-Datasheet.pdf>

Respiration (RIP):

- Website: <https://www.pluxbiosignals.com/products/inductive-respiration-rip-sensor>
- Datasheet: https://support.pluxbiosignals.com/wp-content/uploads/2021/11/Respiration_RIP_Datasheet.pdf

Respiration (PZT – Piezoelectric):

- Website: <https://www.pluxbiosignals.com/products/respiration-pzt>
- Datasheet: https://support.pluxbiosignals.com/wp-content/uploads/2021/11/Respiration_PZT_Datasheet.pdf

SpO2:

- Website: <https://www.pluxbiosignals.com/products/blood-oxygen-saturation-spo2-sensor>*
- <https://www.pluxbiosignals.com/products/blood-oxygen-saturation-spo2-versatile-sensor>**
- Datasheet: https://support.pluxbiosignals.com/wp-content/uploads/2022/02/SpO2_Datasheet.pdf*
- <https://support.pluxbiosignals.com/wp-content/uploads/2021/10/biosignalsplux-SpO2-Versatile-Datasheet.pdf>**

* [with finger clip] ** [without finger clip]

SpO2 Reader (OSL):

- Datasheet: [https://www.downloads.plux.info/OpenSignals/misc/user-manual/SpO2_Reader_\(OSL\)_Datasheet.pdf](https://www.downloads.plux.info/OpenSignals/misc/user-manual/SpO2_Reader_(OSL)_Datasheet.pdf)

Temperature (TMP):

- Website: <https://www.pluxbiosignals.com/products/temperature-tmp>
- Datasheet: https://support.pluxbiosignals.com/wp-content/uploads/2021/11/Temperature_TMP_Datasheet.pdf

Vaginal EMG:

- Datasheet: <https://support.pluxbiosignals.com/wp-content/uploads/2021/10/biosignalsplux-Electromyography-EMG-Datasheet.pdf>

5.14.2 BITalino

The following *BITalino* sensors are compatible with *OpenSignals (r)evolution*. The available channel types (detailed configuration instructions presented in section 2.2.4.6 *Channel Configuration*) for each sensor can be found in, followed by links to each sensor's datasheet and website can be found below this table.

Table 15: Channel types of compatible *BITalino* sensors.

TYPE	SENSOR
ACC	Accelerometer (ACC)
CO2	CO2 sensor
CUSTOM	Customizable channel configuration
DIA	Blood pressure reader (BPR) – diastolic sensor values
ECG	Electrocardiography (ECG)
EDA	Electrodermal Activity (EDA)
EEG	Electroencephalography (EEG)
EKG	Electrogastrography (EKG)
EMG	Electromyography (EMG)
EOG	Electrooculography (EOG)
GLUC	Glucometer Data
HR	SpO2 reader (OSL) – heart rate sensor values
LUX	Light sensor (LUX)
OXI	SpO2 sensor (with or without clip)
PULSEOXI	SpO2 reader (OSL) – oxygen saturation sensor values
RAW	Raw digital signal; compatible with every sensor
RESP	Respiration sensor (RIP & PZT)
SYS	Blood pressure reader (BPR) – systolic sensor values
TEMP	Temperature sensor (TMP)
TEMPw	High resolution temperature sensor
TVOC	Total Volatile Organic Compounds

Accelerometer (ACC):

Website: <https://www.pluxbiosignals.com/products/accelerometer-acc-sensor>

Datasheet: <https://support.pluxbiosignals.com/wp-content/uploads/2021/11/revolution-acc-sensor-datasheet-revb.pdf>

Blood Pressure Reader (BPR):

Datasheet: [https://www.downloads.plux.info/OpenSignals/misc/user-manual/Blood_Pressure_Reader_\(BPR\)_Datasheet.pdf](https://www.downloads.plux.info/OpenSignals/misc/user-manual/Blood_Pressure_Reader_(BPR)_Datasheet.pdf)

Electrocardiography (ECG):

Website: <https://www.pluxbiosignals.com/products/electrocardiography-ecg-sensor>
Datasheet: <https://support.pluxbiosignals.com/wp-content/uploads/2021/11/revolution-ecg-sensor-datasheet-revb-1.pdf>

Electrodermal Activity (EDA):

Website: <https://www.pluxbiosignals.com/products/electrodermal-activity-eda-sensor>
Datasheet: <https://support.pluxbiosignals.com/wp-content/uploads/2021/11/eda-sensor-datasheet-revb.pdf>

Electroencephalography (EEG):

Website: <https://www.pluxbiosignals.com/products/electroencephalography-eeeg-sensor>
Datasheet: <https://support.pluxbiosignals.com/wp-content/uploads/2021/11/revolution-eeeg-sensor-datasheet-revb.pdf>

Electrogastrography (EGG):

Website: <https://www.pluxbiosignals.com/products/electrogastrography-egg-sensor>
Datasheet: <https://support.pluxbiosignals.com/wp-content/uploads/2021/11/egg-sensor-datasheet-revb.pdf>

Electromyography (EMG):

Website: <https://www.pluxbiosignals.com/products/electromyography-emg-sensor>
Datasheet: <https://support.pluxbiosignals.com/wp-content/uploads/2021/11/revolution-emg-sensor-datasheet-1.pdf>

Electrooculography (EOG):

Website: <https://www.pluxbiosignals.com/products/electrooculography-eog-sensor>
Datasheet: <https://support.pluxbiosignals.com/wp-content/uploads/2021/11/eog-sensor-datasheet-rev.pdf>

Flexi Force:

Website: <https://www.pluxbiosignals.com/products/force-fsr>
Datasheet: https://support.pluxbiosignals.com/wp-content/uploads/2021/11/Pressure_FSR_Datasheet.pdf

Glucose Meter Reader (GMR):

Datasheet: [https://www.downloads.plux.info/OpenSignals/misc/user-manual/Glucose_Meter_Reader_\(GMR\)_Datasheet.pdf](https://www.downloads.plux.info/OpenSignals/misc/user-manual/Glucose_Meter_Reader_(GMR)_Datasheet.pdf)

High Definition Temperature (NTC):

Datasheet: <https://www.downloads.plux.info/OpenSignals/misc/user-manual/ntc-sensor-datasheet-revb.pdf>

Light (LUX):

Website: <https://www.pluxbiosignals.com/products/light-lux-sensor>

Datasheet: <https://support.pluxbiosignals.com/wp-content/uploads/2021/11/revolution-lux-sensor-datasheet.pdf>

Pushbutton (BTN):

Website: <https://www.pluxbiosignals.com/products/pushbutton-btn>

Datasheet: <https://support.pluxbiosignals.com/wp-content/uploads/2021/11/revolution-btn-sensor-datasheet.pdf>

Respiration (PZT – Piezoelectric):

Website: <https://www.pluxbiosignals.com/products/respiration-pzt>

Datasheet: https://support.pluxbiosignals.com/wp-content/uploads/2021/11/Respiration_PZT_Datasheet.pdf

SpO2 Reader (OSL):

Datasheet: [https://www.downloads.plux.info/OpenSignals/misc/user-manual/SpO2_Reader_\(OSL\)_Datasheet.pdf](https://www.downloads.plux.info/OpenSignals/misc/user-manual/SpO2_Reader_(OSL)_Datasheet.pdf)

Temperature (TMP):

Website: <https://www.pluxbiosignals.com/products/temperature-tmp-sensor>

Datasheet: <https://support.pluxbiosignals.com/wp-content/uploads/2021/11/revolution-tmp-sensor-datasheet.pdf>

5.15 3rd-party Sensors

OpenSignals (r)evolution and PLUX devices allow the usage and acquisition of data from 3rd party sensors, using the respective accessories (see below). These accessories allow you to connect analog or digital third-party sensors to your PLUX devices which comply with the **specified and compatible electric values of the PLUX device**.

Please read the following sections and review the sensor's and/or device's datasheet **before** connecting any device to your PLUX device or sensor.

5.15.1 I/O Breakout Cable

WARRANTY WARNING

Usage of this breakout cable input interface **VOIDS THE ELECTRICAL WARRANTY** of your device and sensors.

DISCLAIMER

BEWARE when using this accessory to preserve electrical isolation of the user.

We expressly disclaim any liability whatsoever for any direct, indirect, consequential, incidental or special damages, including, without limitation, lost revenues, lost profits, losses resulting from business interruption or loss of data, regardless of the form of action or legal theory under which the liability may be asserted, even if advised of the possibility of such damages.

This accessory allows the connection, within the specified electric values, of a wide range of analog or digital third-party sensors to the *biosignalsplux* hub. This allows, for instance, the acquisition and visualization in real time on *OpenSignals (r)evolution* of a third-party sensor, using the *biosignalsplux* hub.



SPECIFICATIONS

Maximum Output Current: 200mA (with no other sensors connected to the hub)

Input Voltage Range: 0 – 3V

Output Voltage Range: 0 – 3V or $\pm 1.5V$ (if reference is GND)

DOCUMENTATION

Datasheets and user manuals for your device can be found on the following website:

<https://www.pluxbiosignals.com/products/breakout-cable>

Check the table below and the breakout cable datasheet (see annexes) for detailed information about the pinout configuration for the analog ports (standard sensor channels) and digital ports (8-channel *biosignalsplux* devices only) **BEFORE CONNECTING YOUR SENSOR TO YOUR DEVICE** to avoid damaging your PLUX device.

Table 16: Pinout configuration for analog & digital ports.

ANALOG PORT		DIGITAL PORT	
V-	Accelerometer (ACC)	V-	0V
GND	CO2 sensor	GND	Digital output
V+	Customizable channel configuration	V+	3V
IN	Analog	IN	Digital input

5.15.2 Sensor Breakout PCB

WARRANTY WARNING

Usage of this breakout cable input interface **VOIDS THE ELECTRICAL WARRANTY** of your device and sensors.

DISCLAIMER

BEWARE when using this accessory as to preserve electrical isolation of the user.

We expressly disclaim any liability whatsoever for any direct, indirect, consequential, incidental or special damages, including, without limitation, lost revenues, lost profits, losses resulting from business interruption or loss of data, regardless of the form of action or legal theory under which the liability may be asserted, even if advised of the possibility of such damages.

This accessory allows easy access, within the specified electric values, to the pins on *biosignalsplux* sensors and actuators for interfacing them with third-party systems.

Check the sensor datasheets **before** using this accessory to avoid any damage of your system or your sensor(s).



DOCUMENTATION & SPECIFICATIONS

Please read the datasheet of your sensors to confirm the electrical values and compatibility of your sensor.

Table 17: Pinout configuration for analog & digital ports.

ANALOG PORT	
V-	GND
GND	1.5V (to sensor)
V+	3V (to sensor)
IN	Sensor output

DIGITAL PORT	
V-	GND
GND	Sensor output
V+	3V (to sensor)
IN	1.5V (to sensor)

6 Software & APIs

6.1 OpenSignals (r)evolution

OpenSignals (r)evolution is our easy-to-use, versatile, and scalable software for real-time biosignals visualization, capable of direct interaction with all PLUX devices.

Core functionality includes sensor data acquisition from multiple channels and devices, data visualization and recording, as well as loading of pre-recorded signals. *OpenSignals (r)evolution* is also a Python-powered web-based software framework, targeted at rapid application development; a bare bone code base is available on [GitHub](#)

You can download the newest version of *OpenSignals (r)evolution* here:

<https://www.pluxbiosignals.com/pages/opensignals>

6.2 OpenSignals Mobile

OpenSignals Mobile is a slimmed down version of *OpenSignals (r)evolution* specifically designed to run on **Android**® mobile phones or tablets (iOS coming soon), while preserving the ease-of-use and performance for real-time sensor data visualization and recording.

OpenSignals Mobile is currently in development. However, if you are interested in being one of the first few users to get your hands on a beta version of this app, visit our *OpenSignals* website and fill in the form to subscribe for the early access beta version.

6.3 APIs

It is possible to connect the devices to third party applications including your own custom software. The needed toolkits to start your own biosignal software are delivered by PLUX for free. All the functionalities and features of our high-end research devices can be accessed by developers to integrate PLUX systems in custom and third-party applications.

biosignalsplux APIs:

<https://biosignalsplux.com/products/software/apis.html>

BITalino APIs:

<https://bitalino.com/en/development/apis>

7 Troubleshooting

7.1 Windows Bluetooth Stack Troubleshooting

Most internal Bluetooth adapters are compatible with PLUX devices. However, if you're experiencing a great number of connection loss events, reconnection problems, missing samples or similar problems, your internal Bluetooth module might not have enough performance to deal with the data throughput of the device you are using. In this case, reducing the sampling frequency or the sample resolution might solve this issue, if a change of these parameters is possible.

For this reason, PLUX has identified Bluetooth dongles that have been tested and are known to work with our devices. We recommend that you use one of these PLUX-tested Bluetooth dongles, which are either already shipped with your device (e.g. in the case of *biosignalsplux* kits) or can be purchased separately from our store. These dongles are compatible to work with data rates up to 24 Mbps.

The solution is to exchange the internal Bluetooth Adapter to an **external Bluetooth USB dongle that uses the Windows native Bluetooth stack**. For this, you'll need to install the Bluetooth dongle properly, as described in the following steps, which are explained in more detail throughout the next pages for the supported operating systems:

- Step 1:** Removing already paired devices from your operating system
- Step 2:** Installing PLUX-proven Bluetooth dongle
- Step 3:** Changing Bluetooth Stack to Microsoft's native Bluetooth stack

NOTE

If you're already using PLUX devices with the old Bluetooth stack, please remove your device(s) from your computer **before** changing the Bluetooth drivers/changing to Microsoft Bluetooth Stack.

NOTE

We recommend going through all the following steps to set up your Bluetooth dongle correctly, even if Windows may automatically install and set up your dongle.

NOTE

Installing the Bluetooth dongle and changing the Bluetooth stack is only required for Windows computers only. Computers with a Mac OS X operating system do not need to go through these steps and use the internal Bluetooth adapter instead.

7.1.1 Windows 7 & Windows 8

Step 1: Removing already paired devices from your operating system

To remove Bluetooth devices, it is needed to access the *Device Manager* on your computer and view the list of connected devices. The device list can be found following these steps:

Start Button > **Control Panel** > **Hardware and Sound** > **View devices and printers**

In the *Device and Printers* window, PLUX devices can be found in the *Unspecified* section. Right-click on the device you want to remove and select *Remove device* (see Figure 228)

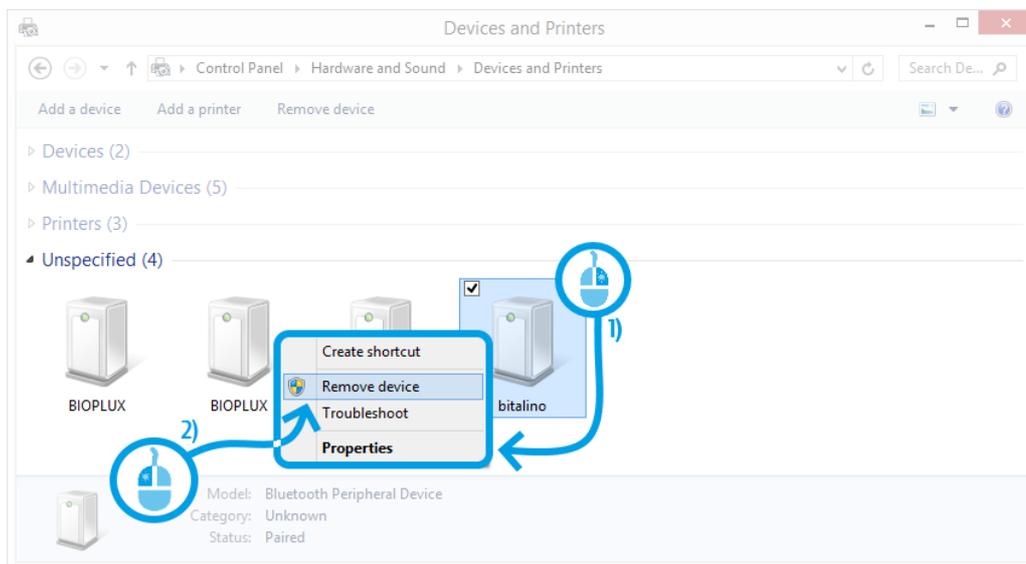


Figure 228: Windows 7 "Devices and Printers" window.

If the device is not listed under *Unspecified* anymore, it has been successfully removed from your system.

Step 2: Installing PLUX-proven Bluetooth dongle

To install the PLUX-proven Bluetooth dongle, plug it in and install all the necessary drivers for the Bluetooth dongle to work (an Internet connection is advisable). Windows should display the window below.

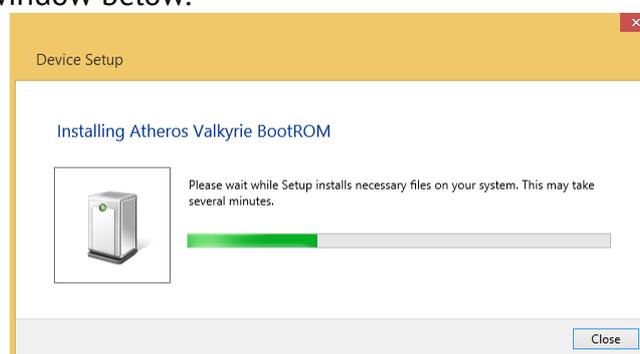


Figure 229: Wait for Windows to install your Bluetooth dongle.

Make sure you wait until your Bluetooth Dongle is installed. Windows will issue a notification and you'll be ready to configure the correct Bluetooth stack following the next step.

Step 3: Changing Bluetooth Stack to Microsoft's native Bluetooth stack

To configure the Bluetooth stack of the Bluetooth dongle, you have to connect your Bluetooth dongle to your computer and access the *Device Manager* on your computer. The device manager can be found following these steps:

Start Button > **Control Panel** > **System and Maintenance** > **Device Manager**

In the device manager, the Bluetooth dongle should be listed as *Generic Bluetooth Adapter* (see Figure 230). If your machine already has an internal Bluetooth Adapter and you have just plugged in the PLUX-proven Bluetooth Dongle, keep in mind that the last one will be automatically disabled by Windows. This is why one of your Bluetooth adapters under Bluetooth in the device manager shows a yellow warning icon (⚠) stating: *Code 10 - This device cannot start.*

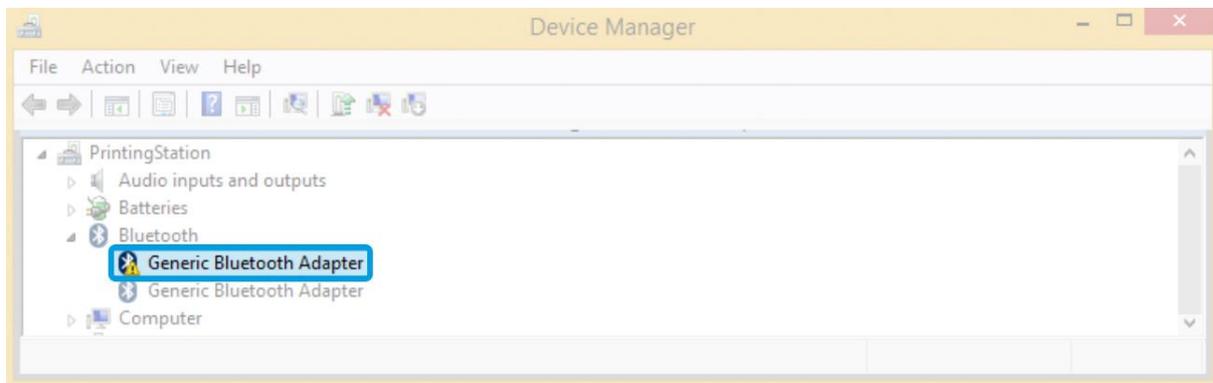


Figure 230: Bluetooth dongle listed in the device manager.

If your Bluetooth dongle is the one disabled, you can manually disable your internal Bluetooth adapter by **right-clicking** on it in the device manager and selecting **Disable**. This action will enable your Bluetooth dongle by default.

If, after this action, your Bluetooth Dongle continues to display a yellow warning icon (⚠), it means that it is not installed correctly and you should try to update the drivers by right-clicking on it and selecting *Update Driver Software...*, reboot your computer (in order to the installed components be fully functional) or fix the problem displayed under *Device Status* in the device properties (right-click the Bluetooth device and select Properties; see screenshot below).

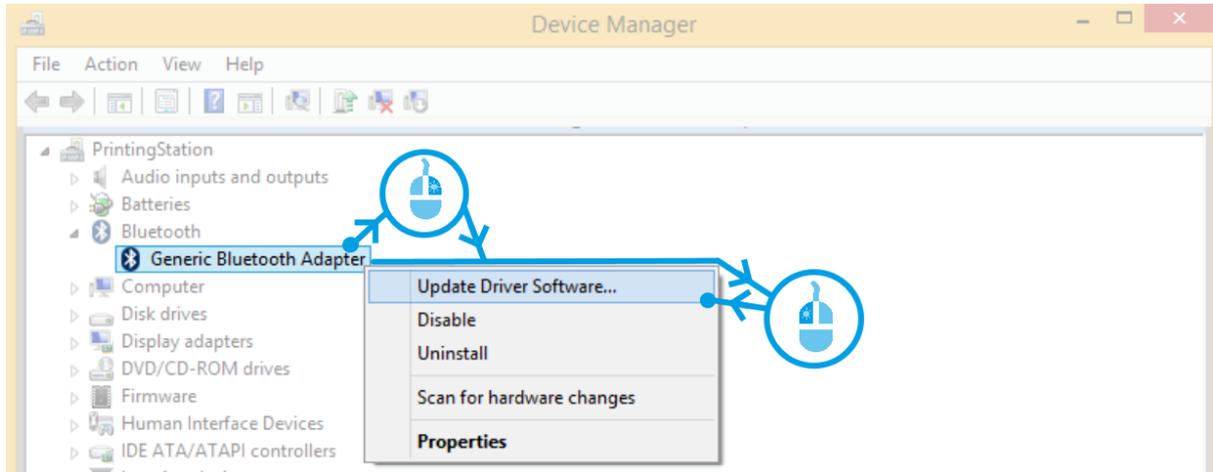


Figure 231: First step to update the driver of the Bluetooth dongle.

Select *Browse my computer for driver software* from the new window.

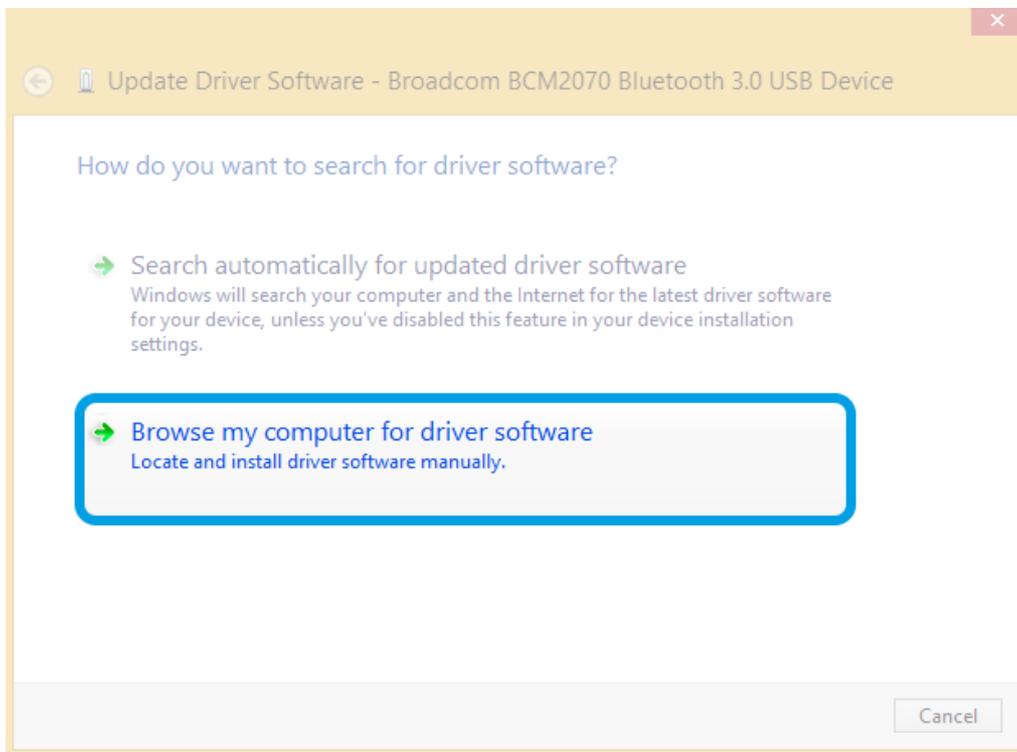


Figure 232: Browse for driver software.

Select *Let me pick from a list of device drivers on my computer.*

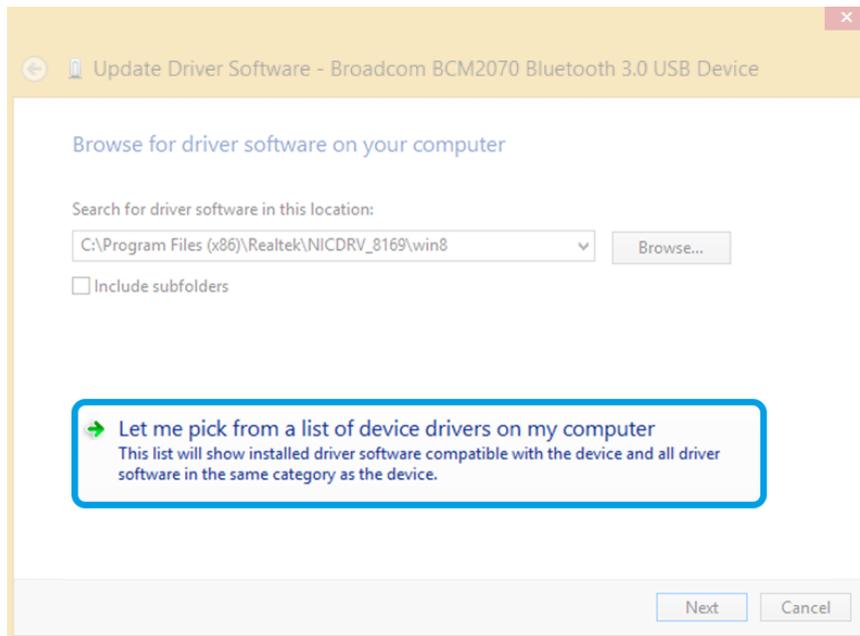


Figure 233: Pick driver from list of device drivers on the computer.

Select *Generic Bluetooth Adapter* and click on *Next* to install the Microsoft Bluetooth Stack. Of several models of *Generic Bluetooth Adapter* are available you can select any of them. It will not make any difference.

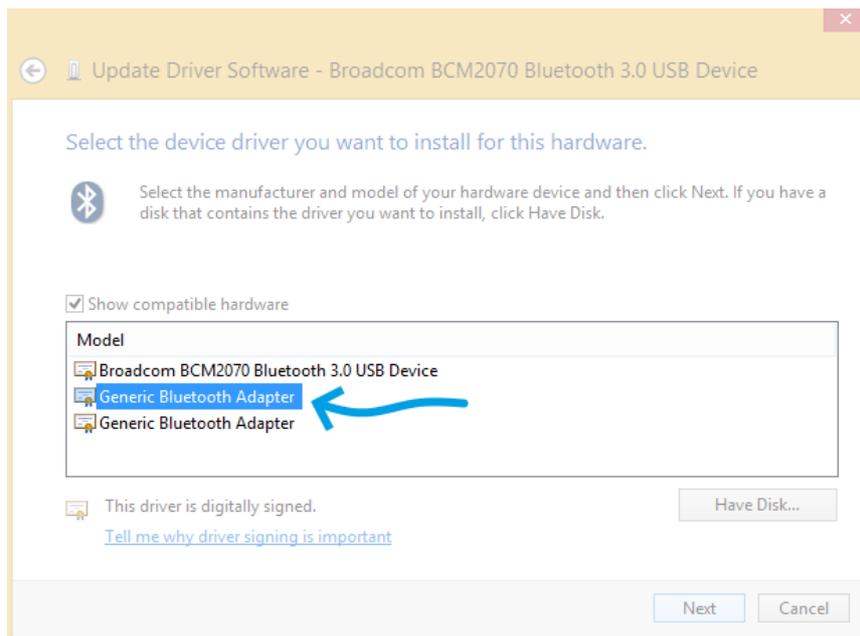


Figure 234: Select *Generic Bluetooth Adapter*

Your device should now be configured properly and have no warning sign or error message in the device manager. Rebuild the connection with your device (see instructions in 1.4.1 Windows 7 or 1.4.2 Windows 8) and test your device by recording signals with the *OpenSignals (r)evolution* software. If, after all these steps, you keep experiencing the same issues, access our support page (<https://support.pluxbiosignals.com/>)

7.1.2 Windows 10

Step 1: Removing already paired devices from your operating system

To remove Bluetooth devices, it is needed to access the Bluetooth configuration of your operating system. Click on the start button at the left lower corner in of your desktop to open the start menu. Click on the gear symbol to open the settings panel of your operating system as can be seen below.

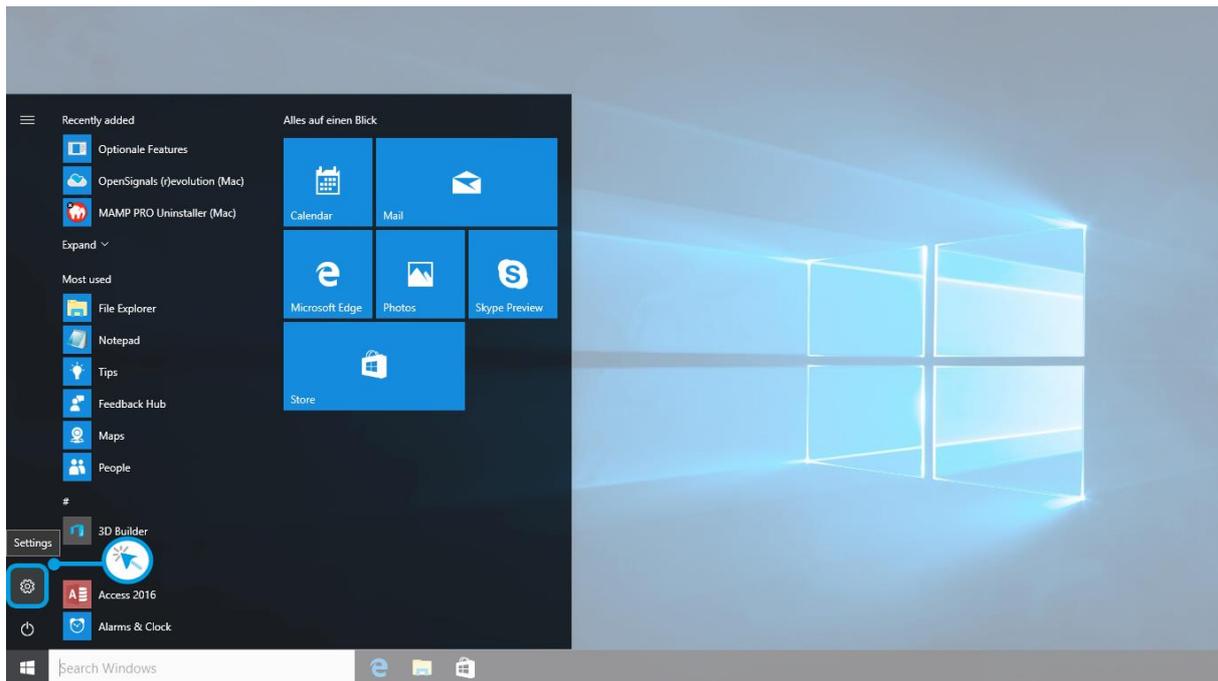


Figure 235: Access to the settings panel in Windows 10.

In the settings panel, click on the *Devices Bluetooth, printers, mouse* field to open the device configurations of your system.

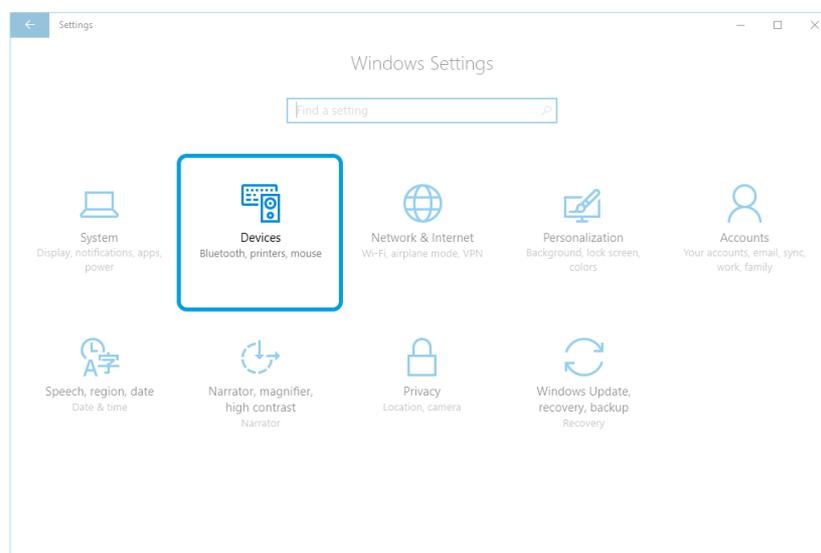


Figure 236: Windows 10 settings panel.

Select *Bluetooth* from the list which is displayed on the left side of your settings window. Note, that in some versions the Bluetooth option might not be available and that Bluetooth devices might be listed under *Other devices*.

Click on your *biosignalsplux* and click on *Remove device* to remove this Bluetooth connection from your computer and confirm the *Are you sure you want to remove this device?* message that will be shown on your screen. Repeat this step with every *biosignalsplux* if you're using several devices.

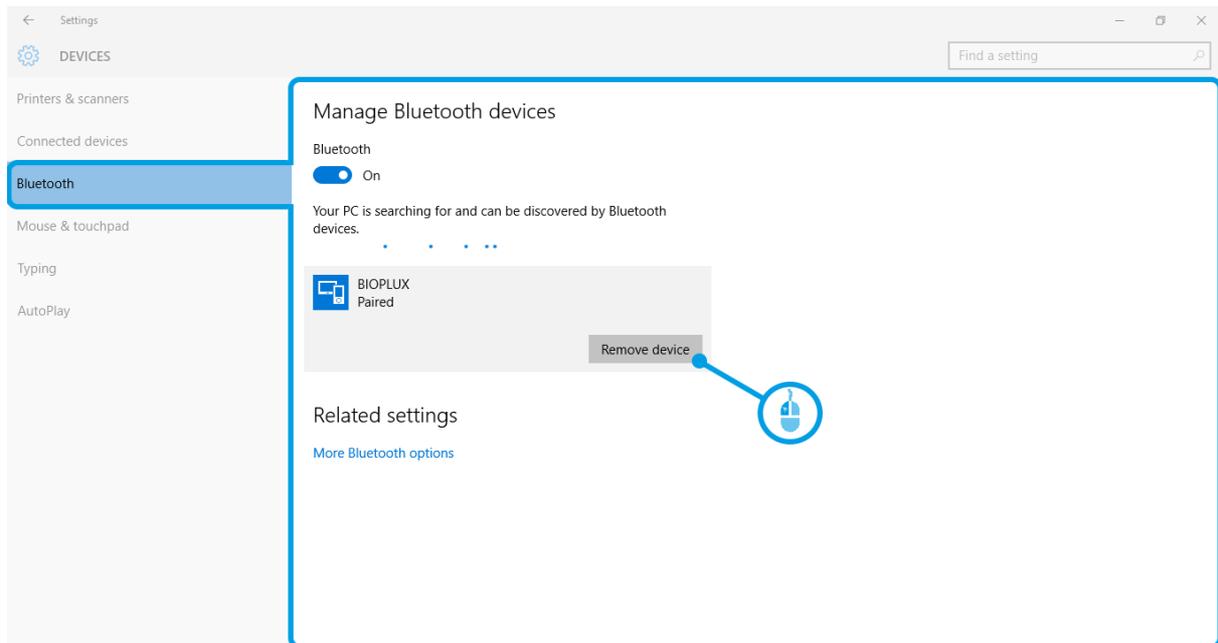


Figure 237: Windows 10 Bluetooth devices

If, after this step, your device is not listed in the list of Bluetooth devices anymore, the device has been successfully removed.

Step 2: Installing PLUX-proven Bluetooth dongle

To install the PLUX-proven Bluetooth dongle, plug it into your computer. Windows will install the Bluetooth dongle and display the window below.

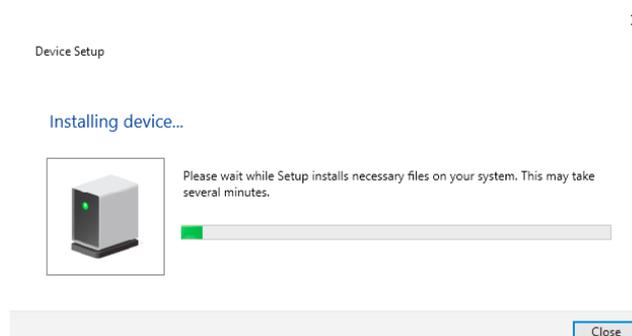


Figure 238: Wait for Windows to install your Bluetooth dongle.

Make sure you wait until your Bluetooth Dongle is installed. Windows will issue a notification and you'll be ready to configure the correct Bluetooth stack following the next step. Note that this step might require a restart of your system. Make sure to close all your open work and programs properly before restarting the system to prevent data losses or other issues.

Step 3: Changing Bluetooth Stack to Microsoft's native Bluetooth stack

To configure the Bluetooth stack of the Bluetooth dongle, you have to connect your Bluetooth dongle to your computer and access the *Device Manager* on your computer. Enter *Device Manager* into the search field of your taskbar and click on the listed device manager.

In the device manager, the Bluetooth dongle should be listed as *Generic Bluetooth Adapter* (see Figure 239). If your machine already has an internal Bluetooth Adapter and you have just plugged in the PLUX-proven Bluetooth Dongle, keep in mind that the last one will be automatically disabled by Windows. This is why one of your Bluetooth adapters under Bluetooth in the device manager shows a yellow warning icon (⚠) stating: *Code 10 - This device cannot start*.

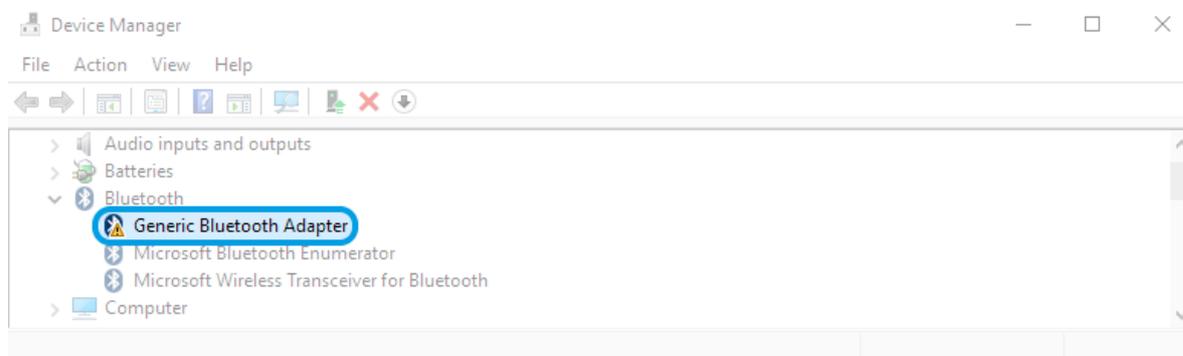


Figure 239: Bluetooth dongle listed in the device manager.

If, after this action, your Bluetooth Dongle continues to display a yellow warning icon (⚠), it means that it is not installed correctly and you should try to update the drivers by right-clicking on it and selecting *Update Driver Software...* or fix the problem displayed under *Device Status* in the device properties (right-click the Bluetooth device and select Properties; see screenshot below).

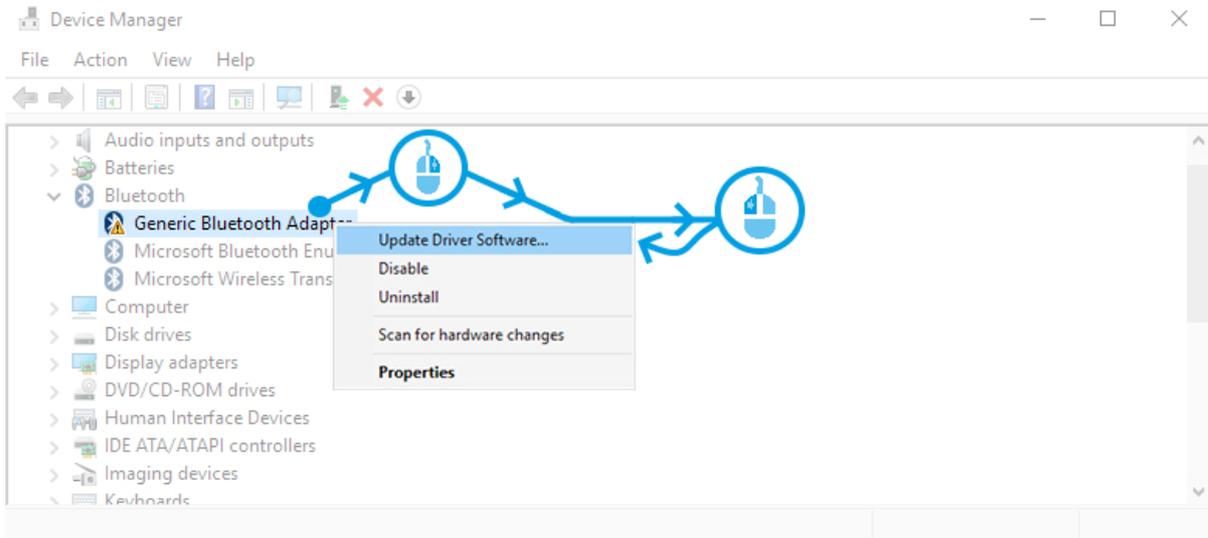


Figure 240: First step to update the driver of the Bluetooth dongle.

Select *Browse my computer for driver software* from the new window.

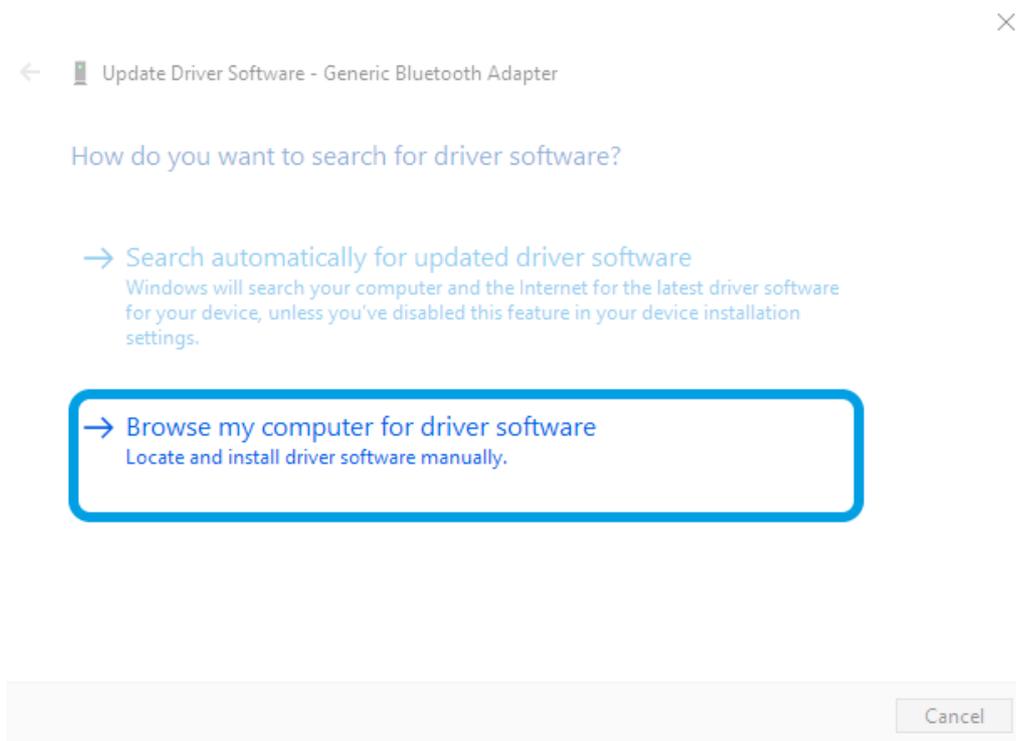


Figure 241: Browse for driver software.

Select *Let me pick from a list of device drivers on my computer*.

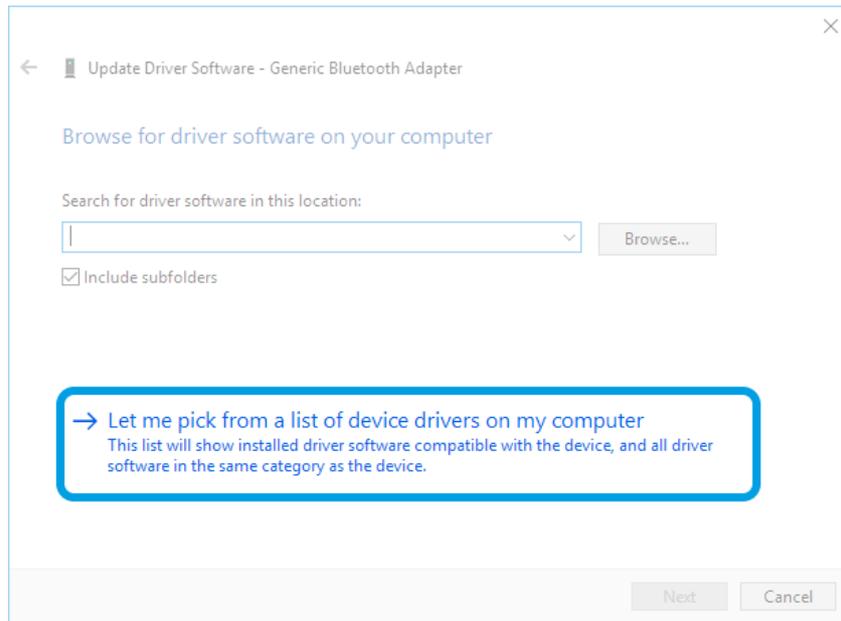


Figure 242: Pick driver from list of device drivers on the computer.

Select *Generic Bluetooth Adapter* and click on *Next* to install the Microsoft Bluetooth Stack. Several models of *Generic Bluetooth Adapter* are available and you can select any of them. It will not make any difference.

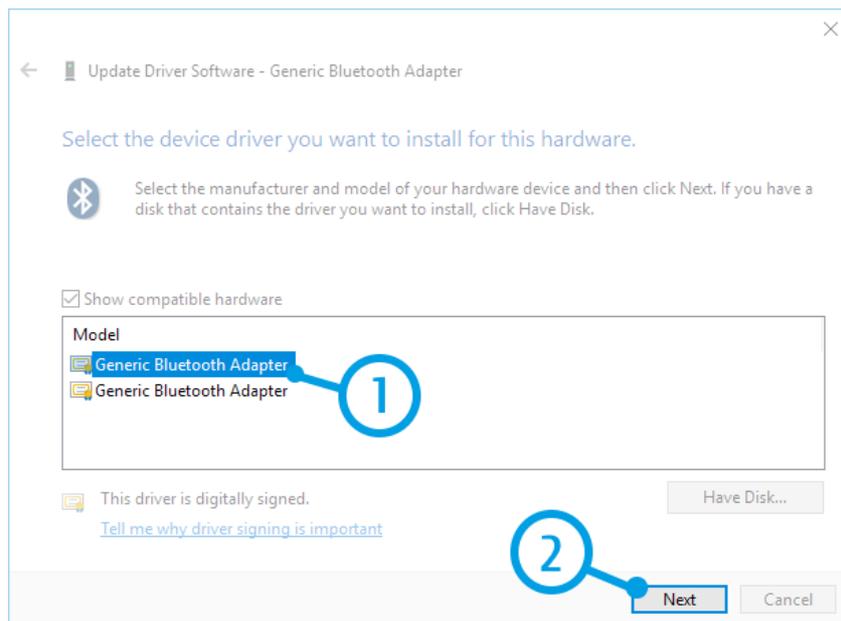


Figure 243: Select Generic Bluetooth Adapter

Your device should now be configured properly and have no warning sign or error message in the device manager. Rebuild the connection with your *biosignalsplux* device (see 1.4 *Connecting Bluetooth Devices*) and test your device by recording signals with the *OpenSignals (r)evolution* software. If, after all these steps, you keep experiencing the same issues, access our support page (<https://support.pluxbiosignals.com/>)

7.2 Device Firmware & Update

biosignalsplux firmware updates are not done automatically. We recommend to frequently check if newer firmware versions are available and to upgrade or update the firmware on your device when newer versions are available to ensure full functionality and compatibility with all *biosignalsplux* sensors and accessories. Updating the *biosignalsplux* firmware lets you also take advantage of all the newest recording resolutions and offline scheduling features.

Newer versions of *OpenSignals (r)evolution* will also issue a notification if the firmware of your device is too old and not supported by your current version of *OpenSignals*.

In both cases, check the following website to see if newer firmware versions are available, and download and extract all the required tools to update or upgrade your firmware (or make the update through *OpenSignals (r)evolution* | see 2.2.4.11 Device Info):

https://www.downloads.plux.info/OpenSignals/misc/user-manual/biosignalsplux_FW_3.7_Installer.zip

NOTE

biosignalsplux devices can be used on Windows, Linux and Mac OSX operating systems. However, at the time being, the firmware update software does only run on Windows operating system.

For Mac OS X and Linux users, the firmware update software can be used in virtual machines running a compatible Windows version (Windows 7 or newer) or directly through the Device Info section on *OpenSignals (r)evolution* (see 2.2.4.11 Device Info).

7.2.1 Checking Firmware Version (external tool)

Before updating the firmware of your *biosignalsplux* device, it is recommended to check the current device's firmware version to see whether a firmware update is required or not.

To do so, make sure that your device is charged and turned on (LED must blink green). If you haven't connected your device to your computer via Bluetooth before, connect your device following the instructions 1.4 *Connecting Bluetooth Devices* before following the next steps.

Open the *Updater.exe* you've just downloaded and extracted. A new window will appear. Click on the *Find Devices* button and wait for your MAC-address to appear in the *Device Mac Address Field*.

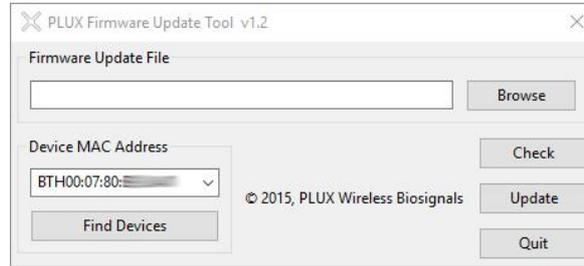


Figure 244: PLUX firmware update tool has found one biosignalsplx.

Check if the MAC-address that is listed in the dialog matches the MAC-address of the *biosignalsplx* device you want to update or upgrade. The MAC-address of the device is placed on its back. Click on the *Check* button.

The firmware update software will check the current version of the *biosignalsplx* and display it in the dialog field.

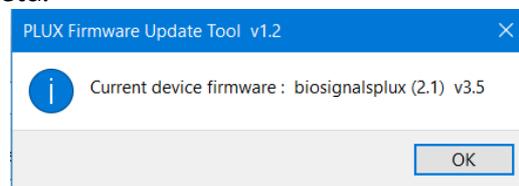


Figure 245: Dialog which contains the firmware version of the device under analysis.

7.2.2 Updating or Upgrading Firmware

If your device is running on an older firmware⁵ it is recommended to update or upgrade the firmware for compatibility reasons with new features and to ensure that your device works properly with the newest version of the *OpenSignals (r)evolution* software.

To do so, make sure that your device is charged and turned on (LED must blink green). If you haven't connected your device to your computer via Bluetooth before, connect your device following the instructions [1.4 Connecting Bluetooth Devices](#) before following the next steps.

Open the *Updater.exe* you've downloaded and extracted. A new window will appear. Click on the **Find Devices** button and wait for your MAC-address to appear in the dialog bellows. Check if the MAC-address that is listed in the dialog matches the MAC-address of the *biosignalsplx* device you want to update or upgrade. The MAC-address of the device is placed on its back.

Click on **Browse** and browse the folder where the firmware .ENC file from the downloaded package is placed and select the .ENC file. The firmware updater should have a similar appearance to the one presented on Figure 246.

In order to complete the update procedure, it will only be necessary to click the *Update* button.

⁵ or if you need to upload another firmware to run with specific sensors (e.g. beta versions of new sensors)

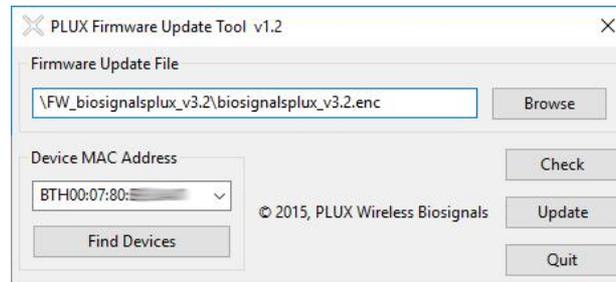


Figure 246: Firmware update tool is ready to update the firmware.

biosignalsplux will turn itself off two times during the firmware update or upgrade. Wait for the *biosignalsplux* to turn itself on again after the first time (this may take some seconds). The *biosignalsplux* firmware will now be installed on the *biosignalsplux* and the device will automatically turn itself off for the second time after the installation is finished. This time check the indications of the update software, and, if no different indications are given, turn your device back on and confirm the dialog of the update software by clicking on the *OK* button.

When the firmware update/upgrade has finished, a dialog will pop up displaying the old and the new version of the *biosignalsplux* firmware. new firmware was successfully installed if the version of the new firmware matches your selected .ENC file and if it is superior to the version of the displayed old firmware.

7.3 Launching Errors in new Linux Distributions

With the emergence of new **Linux** distributions, namely **Debian 10** and **Linux Mint 20**, a dependency required by *OpenSignals (r)evolution* software is not natively available.

Due to this circumstance, *OpenSignals (r)evolution* software is not properly launched in specific cases, being printed the following error message (when the software is executed through the *Terminal*):

```
Error: ImportError: libssl.so.1.0.0: cannot open shared object file:  
No such file or directory
```

Apparently, this error is related with the fact that newer **Linux** distributions use a different version of the **libssl** library and relevant connections to some paths used by *OpenSignals (r)evolution* were lost.

In order to recover these lost paths, it will be necessary to manually install the required **libssl.so.1.0.0** version, as demonstrated in the following steps:

1. Download of the **libssl 1.0.0** version of the missing dependency:
 - a. **Linux Mint/Ubuntu:**
http://archive.ubuntu.com/ubuntu/pool/main/o/openssl1.0/libssl1.0.0_1.0.2n-1ubuntu5_amd64.deb
 - b. **Debian:**
https://ftp.debian.org/debian/pool/main/o/openssl/libssl1.0.0_1.0.1t-1+deb8u8_amd64.deb
2. Using the *Terminal*, the user should navigate to the location where the file downloaded in **Point 1** is stored;
3. Installation of the dependency through the command:
 - a. **Linux Mint/Ubuntu**

```
sudo dpkg --install libssl1.0.0_1.0.2n-1ubuntu5_amd64.deb
```
 - b. **Debian**

```
sudo dpkg --install libssl1.0.0_1.0.1t-1+deb8u8_amd64.deb
```

In addition to the previous steps, for **Linux Mint/Ubuntu** systems another dependency (**libffi.so.6**) is necessary:

- E1. Download the **libffi.so.6** library from:
http://mirrors.kernel.org/ubuntu/pool/main/libf/libffi/libffi6_3.2.1-8_amd64.deb
- E2. Using the *Terminal*, the user should navigate to the location where the file downloaded in **Point E1** is stored;
- E3. Installation of the dependency through the command:

```
sudo apt install sudo apt install ./libffi6_3.2.1-8_amd64.deb
```

7.4 Error List

The following table will help you resolve some of the most common errors that may appear while using *OpenSignals (r)evolution*. Do not hesitate to contact us if you have any other problems or questions. TID stands for Troubleshoot ID.

TID	Error:
1	<ul style="list-style-type: none"> - No Bluetooth adapter found. Verify if an adapter is connected and turned on. - No Bluetooth adapter detected. <p>Solution:</p> <p>If you have an internal Bluetooth Adapter or have the external Bluetooth Adapter Dongle plugged in your computer, the problem might be due to:</p> <ul style="list-style-type: none"> - Driver or installation issue: Follow steps on section 7.1. Windows Bluetooth Stack Troubleshooting - Using wrong drivers: To solve, follow instructions in 7.1. Windows Bluetooth Stack Troubleshooting.
2	<p>The device could not be found. Verify that it is within range and turned on.</p> <p>Solution:</p> <p>Your Bluetooth Adapter is not being able to find the specified device. Either it's turned off, out of range or there's interference in the environment.</p>
3	<p>The communication port does not exist or it is already being used.</p> <p>Solution:</p> <p>This exception usually means that the device to which you are trying to connect is already connected somewhere else. To solve, you can either shutdown the software that is connecting to the device or try turning off and on the device itself, since this action will close the connection.</p> <p>If you keep having this issue multiple times, it may be due to a problem with the Bluetooth Adapter when it comes to closing the connection to the device. Please try the instructions in 7.1. Windows Bluetooth Stack Troubleshooting.</p> <p>If the issue is still not solved, you have to manually force the disconnection of your software with the device after each interaction. To do so, follow the instruction in related to the removal of Paired Devices (7.1. Windows Bluetooth Stack Troubleshooting).</p>

TID	Error:
4	- A socket operation was attempted to an unreachable network. - A socket operation encountered a dead end
	Solution:
	This exception will present itself in your Windows language and when you are using a BITalino. The problem (and solution) is similar to TID 1.
TID	Error:
5	A connection attempt failed because the connected party did not respond after a period of time, or established connection failed because connected host has failed to respond.
	Solution:
	This exception will be presented to you in your native Windows language and if you're using a BITalino. The problem (and solution) is similar to TID 2.
TID	Error:
6	Only one usage of each socket address is normally permitted.
	Solution:
	This exception will be presented to you in your native Windows language and if you're using a BITalino. The problem (and solution) is similar to TID 3.

8 Safety Information

Please read the following safety instructions **before** using your devices system with *OpenSignals (r)evolution* to prevent any damages or issues with the user, test persons and/or devices being used in your acquisition setup and environments. Note, that violations of these instructions can lead to inferior signal quality and/or damages to the system and user.

All PLUX products and PLUX related *OpenSignals (r)evolution* products are hereinafter referred to as “device” and/or “devices”.

- ! The user should always keep the device and its accessories dry.
- ! The user must turn off the device and contact Technical Support if the device (system and/or accessories) reach uncomfortable temperatures.
- ! The user should not use the device in noisy environments (environments with microwaves and other similar equipment). Doing so will lead to noise increase in the acquired signals and Bluetooth connectivity issues.
- ! The user must not use the device near the fire or in potentially explosive atmospheres, such as atmospheres with flammable gas.
- ! The user should only use the detection surfaces or other approved accessories purchased from PLUX or by a PLUX agent.
- ! The user should inspect the sensors on a regular basis to ensure that they remain in good working order.
- ! The user should stop using the device if experience any kind of discomfort or skin irritation.
- ! The user should not use the device continuously for periods of time above 60 minutes. Do not use the system on persons with allergies to silver.
- ! The user should dispose detection surfaces after using the device. Detection surfaces are single-user and disposable. Reusable electrodes should be reused by the same user. Do not use reusable electrodes on several users.
- ! The user must not place the device in the microwave.
- ! The user must not insert objects into the holes of the device.

- ! The user should not open the device or its accessories. The repair of the same should be only done by properly authorized PLUX personnel.
- ! The user should make sure the cables do not obstruct the passage of people.
- ! The user should use the sensor cables with extreme caution to avoid risk of strangulation.
- ! The user should keep a safe distance between the device and other devices to ensure their proper functioning.
- ! The user should only send the device to repair to qualified PLUX personnel.
- ! The user should not immerse the sensors or the *biosignalsplux* device, nor clean with liquid or abrasives.
- ! The user should handle the device with caution and not expose the device or accessories to high accelerations and vibrations.
- ! The devices should not be used in patients with implanted electronic devices of any kind, including pace-makers, electronic infusion pumps, stimulators, defibrillators or similar.
- ! Do not apply electrodes over damaged or irritated skin.
- ! Do not use your device while charging its internal battery.

9 Regulatory & Legal Information

9.1 Disclaimer

All mentioned *OpenSignals (r)evolution*, *biosignalsplux* and *BITalino* products in this manual are intended for use in life science education and research applications only; they are not medical devices, nor medical software solutions, nor are they intended for medical diagnosis, cure, mitigation, treatment or prevention of disease and is provided to you “as is”.

We expressly disclaim any liability whatsoever for any direct, indirect, consequential, incidental or special damages, including, without limitation, lost revenues, lost profits, losses resulting from business interruption or loss of data, regardless of the form of action or legal theory under which the liability may be asserted, even if advised of the possibility of such damages.

9.2 Certification

OpenSignals (r)evolution and any PLUX device connected to this software do not have a medical device certification and are, therefore, not a medical device.

PLUX research products are intended for use in life science education and research applications with humans and not intended for diagnostics, cure, mitigation, treatment or prevention of disease.

9.3 Contact & Support

Contact us if you're experiencing any problems that cannot be solved with the information given in the *biosignalsplux* or *OpenSignals (r)evolution* manual. We'll get back to you as soon as possible to find the best solution for your problem.

Please send us an e-mail with precise information about the error occurrence, device configuration, and, if possible, screenshots of the problem.

Additional support and contact info can be found at <https://support.pluxbiosignals.com/>.