



---

## **Application Note 004**

OpenGo Synchronization

## Contact

Moticon ReGo AG  
Address: Machtlfinger Str. 21, 81379 Munich, Germany  
EMail: support@moticon.com  
Phone: +49 89 2000 301 50

## Legal Note and Disclaimer

Copyright (c) 2023, Moticon ReGo AG  
All rights reserved.

Any redistribution in source or binary forms, with or without modification, is not permitted.

THIS APPLICATION NOTE AND ANY REFERENCED SOFTWARE IS PROVIDED BY MOTICON REGO AG "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, NONINFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL MOTICON REGO AG OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## Changelog

Version	Date	Changes
2.2	11.09.2022	Add link to OpenGo Sync Box
2.1	05.09.2022	Synchronization using TTL signal
2.0	01.09.2021	Changed product name from SCIENCE to OpenGo
1.0	18.02.2020	Initial version

# Contents

<b>1</b>	<b>General Notes</b>	<b>4</b>
<b>2</b>	<b>Synchronization Overview</b>	<b>5</b>
<b>3</b>	<b>Synchronous Data Collection</b>	<b>6</b>
3.1	TTL Trigger . . . . .	6
3.2	Numeric Data (UDP Protocol) . . . . .	6
3.3	Video (Webcam) . . . . .	7
<b>4</b>	<b>Separate Data Collection</b>	<b>7</b>
4.1	Numeric Data (Text Import/Export) . . . . .	7
4.2	Video . . . . .	7
<b>5</b>	<b>Time Alignment</b>	<b>8</b>
5.1	Channels and Channel Groups . . . . .	8
5.2	Text Export Resampling . . . . .	9
5.3	Data Units . . . . .	9

# 1 General Notes

This application note describes ways of synchronizing Moticon OpenGo measurement data with other (“external”) measurement systems. Typical examples for such systems are EMG, force plates, IMUs, or video cameras.

In this context, “synchronization” means one of the following:

- To obtain data of sensor insoles and external measurement systems, all having an absolute time which, if not from a common reference clock, at least has acceptable concurrency.
- To have both sensor insole data and numeric data of external measurement systems inside the OpenGo Software, aligned in time and allowing for synchronous replay of the data (external data is displayed in the line graph widget only).
- To have both sensor insole data and video inside the OpenGo Software, aligned in time and allowing for synchronous replay.

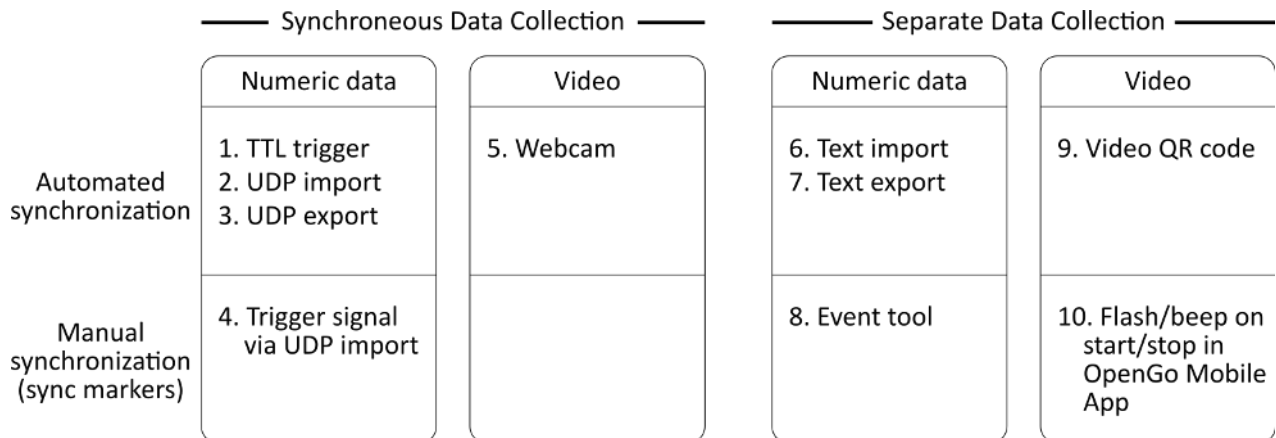
To achieve the above, the following sections describe:

- How to collect data simultaneously, such that it is automatically synchronized.
- How to trigger sync markers or signals, which help when manually aligning measurement data after measurements have been completed.
- How to manipulate timing information to actually carry out time shift adjustments with minimal effort.

The following assumes that you are familiar with the general operation of OpenGo Sensor Insoles and the OpenGo Software.

## 2 Synchronization Overview

The following figure displays the different options for synchronizing measurement data:



**Figure 1** — Data synchronization options. The numbers are referenced in Sec. 3 and Sec. 4.

Ideally, the data of different measurement systems is available as a time-synchronous data collection at the moment a measurement is stopped. With OpenGo, this is possible when carrying out live capture measurements, where data is transmitted in real time from the sensor insoles to the PC. Sec. 3 summarizes the available options in this setup.

If the different measurement systems cannot be coupled directly, it is still possible to obtain synchronous data by post-measurement time alignment. Sec. 4 summarizes automated and manual approaches. For the OpenGo Sensor Insoles, this applies to both live capture and recording.

Finally, it may be necessary to manually set (or fine-tune) the data time alignment. Sec. 5 explains how time offsets of different measurement systems can be easily set and adjusted in the OpenGo Software.

**Synchronization accuracy** Delays inherently caused by the involved IT components can lead to synchronization offsets. If this is a particular concern, tests should be carried out in the respective system environment in order to quantify potential offsets.

## 3 Synchronous Data Collection

### 3.1 TTL Trigger

The OpenGo App provides a function to generate an audio signal, which can be turned into a TTL (Transistor-Transistor Logic) trigger signal via the OpenGo Sync Box <https://moticon.com/opengo/app#sync-box>

1. This TTL signal may then be used to automatically and synchronously start other measurement systems.

Depending on how other measurement systems operate, another use case can be to just record the TTL signal, and use it as sync marker for later synchronization.

### 3.2 Numeric Data (UDP Protocol)

The OpenGo Software provides UDP import and UDP export functions during live capture measurements. This way, measurement systems able to expose their data in real time as UDP data packets, or vice versa able to include UDP data packets in their own data stream, can be coupled.

Even if the external measurement system does not support UDP, but a different real-time data interface, developers can program a UDP software bridge. Related Python code is provided at <https://account.moticon.com/doc-opengo-udp>.

2. By configuring the UDP import, each received external data packet is included in the measurement file in an “external” data channel group (see Sec. 5.1). The time of the corresponding data sample is the current PC system time. If the PC system time and the time of the mobile device controlling the sensor insoles is synchronous, then the resulting measurement file will contain synchronous sensor insole and external data.
3. By configuring the UDP export, each received sensor insole data packet is automatically re-transmitted via UDP (in addition to being included in the .go measurement file). It is the responsibility of the external measurement system to align received UDP data with the external data.

If the external measurement system cannot provide or handle UDP data, there may still be a system-wide trigger signal starting different measurement systems.

4. Using a UDP software bridge (see above), it is possible to include the trigger signal as external data channel. This signal is then included in the text export, and can be used for data synchronization.

For more information about UDP, see <https://account.moticon.com/doc-opengo-udp>.

### 3.3 Video (Webcam)

The video of webcam cameras can be included in the .go file along with the live capture data from sensor insoles:

5. The webcam is configured in the settings of the Record section of the OpenGo Software, which also provides a webcam widget. For more details, see <https://account.moticon.com/doc-opengo-record-video>

## 4 Separate Data Collection

### 4.1 Numeric Data (Text Import/Export)

The following functions can be used for post-measurement time alignment of numeric data:

6. If the data of external measurement systems is available as a text file with known absolute start time, the text import function of the OpenGo Software can be used to have this data automatically aligned with sensor insole data. For details about the required text file format: <https://account.moticon.com/doc-opengo-import>
7. Vice versa, the OpenGo Software provides a text export function, in case the time alignment based on absolute start times shall be carried out by different software. For more details, see <https://account.moticon.com/doc-opengo-available-reports#text-export>
8. The event widget in the Record section of the OpenGo Software is helpful to set markers relevant for manual time synchronization, e.g. the point in time when a different measurement system was started. Later, this event marker may either be used to crop the sensor insole data before exporting it, or to align imported external data in the line graph widget. For more details, see <https://account.moticon.com/doc-opengo-record-events>

### 4.2 Video

There is both an automatic and a manual solution for post-measurement alignment of videos:

9. The absolute time of a video can be automatically determined by the OpenGo Software, if it shows a sequence of a QR code containing the absolute time. Both the OpenGo Software and the OpenGo App provide a video sync widget (QR code), which should be filmed for at least 5 seconds. For cases where the automatic detection fails, the sync widget also contains the coded absolute time (in ms) in human readable form. For more details, see <https://account.moticon.com/doc-opengo-record-video#external-camera>
10. For manual alignment of video, the OpenGo App can flash the camera flash light and play a beep when starting and stopping a measurement. This allows for manual synchronization of the video with the point in time when the start/stop button was pressed in the Record screen of the OpenGo App.

**Note** Due to minor delays between button press and start of the measurement on the sensor insoles (which is larger than the delay when stopping), the stop signal (flash/beep) should be preferred over the start signal for synchronization.

## 5 Time Alignment

### 5.1 Channels and Channel Groups

Inside the OpenGo Software, data from different measurement systems (left sensor insole, right sensor insole, external data) is organized in channel groups. Think of each channel group as being a spreadsheet table, with the first column being the time channel, and all further columns being the data channels (in the case of sensor insoles: pressure sensors, total force, acceleration, etc.).

Each channel group starts at an absolute time point, and the time channel is then a vector of relative offsets to this start time.

This way of data organization offers great flexibility: Not only can different measurement systems have their data starting at different points in time. It also supports differing sampling rates. Since the time of each data sampled is stored, there is not even a need for a fixed sampling rate, and dropped/missing data is correctly reflected in the .go file format.

What this means for the user:

- There are no limitations in having e.g. sensor insole data at 100 Hz along with e.g. force plate data at 200 Hz in the same measurement.
- Setting (or fine-tuning) the data time alignment is simply a matter of changing the start time of a channel group. To do so:
  - Go to the Analyze screen.
  - Select the measurement of interest.
  - Select the “Edit Channels” tool from the corresponding button dropdown, and click on it.
  - Select the channel group of interest, and click on “Edit”.
  - Edit the start time of the channel group, and click OK.
  - The change has immediate effect on the data, and the currently configured widgets are automatically updated.
- Things are even easier for video, see <https://account.moticon.com/doc-opengo-video#video-settings>:
  - Configure different widgets in the Analyze section for replaying both data and video. Note: If you have multiple videos from different view points, you can configure multiple video widgets for simultaneous replay.
  - Use the play bar or line graph widget to set the time cursor at a time point of interest observed in the video.
  - In the video widget, click on the lock button, which will freeze the video at the current frame.
  - Use the play bar or line graph widget to set the time cursor to the point matching the video.



- Release the lock button in the video widget.

The resulting video offset can be edited in the video folder, located in the measurement tree.

## 5.2 Text Export Resampling

When using the text export tool of the Report section, all available numeric data (including imported external data) is exported as one big table, having a common time channel as first column. Since, originally, each channel group has its own time channel, this means that the data has to be resampled.

This is of great benefit for users who, at this point, can execute algorithms on exported text files without having to deal with time alignment, resampling etc. Even event channels are exported as (sparse) columns.

Note that the resampling may be quite time-consuming, depending on the amount of data.

## 5.3 Data Units

Not really related to synchronization, but still good to know: Just as sensor insole data, imported (external) data can be associated with units (both in UPD import and text import), such as Newton, volt, etc. The line graph widget can show up to two y-axes (left/right vertical axis), where data channels having equal units share the same y-axis.

This does not affect the manual time alignment, which works along the x-axis.