Otii Product Suite

User Manual

QOITECH

1 Otii products

⚠ Otii is highly addictive

Otii Product Suite is state-of-the-art solution for power profiling, battery life estimation, battery profiling and emulation, and energy harvesting testing for embedded electronics. The Otii products are designed for every-day power analysis and testing at the developer's bench. They are also scalable for automated power testing and optimization across teams, product lines, and development stages.

One comprehensive product suite. 20+ powerful capabilities designed to bring you and your team closer to optimal battery life of your embedded product.

Otii instruments

- Otii Ace Pro (≤ 25V, 5A)
- Otii Arc Pro (≤ 5V, 5A)

Otii instruments come with powerful free-of-charge and perpetual Otii desktop application for Windows/Ubuntu/macOS. Otii desktop application is on version 3, called Otii 3 software.

What is included?

The Otii instrument arrives in a box; depending on the version chosen, it will be labeled as Otii Arc or Otii Ace.

Once opened, you will find:

- Otii Arc Pro / Otii Ace Pro hardware unit
- USB cable
- Adaptor C (only for Otii Ace Pro)

Otii Toolboxes

You can upgrade your Otii instruments with additional specialized feature sets using Otii Toolboxes. These are shareable licenses available as monthly, yearly, or perpetual subscriptions.

Important to know! - An Otii Toolbox license is associated with a user, not a specific piece of hardware. - The license can be shared; a user can share it with another user, but it can only be used by one user at a time. - You can also configure a license pool that automatically shares the licenses

among your automation setups. When using a shared pool, your test scripts can be configured to wait for a license to become available before execution.

2 Getting started

Start energy optimizing embedded systems and IoT devices with the **Otii Product Suite** in a five steps:

1. Download and install the **Otii 3 Desktop App**. If not already done, follow these instructions.
2. Connect the Otii hardware to the computer and the DUT. If not already done, follow these instructions.
3. Open the **Otii 3 Desktop App** and create a new project. If the Otii hardware is connected correctly, it will be identified in the left sidebar.
4. Click the *Add* button next to the **Otii hardware** identified to add it. Set the parameters needed for the DUT (e.g., voltage, overcurrent protection) and select the desired channels to be measured.
5. Press the *Record* button, dhen turn on the **Otii hardware** to power up the DUT and measure the selected channels.

Refer to the Wiring up page to explore more ways to connect your hardware, including how to expand the measurement capabilities using the expansion port.

3 Installation

Getting started with the **Otii software** is simple and fast.

Download the latest version of the **Otii 3 Desktop App** directly from the Qoitech User Management. Multiple operating systems (OS) are supported, including Windows, macOS, Ubuntu and Raspberry Pi. Detailed instructions are below.

3.1 Windows

3.1.1 Requirements

- Windows 10 version 1903 or later (Intel 64-bit)
- 16 GB of RAM

3.1.2 Installation

1. Visit the Qoitech User Management portal, and access your account. In case of not having one already, create it by simply clicking on Create an account – once created, the account must be verified with the associated email address to activate it.
2. Within the User Management portal, navigate to the Download page on the left bar and download the latest version of the Otii 3 Software.
3. To download it, the EULA terms must be accepted by clicking the check box.
4. Select the version for Windows.
5. Once downloaded, open the .exe file by double-clicking. The application takes a couple of seconds to install and will open automatically when finished.
6. The application's welcome interface will be opened. Here, you can 1) Create a new Otii project, 2) Open an Otii project, and 3) Read the documentation.

3.1.3 Silent installation

You can also install the software silently from the command line:

```
PS C:\> .\Setup_otii_x64_3.x.x.exe /s
```

Explore the documentation below to learn more about Otii 3 Desktop App's main window layout and keyboard navigation.

3.2 Ubuntu

3.2.1 Requirements

- Ubuntu 20.04 or later (Intel 64-bit)
- 16 GB of RAM

3.2.2 Installation

1. Visit the Qoitech User Management portal, and access your account. In case of not having one already, create it by simply clicking on “Create an account” – once created, the account must be verified with the associated email address to activate it.
2. Within the User Management portal, navigate to the Download page on the left bar and download the latest version of the Otii 3 Software.
3. To download it, the EULA terms must be accepted by clicking the check box.
4. Select the version for Ubuntu.
5. Once downloaded, open the .deb file by double-clicking. The application takes a couple of seconds to install and will open automatically when finished.
6. The Otii 3 Desktop App’s welcome interface will be opened. Here, you can 1) Create a new Otii project, 2) Open an Otii project, and 3) Read the documentation.

3.3 macOS

3.3.1 Requirements

- macOS 11.0 or later (Apple Silicon)
- macOS 10.15 or later (Intel)
- 16 GB of RAM

3.3.2 Installation

1. Visit the Qoitech User Management portal, and access your account. In case of not having one already, create it by simply clicking on “Create an account” – once created, the account must be verified with the associated email address to activate it.
2. Within the User Management portal, navigate to the Download page on the left bar and download the latest version of the Otii 3 Software.
3. To download it, the EULA terms must be accepted by clicking the check box.
4. Select the version for macOS based on the computer’s characteristics.
5. Once downloaded, open the .dmg file by double-clicking. Then, move the Otii 3 Desktop App to the computer’s Applications folder to finish the installation.

6. Once installed, look for “Otii 3” within your applications and start it. The application’s welcome interface will be opened. Here, you can 1) Create a new Otii project, 2) Open an Otii project, and 3) Read the documentation.

3.4 Raspberry PI

3.4.1 Requirements

- Raspberry Pi 4B rev 1.2 8 GB or Raspberry Pi 5 8GB
- NVME disk or USB SSD disk
- External power supply for the Otii device
- Raspberry Pi OS 64-bit

⚠ Otii server will run on a Raspberry Pi 4/5 with less than 8 GB RAM, but it is not recommended

⚠ Running the system on an SD card will degrade performance, and the memory card will likely quickly wear out. Use a NVME disk or USB SSD disk instead, as recommended.

3.4.2 Installation

The Raspberry Pi release does only include **otii_server**, not the Otii Desktop.

1. Visit the Qoitech User Management portal, and access your account. In case of not having one already, create it by simply clicking on “Create an account” – once created, the account must be verified with the associated email address to activate it.
2. Within the User Management portal, navigate to the Download page on the left bar and download the latest version of the Otii 3 Software.
3. To download it, the EULA terms must be accepted by clicking the check box.
4. Select the version for Raspberry Pi.

Once downloaded install the .deb file:

```
sudo dpkg -i otii-server_3.x.x_arm64.deb
```

4 Wiring up

Explore hardware setup examples for a variety of use cases.

Otii hardware features two main connectors and one expansion port. Multiple channels can be measured with the available inputs, such as main current, main power, main voltage, ADC current, ADC voltage, SENSE+ voltage, SENSE- voltage, GPI1, and GPI2, as well as capturing serial logs from any embedded system or IoT devices.

To get the most out of the **Otii Arc Pro** and **Otii Ace Pro**, start by checking out the Expansion port to explore how to enhance measurement capabilities.

4.1 Power supply

Otii instruments are powered via USB cable from the computer. Both Otii Arc and Ace Pro can be powered by an external DC adapter. When using an external DC adapter, choose one from an established supplier.

The adapter should be low-noise, comply with IEC 60950-1 as a Limited Power Source, and have an output of 7 to 9 VDC for the Otii Arc Pro and 7 to 20 VDC for the Otii Ace Pro, with a maximum of 5 A.

Visit the FAQ page for more information on selecting the proper power supply.

4.2 Grounding (GND)

Otii Ace Pro is an isolated power supply. There are three isolation domains, the main channel (banana connectors), the expansion port and the USB/DC adaptor plug. When using the expansion port, there is a need to also connect AGND (Analog Ground) or DGND (Digital Ground) to DUT Ground. Otherwise, the expansion port will be floating.

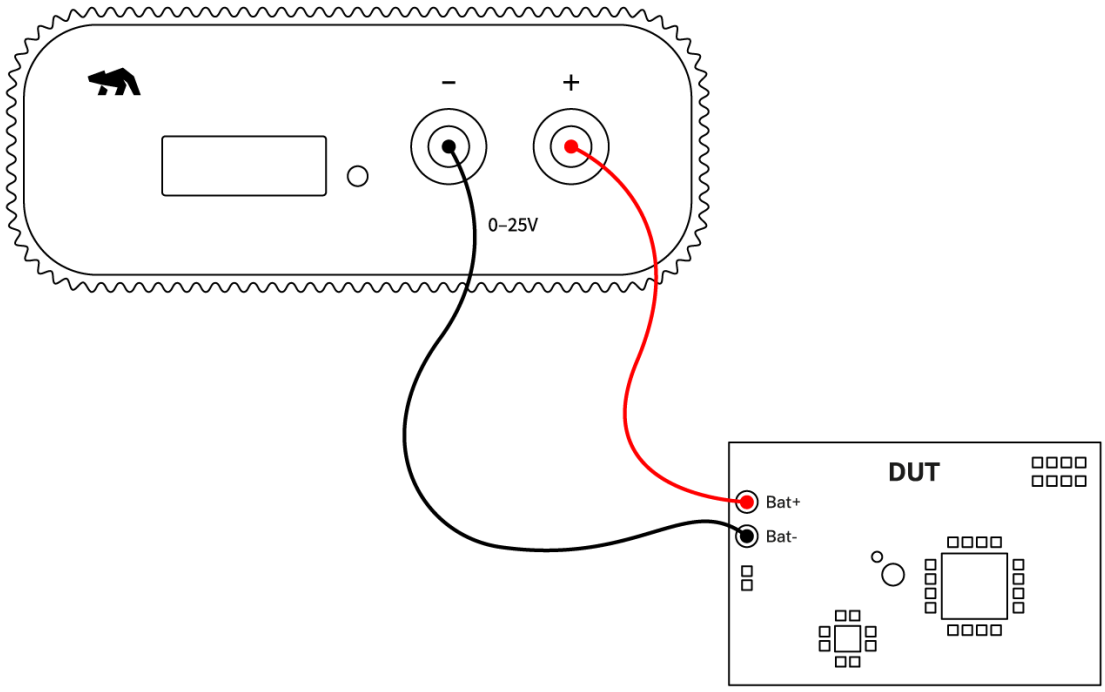
Otii Arc Pro is not an isolated power supply, but we highly recommend to use a separate ground wire for the signal ground, attached to either DGND or AGND on the expansion port. This will prevent disturbances and measurement errors caused by voltage drops in the negative battery wire.

- AGND is utilized as a reference point for analog measurements
- DGND is used as a reference point for digital signaling

Internally **Otii hardware**, AGND and DGND are connected for both **Otii Ace Pro** and **Otii Arc Pro**

4.3 Measuring DUT

Power source and measure a Device Under Test (DUT).



Set up **Otii hardware**, either the **Otii Arc Pro** or **Otii Ace Pro**, to work as a power source box providing constant voltage to the DUT, ranging from `0.5 V – 5 V` / `0 V – 25 V`, and enable it to measure output current and voltage.

4.3.1 Otii hardware setup

1. Connect **Otii hardware** to your computer via USB. If needed, add a DC power supply via the DC plug.
2. Connect **Otii hardware**'s banana connector positive lead to **DUT positive (+)** battery connector/power connector.
3. Connect **Otii hardware**'s banana connector negative lead to **DUT negative (-)** battery connector/power connector (GND).

⚠ In case of connecting the DUT using the battery connector, remove the battery prior to connecting it to the Otii hardware.

4.3.2 Otii software setup

1. Open the **Otii 3 Desktop App**, then select *Create a new Otii Project*
2. In the left sidebar, find the *CONTROL* section. Within this section, add the **Otii hardware** identified by your computer. Just click the *add* button on the right side of the hardware identified.

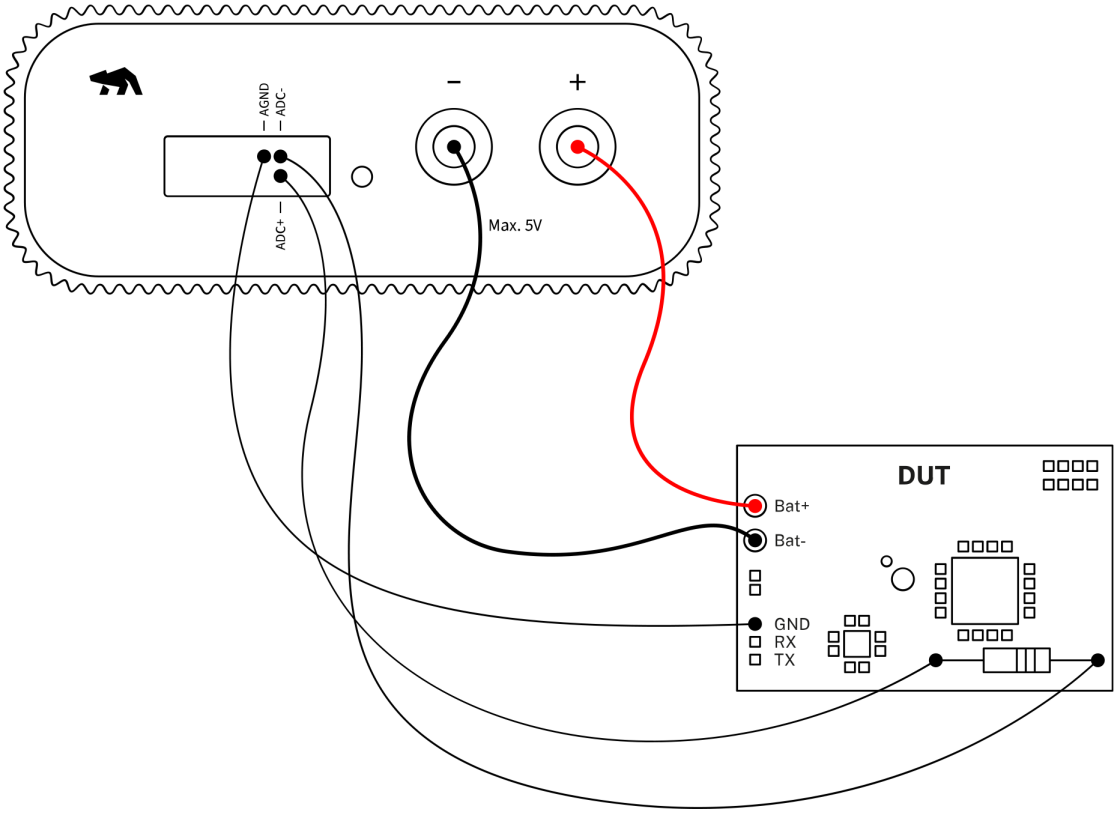
3. Considering the DUT's power ranges, set the required voltage for the added **Otii hardware** in the voltage box located on the left side of the power supply button.
4. Under *General Settings*, make sure that the *Power Box* option is selected, and set an OC (overcurrent) protection for the DUT.
5. Under *Channels*, select *Main current* to measure DUT's output current and *Main voltage* to measure DUT's output voltage. The *Main power* channel is assigned by default when the main current is selected to measure DUT's energy consumption.
6. [Optional] The previously selected channels will be listed under the *MEASUREMENTS* section. In this section, the sampling rate, up to 50 ksps, can be set. (Only possible with the **Otii Ace Pro**)
7. Click the *record* button in the upper left corner of the toolbar to start recording DUT measurements. Since your DUT is not powered on yet, only noise measurements will be observed at first.
8. Turn on the DUT by clicking the *power button* next to the desired voltage assigned. Once turned on, the DTU measurements are being recorded.
9. Now, it's time to validate, analyze, and optimize your embedded system or IoT devices.

ⓘ Use wires with the lowest possible resistance and keep them as short as possible to minimize voltage drop, especially when the DUT consumes high current. The resistance in wires will be added to the total series resistance within the system, causing an unwanted voltage drop and an error in power and energy consumption measurements.

ⓘ If the DUT consumes high current during power-up, causing high voltage drops in the attached wires, which directly affects the power-up cycle of the DUT since the voltage is too low. In case of experiencing this, connect one or several capacitors from Bat+ to Bat- on the DUT side to stabilize the voltage to start up the DUT properly. Note that the Otii Arc and Otii Ace have very low output capacitance for being a power source for the DUT but also a programmable load.

4.4 Measuring a subsystem

Power source and measures subsystem's current and voltage.



Set up **Otii hardware**, **Otii Arc Pro**, or **Otii Ace Pro** to measure a subsystem current and voltage over a shunt resistor in addition to measuring the complete system. Note that the resistor can be part of the actual system or applied externally.

4.4.1 Otii hardware setup

1. Connect **Otii hardware** to your computer via USB. If needed, add a DC power supply via the DC plug.
2. Connect **Otii hardware**'s banana connector positive lead to **DUT positive (+)** battery connector/power connector.
3. Connect **Otii hardware**'s banana connector negative lead to **DUT negative (-)** battery connector/power connector (GND).
4. Connect **Otii hardware**'s ADC+ and ADC- across the shunt resistor for current measurement; no additional ground is needed for this. Make sure to connect the shunt resistor side with the high side to ADC+ and the one with the low side to ADC-. Note that if negative currents are measured, the high side has a lower potential than the low side.
5. [Optional] To measure voltage, connect AGND to the DUT's ground and measure the voltage between ADC+ and AGND.

⚠ In case of connecting the DUT using the battery connector, remove the battery prior to connecting it to the Otii hardware.

4.4.2 Otii software setup

1. Open the **Otii 3 Desktop App**, then select *Create a new Otii Project.*
2. In the left sidebar, find the *CONTROL* section. Within this section, add the **Otii hardware** identified by your computer. Just click the add button on the right side of the hardware identified.
3. Considering the DUT's power ranges, set the required voltage for the added **Otii hardware** in the voltage box located on the left side of the power supply button.
4. Under *General Settings*, make sure:

   - *Power Box* option is selected.
   - Set an OC (overcurrent) protection for the DUT.
   - Set the value of the shunt resistor in the *ADC resistor* field. Check how to choose the resistor if you are not sure how to do it.

5. Under *Channels*, select *ADC current* to measure the subsystem current over a shunt resistor. In the case of measuring voltage, the "ADC voltage" must be selected as well. If the system itself needs to be measured, it can also be measured; select the channels desired to be measured in addition to the ones previously selected.
6. [Optional] The previously selected channels will be listed under the MEASUREMENTS section. In this section, the sampling rate, up to 50 ksps, can be set. (Only possible with the **Otii Ace Pro**)
7. Click the record button in the upper left corner of the toolbar to start recording DUT measurements. Since your DUT is not powered on yet, only noise measurements will be observed at first.
8. Under the *CONTROL* section, turn on the DUT by clicking on the power button located right next to the desired voltage assigned. Once turned on, the DTU measurements are being recorded.
9. Now it's time to validate, analyze, and optimize your embedded system or IoT devices.

ⓘ The voltage over the shunt resistor is measured differentially. No extra signal ground is needed if you only want to measure the current.

ⓘ If voltage measurement is required, it is measured between ADC+ and AGND. In this case, AGND needs to be connected to the ground of the DUT.

⚠ The maximum voltage range on ADC pins are 0 V to 5 V for Otii Arc Pro and -10 V to 25 V for Otii Ace Pro

4.4.3 How to choose the resistor

4.4.3.1 For Otii Arc Pro The differential input voltage range for the **Otii Arc Pro** ADC inputs goes from -81.9175 mV to 81.92 mV. Based on these values, the absolute maximum (peak) current through the sensing resistor must be determined.

Remember that by using Ohm's Law (V = I x R | I = V / R | R = V / I), the relationship between voltage, current, and resistance in an electrical circuit can be determined.

Here's an example to illustrate how to choose the resistor:

Suppose the system has a maximum peak current of 200 mA. By calculating, 0.08192 V / 0.2 A, the resistor value is equal to 0.41 ohms. But, the closest standard resistor value is 0.39 ohms. Resulting in a measuring range of 0.08192 V / 0.39 ohms = 0.210 A.

In conclusion, a 0.39 ohm sensing resistor will result in a measurement range of +/- 0.210 A.

4.4.3.2 For Otii Ace Pro Following the same example as above, let's do the same for now but for the **Otii Ace Pro**.

The differential input voltage range for the **Otii Ace Pro** ADC inputs goes from -102.4 mV to 102.4 mV. Based on these values, the absolute maximum (peak) current through the sensing resistor must be determined.

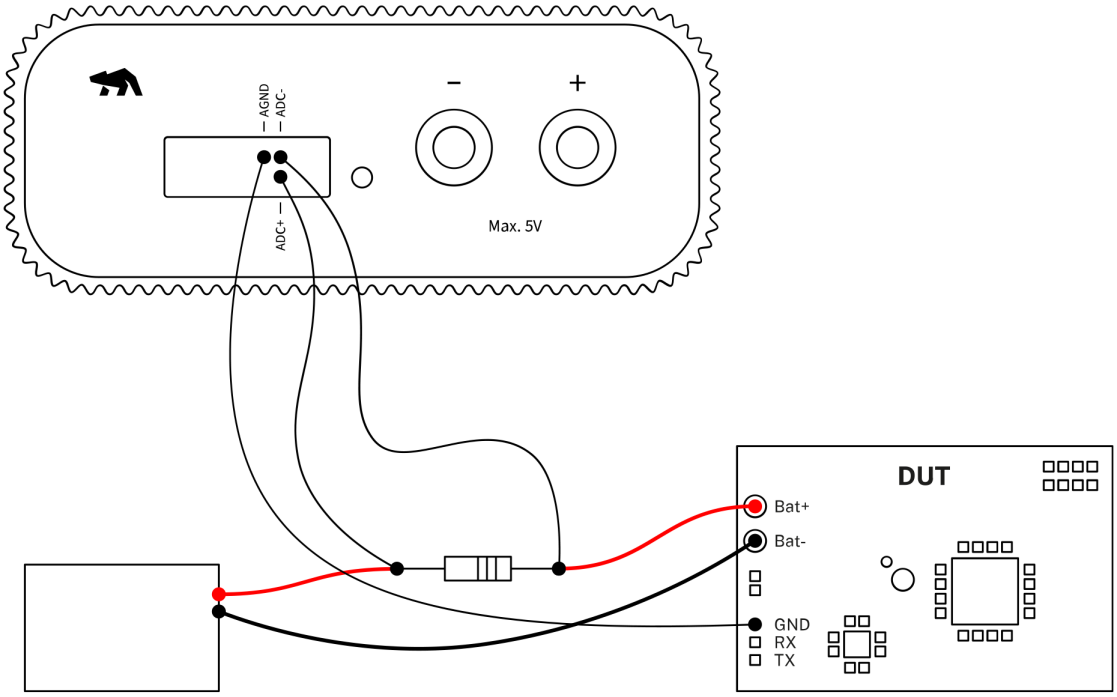
Here's an example to illustrate how to choose the resistor:

Suppose the system has a maximum peak current of 200 mA. By calculating, 0.1024 V / 0.2 A, the resistor value is equal to 0.512 ohms. But, the closest standard resistor value is 0.51 ohms. Resulting in a measuring range of 0.1024 V / 0.51 ohms = 0.200 A.

In conclusion, a 0.51 ohm sensing resistor will result in a measurement range of +/- 0.200 A.

4.5 Ampere meter mode

Measures DUT's current and voltage while it is powered by an external power source.



If the Device Under Test (DUT) needs to be powered from an external source while still measuring system current and voltage, an external shunt resistor connected to the ADC inputs of the **Otii hardware**, whether the **Otii Arc Pro** or **Otii Ace Pro**, can manage this.

4.5.1 Otii hardware setup

1. Connect **Otii hardware** to your computer via USB. If needed, add a power supply via a DC power adaptor.
2. Connect the external shunt resistor in series between the external power supply's positive (+) line and DUT's positive (+) battery connector.
3. Connect ADC+ and ADC- across the resistor to measure the voltage over the resistor.
4. Connect **Otii hardware**'s AGND to the DUT's GND.

4.5.2 Otii software setup

1. Open the **Otii 3 Desktop App**, then select *Create a new Otii Project*.
2. In the left sidebar, find the *CONTROL* section. Within this section, add the Otii hardware identified by your computer. Just click the *Add* button on the right side of the hardware identified.
3. Under *General Settings*, make sure:

- *Power Box* option is selected.

- Set the value of the shunt resistor in the *ADC resistor* field. Check how to choose the resistor if you are not sure how to do it.

4. Under *Channels,* select *ADC current* to measure the current over a shunt resistor. In the case of measuring voltage over the shunt resistor, the *ADC voltage* must be selected as well.
5. [Optional] The previously selected channels will be listed under the *MEASUREMENTS* section. In this section, the sampling rate, up to `50 ksps`, can be set. (Only possible with the **Otii Ace Pro**)
6. Click the record button in the upper left corner of the toolbar to start recording DUT measurements. Since your DUT is not powered on yet, only noise measurements will be observed at first.
7. Under the *CONTROL* section, turn on the DUT by clicking the power button. Once turned on, the DUT measurements are being recorded.
8. Now it's time to validate, analyze, and optimize your embedded system or IoT devices.

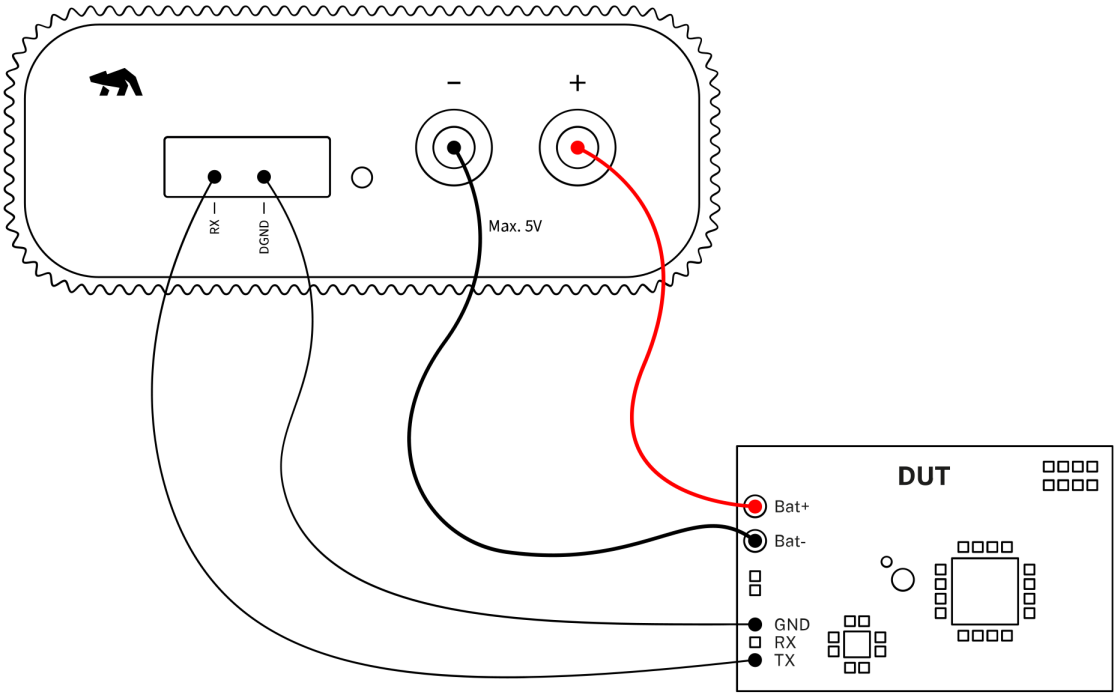
ⓘ The voltage over the shunt resistor is measured differentially. No extra signal ground is needed if you only want to measure the current.

ⓘ If voltage measurement is required, it is measured between ADC+ and AGND. In this case, AGND needs to be connected to the ground of the DUT.

⚠ The maximum voltage range on ADC pins are 0V to 5 V for Otii Arc Pro -10V to 25V for Otii Ace Pro

4.6 Capture UART logs

Capture and record Device Under Test (DUT) serial logs.



Set up **Otii hardware**, **Otii Arc Pro**, or **Otii Ace Pro** to capture logs from the DUT serial log.

4.6.1 Otii hardware setup

1. Connect **Otii hardware** to your computer via USB. If needed, add a DC power supply via the DC plug.
2. Connect **Otii hardware**'s banana connector positive lead to **DUT positive (+)** battery connector/power connector.
3. Connect **Otii hardware**'s banana connector negative lead to **DUT negative (-)** battery connector/power connector (GND).
4. Connect **Otii hardware**'s RX pin to the DUT TX pin.
5. Connect **Otii hardware**'s DGND to the DUT GND pin.

⚠ In case of connecting the DUT using the battery connector, remove the battery prior to connecting it to the Otii hardware.

4.6.2 Otii software setup

1. Open the **Otii 3 Desktop App**, then select *Create a new Otii Project*.
2. In the left sidebar, find the *CONTROL* section. Within this section, add the **Otii hardware** identified by your computer. Just click the add button on the right side of the hardware identified.

3. Considering the DUT's power ranges, set the required voltage for the added **Otii hardware** in the voltage box located on the left side of the power supply button.
4. Under *General Settings*, make sure:

   - *Power Box* option is selected.
   - Set digital voltage according to the DUT's digital voltage level for the UART
   - Set an OC (overcurrent) protection for the DUT.

5. Under *Channels*, choose the measurements desired to be recorded (e.g., main current, main voltage, main power, among others)
6. Under *UART*, make sure:

   - Assign the DUT's baud rate.
   - Select *UART log* to enable the serial communication. Once enabled, the console log will automatically open.

7. Click the record button in the upper left corner of the toolbar to start recording DUT measurements. Since your DUT is not powered on yet, only noise measurements will be observed at first.
8. Under the *CONTROL* section, turn on the DUT by clicking on the power button located right next to the desired voltage assigned. Once turned on, the DUT measurements are being recorded.
9. Now it's time to validate, analyze, and optimize your embedded system or IoT devices.

4.7 4-wire mode

In 4-wire mode, also known as Kelvin connection, the voltage measurement (sensing) is separated from the power. The reason for separating the cables is that in the cables that also powers the Device Under Test (DUT) there will be a voltage drop, due to the cable resistance and the current. In the sensing cables, there is no, or very low, current, so there is very low voltage drop. When 4-wire is enabled, there will be 4W symbol highlighted next to the voltage measurement, to show that the voltage is measured from the SENSE pins.

In 4-wire mode, the regulated voltage will not be changed, only the measured voltage.

There are requirements on the SENSE pin measurement to have 4-wire active and they differ from **Otii Arc Pro** and **Otii Ace Pro**. For Arc, (SENSE+) must be withing 400mV from (Main+) and (SENSE-) must be within 400mV from (Main-), that also is GND. For Ace, (SENSE+) - (SENSE-) must be within 500mV from (Main+) - (Main-)

4.7.1 Otii hardware setup

1. Connect **Otii hardware** to your computer via USB. If needed, add a DC power supply via the DC plug.

2. Connect **Otii hardware**'s banana connector positive lead to **DUT positive (+)** battery connector/power connector.
3. Connect **Otii hardware**'s banana connector negative lead to **DUT negative (-)** battery connector/power connector (GND).
4. Connect **Otii hardware**'s SENSE+ pin to **DUT positive (+)** battery connector/power connector.
5. Connect **Otii hardware**'s SENSE- pin to **DUT negative (-)** battery connector/power connector (GND).
6. Connect **Otii hardware**'s AGND pin to **DUT GND**

4.7.2 Otii software setup

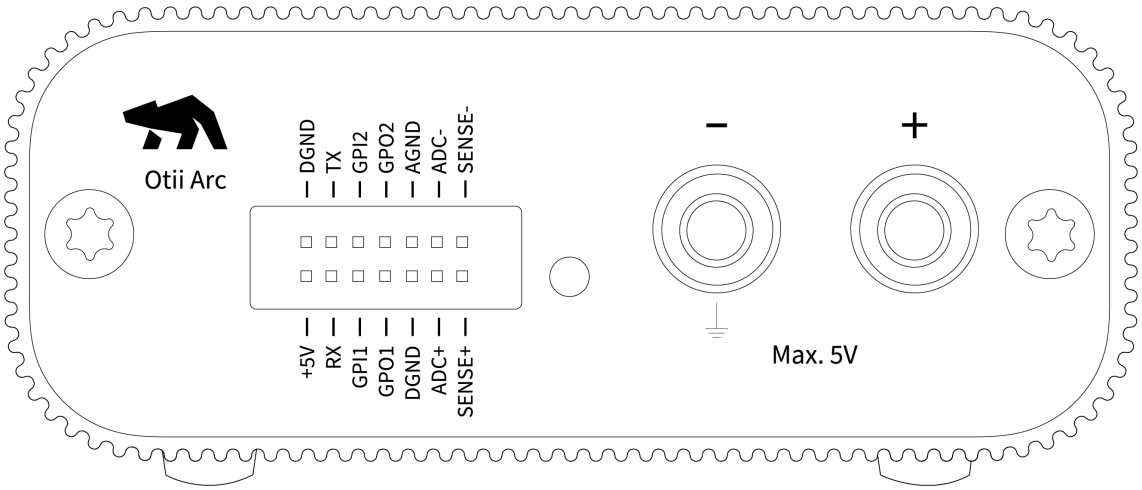
1. Open the **Otii 3 Desktop App**, then select *Create a new Otii Project*.
2. In the left sidebar, find the *CONTROL* section. Within this section, add the Otii hardware identified by your computer. Just click the *Add* button on the right side of the hardware identified.
3. Considering the DUT's power ranges, set the required voltage for the added **Otii hardware** in the voltage box located on the left side of the power supply button.
4. Under *General Settings*, make sure:

   - *Power Box* option is selected.
   - *4-wire* is selected.

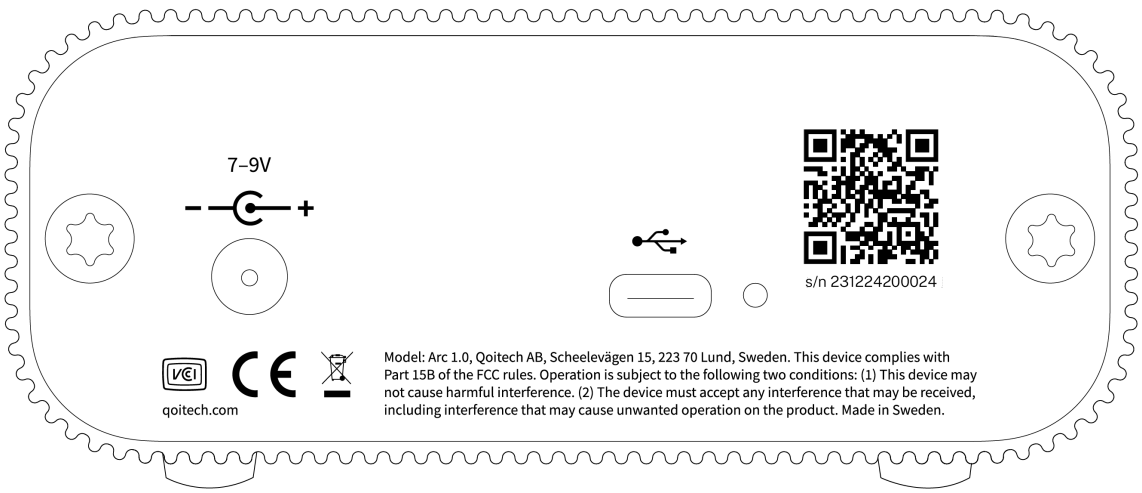
5. Under *Channels,* select *Main current* and *Main voltage* to measure the current and voltage.
6. [Optional] The previously selected channels will be listed under the *MEASUREMENTS* section. In this section, the sampling rate, up to `50 ksps`, can be set. (Only possible with the **Otii Ace Pro**)
7. Click the record button in the upper left corner of the toolbar to start recording DUT measurements. Since your DUT is not powered on yet, only noise measurements will be observed at first.
8. Under the *CONTROL* section, turn on the DUT by clicking the power button. Once turned on, the DUT measurements are being recorded.
9. Now it's time to validate, analyze, and optimize your embedded system or IoT devices. The Main Voltage measurement is now from the SENSE pins, instead of the banana connectors. # Otii Arc Pro

4.8 Hardware overview

The front side of the **Otii Arc Pro** has the main connectors, additional ports to extend measurement capabilities, plus a status LED.

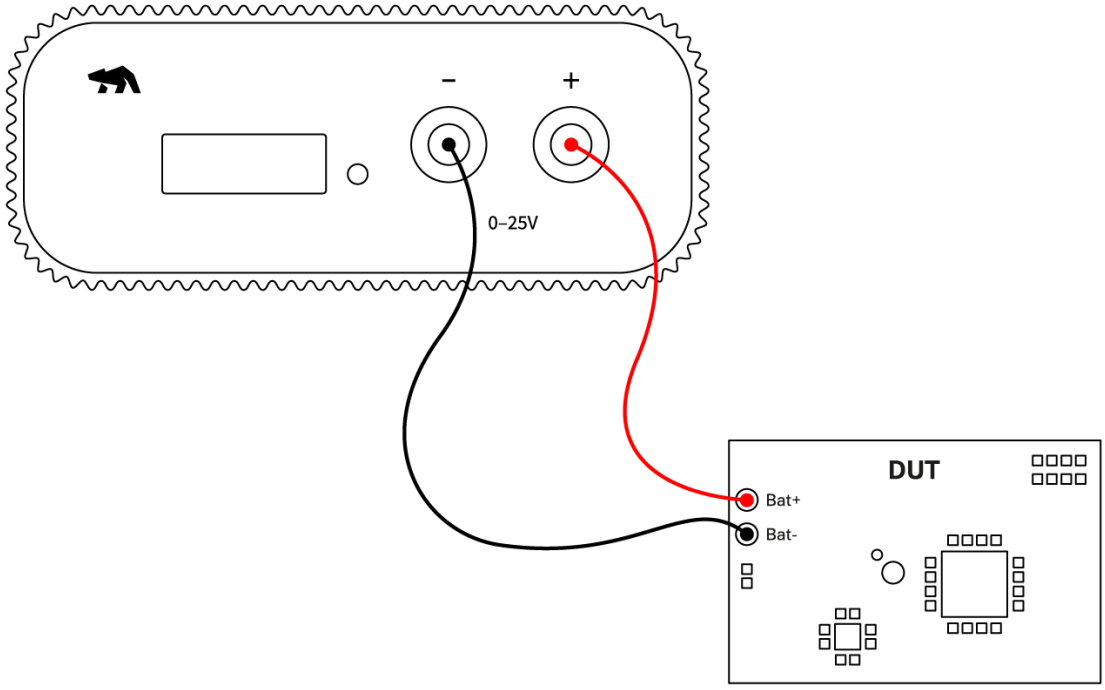


The back side of the **Otii Arc Pro** has the host Micro-USB connector and an input socket to power up the unit with an external DC adapter when required.

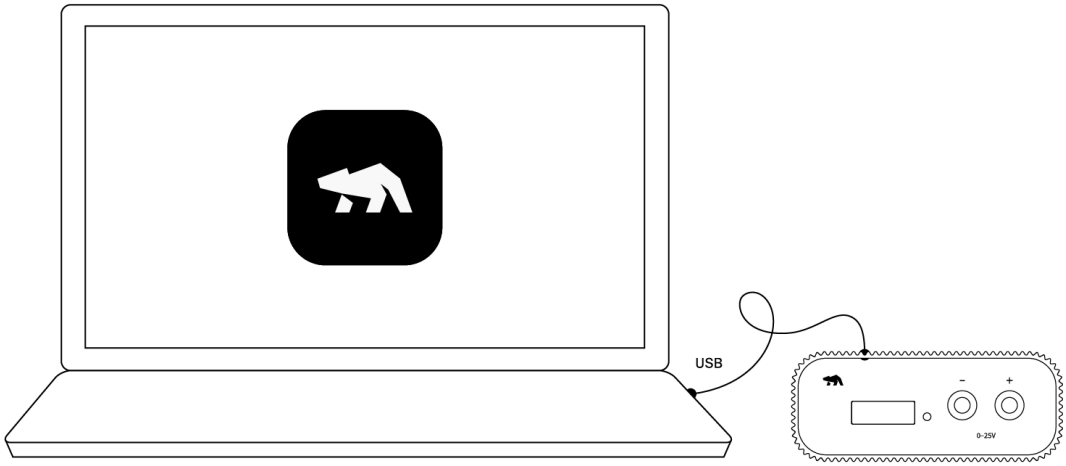


4.9 Main connections

The **Otii Arc Pro** has the voltage(+) and voltage(-) binding post (red and black connectors located on the right side of the instrument), which are used to source the DUT but also to measure the current, voltage, and power of the device. The DUT can be connected through its battery connectors or DC input; refer to the connection diagram below for better understanding.



To power up the Otii Arc Pro and enable communication with the host computer, connect the unit to your computer using the provided USB micro B to USB A cable.



Check out the Wiring up page to explore all possible connection configurations between the Otii Arc Pro and your DUT.

4.10 Expansion Port

The expansion port on the front of the **Otii Arc Pro** is designed to enhance the hardware's functionality, allowing users to measure additional voltages and currents or trigger external events. These additional pins feature multi-input and multi-output for analog and digital signals, serial

communication, and additional power and ground points. Below is a general overview of the additional pins available:

Connector	Description	Maximum rating
+5V	5 V output	500 mA
DGND	Digital ground (return current)	
RX	UART RX / Digital Input (Exclusively under Scripting)	0 V to 5.5 V
TX	UART TX / Digital Output (Exclusively under Scripting)	0 V to 5.5 V
GPI1	Digital input 1	0 V to 5.5 V
GPI2	Digital input 2	0 V to 5.5 V
GPO1	Digital Output 1	0 V to 5.5 V
GPO2	Digital Output 2	0 V to 5.5 V
DGND	Digital ground (return current)	
AGND	Analog ground (return current) / Analog measurements	
ADC+	Differential input for current measurement. It also measures single ended voltage with respect to AGND	0 V to 5.5 V
ADC-	Input for current measurement	0 V to 5.5 V
SENSE+	Analog input with the AGND as reference/return	0 V to 5.5 V
SENSE-	Analog input with the AGND as reference/return	0 V to 5.5 V

4.11 Status LED

The table below explains the **Otii Arc Pro**'s status LED behaviors and their meanings to help you quickly identify the device's state.

LED behavior	Otii Arc Pro Status
Flashing green	Otii Arc Pro is active
Solid green	Relay closed, sinking/sourcing power

4.12 Arc Tech spec

4.12.1 General

Operating environment	Min	Typical	Max
Temperature	10°C / 50°F		30°C / 86°F
Humidity	30 %		60 %

USB power supply [1]	Min	Typical	Max
Output voltage (auto range)	0.5 V		3.75 V
Output voltage (locked to high current range)	0.5 V		4.2 V
Output voltage setting resolution		1 mV	

USB power supply [1]	Min	Typical	Max
Output current, max continuous [2]		up to 1 A [3]	2.4 A
Output current, max peak [4]		up to 1 A [5]	5 A

External 7.5 v - 9 v power supply [6]	Min	Typical	Max
Output voltage (auto range)	0.5 V		4.55 V
Output voltage (locked to high current range)	0.5 V		5.0 V
Output voltage setting resolution		1 mV	
Output current, max continuous [7]			2.4 A
Output current, max peak [8]			5 A

Programmable current sink (requires an Otii Toolbox)	Min	Typical	Max
Sink current	0 A		2.5 A
Sink current, setting resolution		39 µA	
Sink voltage, USB power supply	0.85 V [9]		4.2 V
Sink voltage, external power supply	0.85 V [10]		5.0 V

Current measurement	Min	Typical	Max
Accuracy		±(0.1 % + 50 nA)[11]	
Sample Rate in ±19 mA range		4 ksps	
Sample Rate in ±2.7A range		1 ksps	
Sample Rate in ±5.0 A range		1 ksps	
Analog bandwidth (3 dB)		400 Hz	

Voltage measurement	Min	Typical	Max
Total accuracy		±(0.1 % + 1.5 mV)	
Sample rate		1 ksps	

[2] Depends on chosen power supply. Otii Arc Pro will monitor internal temperature and cut off if temperature limit is reached.
[3] USB power capacity and reliability in laptops and desktops greatly depend on host USB port/cable design.
[4] Depends on chosen power supply. Otii Arc Pro will monitor internal temperature and cut off if temperature limit is reached.
[5] USB power capacity and reliability in laptops and desktops greatly depend on host USB port/cable design.
[1] USB power capacity and reliability in laptops and desktops greatly depend on host USB port/cable design.
[7] Depends on chosen power supply. Otii Arc Pro will monitor internal temperature and cut off if temperature limit is reached.
[8] Depends on chosen power supply. Otii Arc Pro will monitor internal temperature and cut off if temperature limit is reached.
[6] See list of recommended external power supplies and powered USB hubs at our FAQ.
[9] Sink voltage can go below this specification if locked to high range. It is possible to go down to 0.5 V if the sink current is below 1.9 A. For currents below 19 mA, the measurement will have more noise when locked to high range than in auto range.
[10] Sink voltage can go below this specification if locked to high range. It is possible to go down to 0.5 V if the sink current is below 1.9 A. For currents below 19 mA, the measurement will have more noise when locked to high range than in auto range.
[11] Up to 19 mA current in auto range, for higher currents, the accuracy is ±(0.1 % + 150 µA). Average > 1s.

4.12.2 Expansion port

UART	Min	Typical	Max
Bitrate	110 bps		5.25 Mbps

Digital I/O [12]	Min	Typical	Max
Digital IO operating voltage	1.2 V	Vio [13]	5 V [14]
VIL Low-level input voltage			Vio * 0.2 V
VIH High-level input voltage	Vio * 0.8 V		
Imax, Max sink/source current (total for GPIOs)	-10 mA		10 mA

Differential ADC, pins ADC- and ADC+ [15]	Min	Typical	Max
Voltage input	0 V		5 V
Shunt voltage range	-81.9175 mV		81.2 mV
Resolution		2.5 µV	
Accuracy		±(0.1 % + 10 µV)	
Input impedance		220 kΩ	

Single ended ADC, pins ADC- and ADC+ [16]	Min	Typical	Max
Voltage input	0 V		5 V
Resolution		1.25 mV	
Accuracy		±(0.1 % + 7.5 mV)	
Input impedance		830 kΩ	

SENSE, pins SENSE- and SENSE+	Min	Typical	Max
Voltage input	0 V		5 V
Resolution		1.5 mV	
Accuracy		1 %	
Input impedance		1 MΩ	

Expansion port power supply	Min	Typical	Max
Output voltage		5 V [17]	
Output current		500 mA	

[13] Expansion Port Digital voltage level is set by user in Otii SW.
[14] Maximum voltage will depend on your USB power supply and USB cable.
[12] See Nexperia SN74LVC8T245 for details.
[15] See TI INA226 for details.
[16] See TI INA226 for details.
[17] Depends on available input power

4.12.3 USB and DC jack

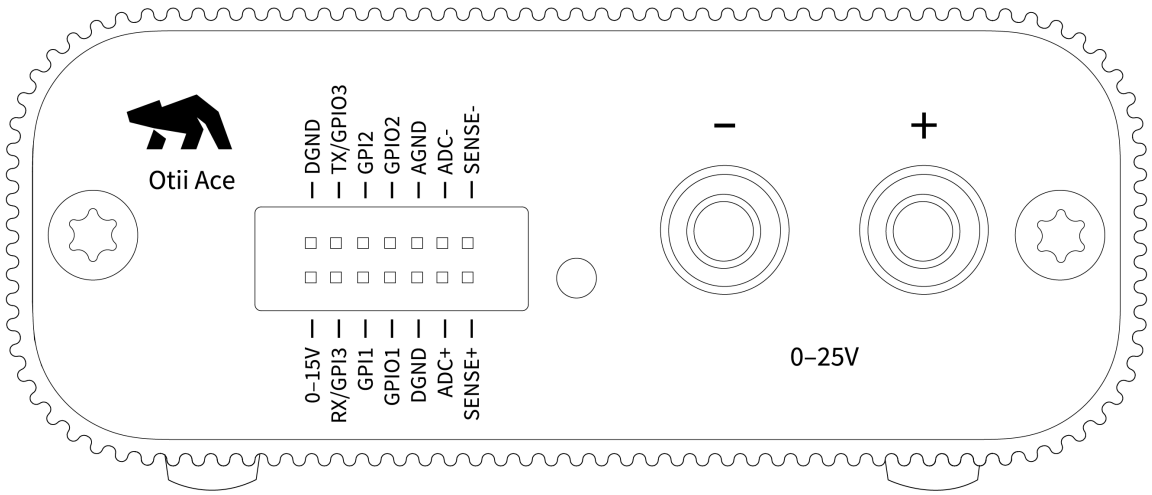
DC jack	Min	Typical	Max
Input voltage	7 V		9 V
Input current			5 A

USB	Min	Typical	Max
VBUS voltage	4.75 V		5.25 V
VBUS current			3 A

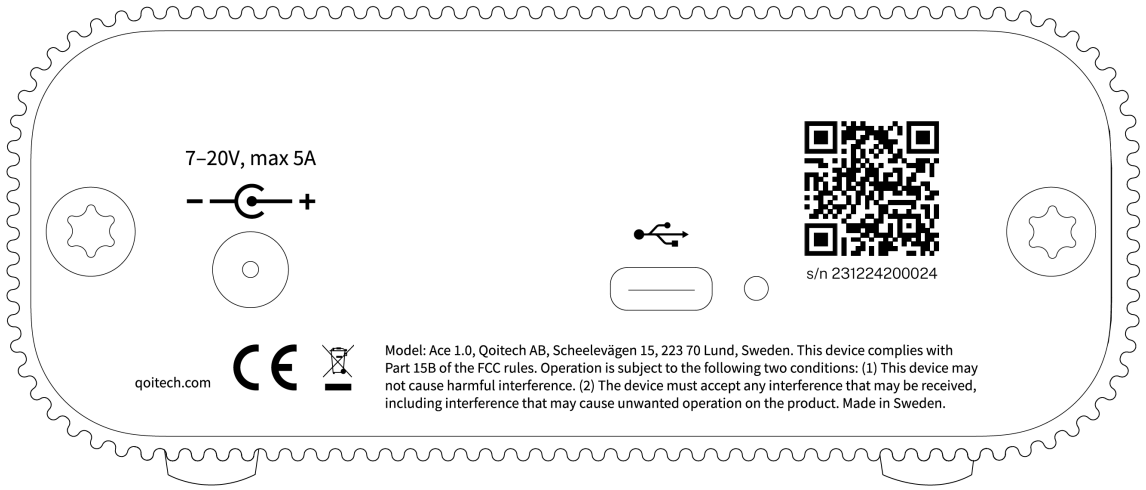
5 Otii Ace Pro

5.1 Hardware overview

The front side of the **Otii Ace Pro** has the main connectors, additional ports to extend measurement capabilities, plus a status LED.

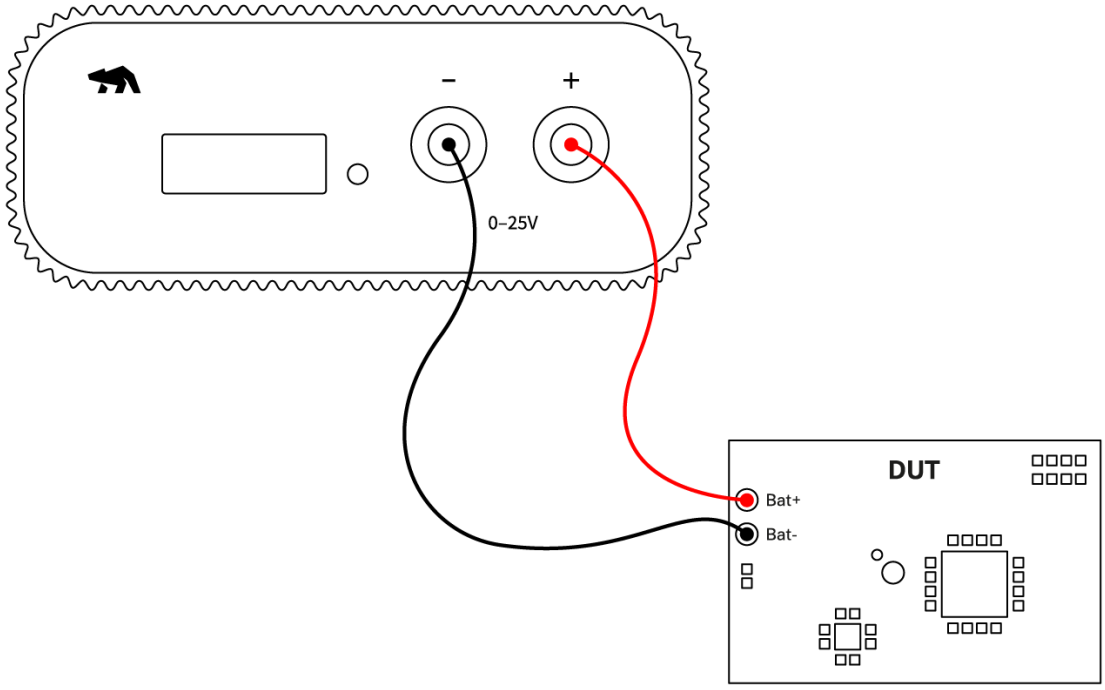


The back side of the **Otii Ace Pro** has the host USB-C connector and an input socket to power up the unit with an external DC adapter when required.

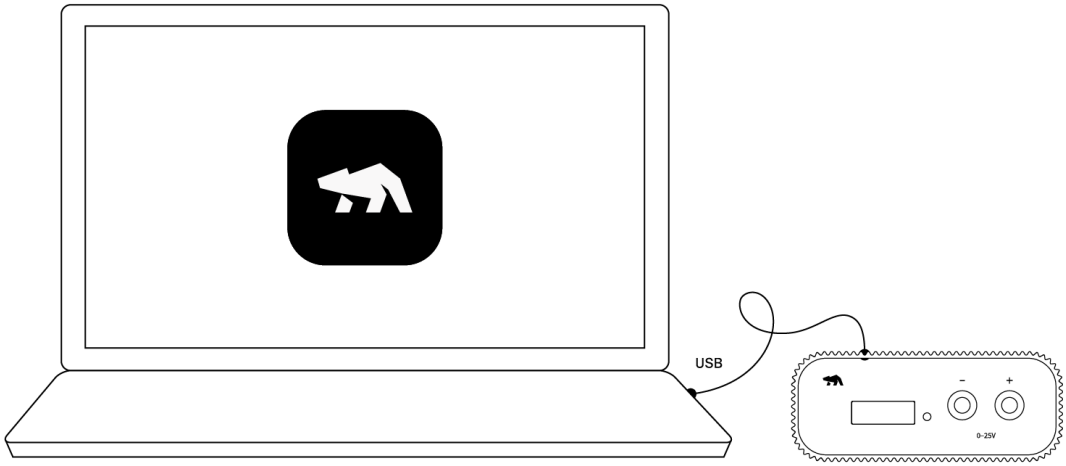


5.2 Main connections

The **Otii Ace Pro** has the voltage(+) and voltage(-) binding post (red and black connectors located on the right side of the instrument), which are used to source the DUT but also to measure the current, voltage, and power of the device. The DUT can be connected through its battery connectors or DC input; refer to the connection diagram below for better understanding.



To power up the **Otii Ace Pro** and enable the communication with the host computer, use the provided USB C cable to connect the unit to your computer.



Check out the Wiring up page to explore all possible connection configurations between the Otii Ace Pro and your DUT.

5.3 Expansion Port

The expansion port on the front of the **Otii Ace Pro** is designed to enhance the hardware's functionality, allowing users to measure additional voltages and currents or trigger external events. These additional pins feature multi-input and multi-output for analog and digital signals, serial

communication, and additional power and ground points. Below is a general overview of the additional pins available:

Connector	Description	Maximum rating
0-15V	Adjustable output between 0V and 15V	600 mA
DGND	Digital ground (return current)	
RX	UART RX / Digital Input (Exclusively under Scripting)	0 V to 5.5 V
TX	UART TX / Digital Output (Exclusively under Scripting)	0 V to 5.5 V
GPI1	Digital input 1	0 V to 5.5 V
GPI2	Digital input 2	0 V to 5.5 V
GPO1	Digital Output 1	0 V to 5.5 V
GPO2	Digital Output 2	0 V to 5.5 V
DGND	Digital ground (return current)	
AGND	Analog ground (return current) / Analog measurements	
ADC+	Differential input for current measurement. It also measures single ended voltage with respect to AGND	-10 V to 25 V
ADC-	Input for current measurement	-10 V to 25 V
SENSE+	Analog input with the AGND as reference/return	-10 V to 25 V
SENSE-	Analog input with the AGND as reference/return	-10 V to 25 V

5.4 Status LED

The table below explains the **Otii Ace Pro**'s status LED behaviors and their meanings to help you quickly identify the device's state.

LED behavior	Otii Ace Pro Status
Solid blue	Otii Ace Pro is active
Solid green	Relay closed, sinking/sourcing power or in-line mode
Flashing white	Calibrating
Flashing red/blue	Upgrading firmware

5.5 Ace Tech spec

5.5.1 General

Operating environment	Min	Typical	Max
Temperature	10°C / 50°F		30°C / 86°F
Humidity	30 %		60 %

5.5.2 Main

Power supply	Min	Typical	Max
Output voltage	0 V		25 V
Output voltage setting resolution		1 mV	
Self-consumption		3.5 W	
Output power, max continuous		30 W [18] [19]	
Output power, max peak		50 W [20]	
Voltage between USB/DC jack and Main –	-200 V		-200 V
Voltage between DGND/AGND and Main –	-200 V		-200 V

Programmable current sink (requires an Otii Toolbox)	Min	Typical	Max
Sink current	0 A		5 A
Sink current, setting resolution		1 µA	
Sink voltage	0 V		25 V
Sink power, max continuous		15 W	
Sink power, max peak		125 W	

Current measurement	Min	Typical	Max
Accuracy -5A to 5 A		±(0.05 % + 25 nA)	
Resolution		0.4 nA	
Sample rate		50 ksps	
Analog bandwidth (3 dB)		50 kHz	

Voltage measurement	Min	Typical	Max
Accuracy		±(0.01 % + 1 mV)	
Sample rate		50 ksps	
Analog bandwidth (3 dB)		50 kHz	
Output voltage readback resolution		3.5 µV	

5.5.3 Expansion port

UART	Min	Typical	Max
Bitrate	50 bps		5.25 Mbps

Digital I/O	Min	Typical	Max
Digital IO operating voltage	1.2 V	Vio [21]	5 V
VIL Low-level input voltage			Vio * 0.2 V
VIH High-level input voltage	Vio * 0.8 V		

[18] Depends on available input power
[19] Max 3A in on DC plug-in and max 4A output current
[20] Depends on available input power
[21] Otii Battery Toolbox emulates the total ESR.

Digital I/O	Min	Typical	Max
Imax, Max sink/source current (total for GPIOs)	-10 mA		10 mA

Differential ADC, pins ADC- and ADC+	Min	Typical	Max
Voltage input	-10 V		25 V
Shunt voltage range	-102.4 mV		102.4 mV
Resolution		12.2 nV	
Accuracy		±(0.1 % + 1 µV)	
Input impedance		> 100 Mohm	

Single ended ADC, pins ADC- and ADC+	Min	Typical	Max
Voltage input	-10 V		25 V
Resolution		3.1 µV	
Accuracy		±(0.1 % + 250 µV)	
Input impedance		> 100 Mohm	

SENSE, pins SENSE- and SENSE+	Min	Typical	Max
Voltage input	-10 V		25 V
Resolution		3.1 µV	
Accuracy		±(0.1 % + 250 µV)	
Input impedance		> 100 Mohm	

Expansion port power supply	Min	Typical	Max
Output voltage	0 V		15 V
Output voltage setting resolution		5 mV	
Output current			600 mA
Voltage between USB/DC jack & DGND/AGND	-200 V [22] [23]		-200 V [24] [25]

5.5.4 USB and DC jack

DC jack	Min	Typical	Max
Input voltage	7 V		20 V
Input current			5 A

[22] USB and DC jack GND is connected internally to chassis GND
[23] DGND and AGND are internally connected
[24] USB and DC jack GND is connected internally to chassis GND
[25] DGND and AGND are internally connected

USB	Min	Typical	Max
VBUS voltage [26]	4.75 V		20 V
VBUS current [27]			3 A

[26] USB PD 2.0
[27] USB PD 2.0

6 Projects

Otii 3 Desktop App works on a project basis, which can be saved and later opened again.

An Otii project contains all the settings of the device under test (DUT) used, plus all measured measurement recordings of the multiple channels available for current, voltage, and power data.

6.1 Project storage

Otii 3 creates a lot of sample data when doing lengthy recordings.

This makes it important to choose the right storage location for your projects:

- Only store the project on a fast locally connected drive.
- When creating a new project, it is recommended to save it before starting a lengthy recording. This is because an unsaved new project is stored in the temporary directory of the OS.
- Do not store the project in a folder that is synced to a cloud service like OneDrive, Dropbox or Google Drive. On many installations the Documents and Desktop folders are cloud synced.
- Do not store the project on a network connected drive (NAS).

The size of the data for a one channel in one recording is:

```
4 bytes * sample_rate * no_of_seconds
```

The default sample rate of the **Otii Arc** is 4 ksps, and for the **Otii Ace** 50 ksps.

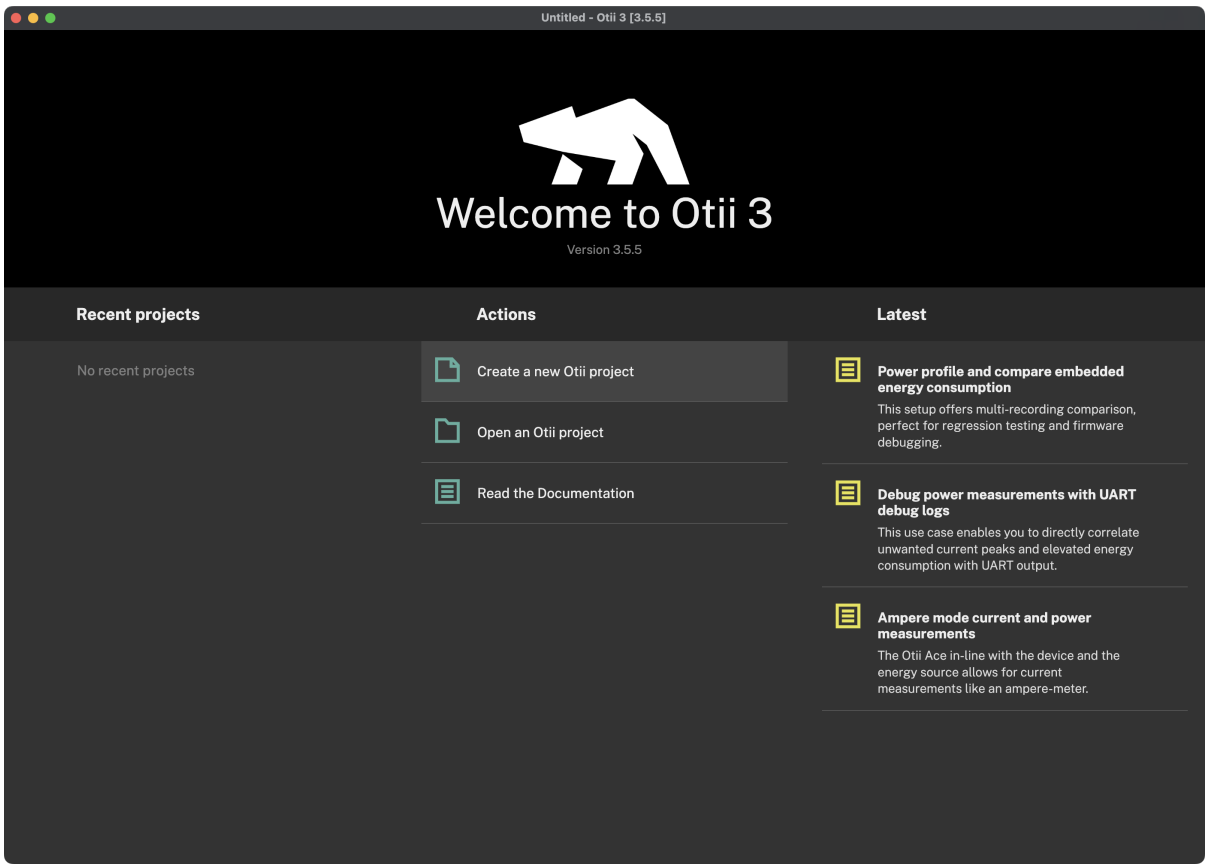
If you have an **Otii Ace** you can change the sample rate down to 1 ksps.

6.2 Create a new project

When opening the Otii 3 Desktop App, users can create a new project or choose to do so while working on an existing one.

6.2.1 Create a new project when starting the Otii 3 Desktop App

1. Open Otii 3 Desktop App.
2. On the welcome window, below the Actions section, click *Create a new Otii project*.



6.2.2 Create a new project while working on a project

1. Navigate to:

- **Windows & Ubuntu**: File > New, or by pressing `Ctrl-N`
- **MacOS**: File > New, or by pressing `⌘-N`

6.3 Save project

Users can save Otii projects in a desired folder and/or export them as a zip archive.

6.3.1 Save .otii3 projects

Projects in Otii3 can be saved with “Save” or “Save as”. When the “Save” option is selected for the first time in a new project, it will function as “Save as,” allowing the project to be saved in the desired location and with the assigned name. Later, the “Save” option will only update the project files previously saved.

6.3.1.1 Save projects using “Save”

1. Navigate to:

- **Windows & Ubuntu**: *File* > *Save*, or by pressing `Ctrl-S`
- **macOS**: *File* > *Save*, or by pressing `⌘-S`

2. Set a name for the Otii project, and click *Save*

6.3.1.2 Save projects using “Save as”

1. Navigate to:

- **Windows & Ubuntu**: File > *Save as...*, or by pressing `Shift-Ctrl-S`
- **macOS**: *File* > *Save as...*, or by pressing Shift-⌘-S.

2. Set a name for the Otii project, choose the folder to save the project, and click *Save*

6.3.2 Export a project as a zip

Sharing Otii projects is easy by exporting them as a zip file, which can be directly opened in the Otii 3 Desktop App, whether or not the user owns an Otii Arc Pro or Otii Ace Pro.

1. Navigate to:

- **Windows & Ubuntu**: *File* > *Export project...*, or by pressing `Ctrl-E`
- **macOS**: *File* > *Export project...*, or by pressing `⌘-E`

2. Set a name for the Otii project, choose the folder to save the project, and click *Save*

ⓘ Note that the selected folder cannot be the same folder as the project is in.

6.4 Open project

Otii 3 Desktop App allows users to open a saved project either when opening the application or while working on a project. These projects may be located in a folder or a .zip file previously saved or exported.

When the Otii project is opened, it contains all previous settings such as battery emulator, profiler, analytics configurations, and measurement recordings. This enables users to perform new measurements within the same project.

6.4.1 Open a saved project when starting the Otii 3 Desktop App

1. Open **Otii 3 Desktop App**
2. On the welcome window, below the Actions section, click *Open an Otii project*
3. Locate the Otii project file and open it

6.4.2 Open a saved project while working on a project

1. Navigate to:

- **Windows & Ubuntu**: *File* > *Open*, or by pressing `Ctrl-O`
- **macOS**: *File* > *Open*, or by pressing `⌘-O`

2. Locate the Otii project file and open it.

6.5 Import an Otii 2.0 project

ⓘ If you have been using the Otii Product Suite for some time, you may be familiar with Otii 2. We highly recommend that you download the latest version of the Otii 3 Desktop App to take advantage of its improved features and extended capabilities for testing. Read more: “Updated license model and Otii software release”

An Otii 2 project can be imported within the Otii 3 Desktop App. Just be aware that once the project is saved in Otii 3 version, it is not possible to open the saved project again in Otii 2.

6.5.1 Open an Otii 2 project when starting the Otii 3 Desktop App

1. Open **Otii 3 Desktop App**
2. On the welcome window, below the *Actions* section, click *Open an Otii project*
3. Locate the Otii 2 project file and open it.

6.5.2 Open an Otii 2 project while working on a project

1. Navigate to:

- **Windows & Ubuntu**: *File* > *Open*, or by pressing `Ctrl-O`
- **macOS**: *File* > *Open*, or by pressing `⌘-O`

2. Locate the Otii 2 project file and open it.

6.6 Export recordings as CSV

Exporting recordings in CSV format broadens compatibility with third-party data analysis tools when making a more precise analysis of the collected recordings. A practical alternative to generate graphs for dedicated reports or analyze data further.

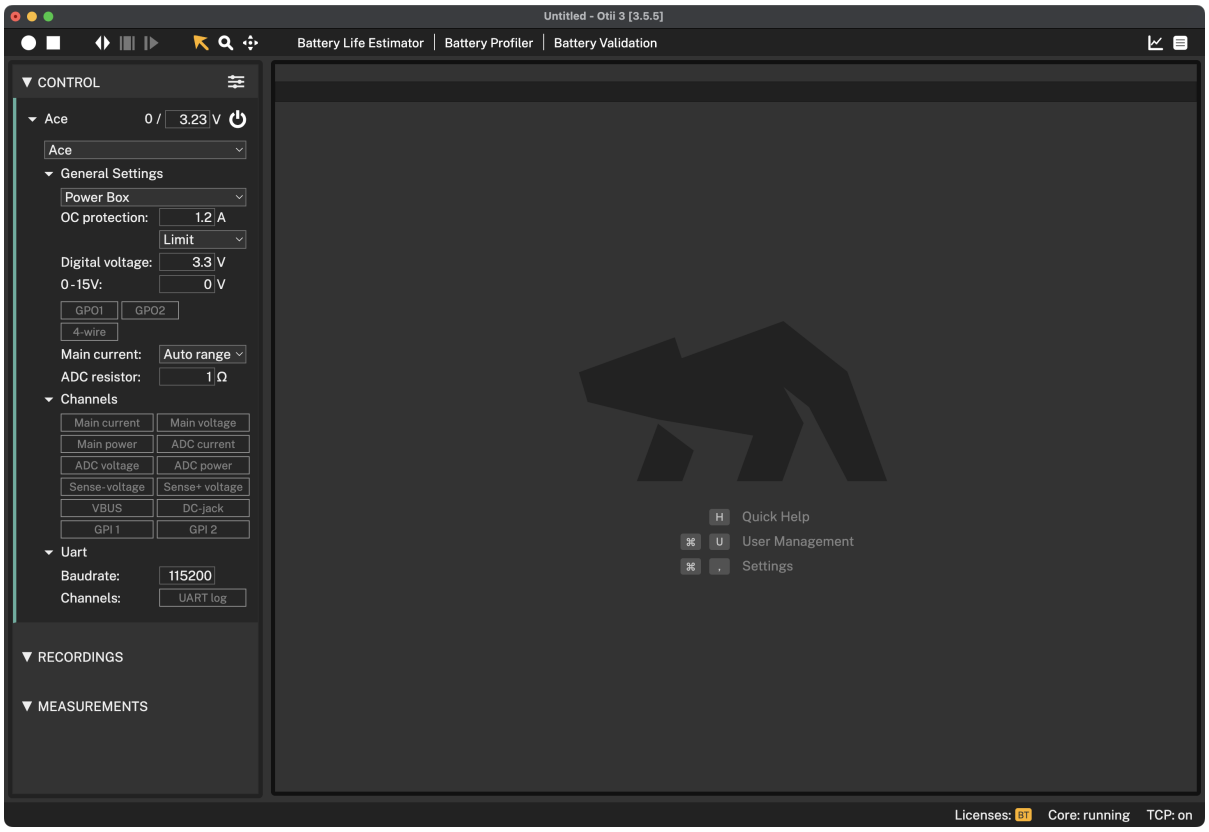
6.6.1 Export measurement recordings in CSV

1. Open **Otii 3 Desktop App**
2. Create a new Otii project or open a previously saved project. In case of creating a new project, at least one recording must be recorded to be exported
3. In the left sidebar, navigate to the *RECORDINGS* section, right-click on the desired recording, and select the *Export to CSV...* option
4. Assign the desired name for the file, and click *Save*

7 Control

The **Otii 3 Desktop App** has a control panel on the left sidebar that facilitates the connection between the **Otii hardware** and the application, assuming it is correctly connected to the computer.

The *CONTROL* section enables the **Otii hardware** to be set up according to the technical parameters of the devices under test (DUT), the mode of use of the **Otii hardware**, and the channels to be measured. These configurations are independently set for each connected **Otii hardware**, regardless of whether it is an **Otii Arc** or **Otii Ace**.

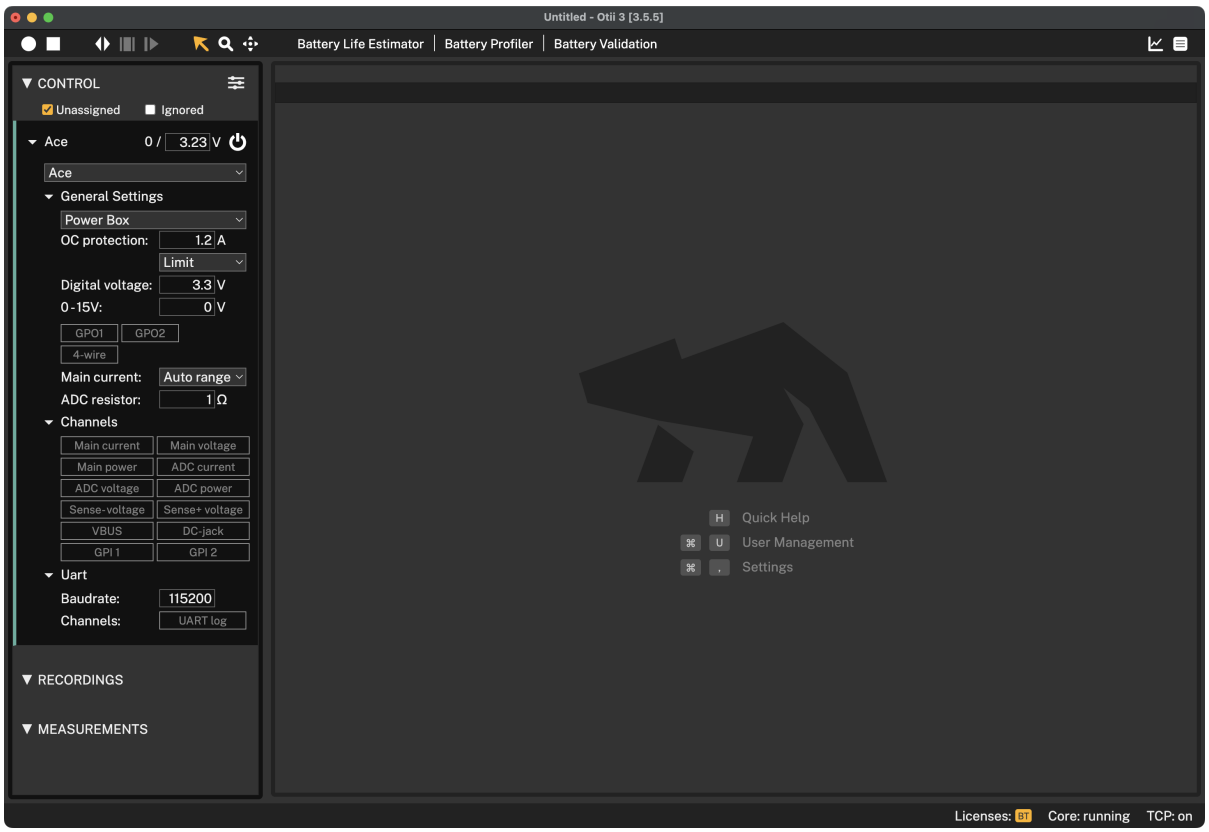


As shown in the image above, the *CONTROL* panel features the following:

- Arc/Ace: Define power supply voltage, and it turns ON/OFF
- General settings: Set the operation mode for the Otii hardware and define the settings in which it will operate based on the DUT's characteristics
- Channels: Channels available for measurement by the Otii hardware
- Uart: Enable and set up serial communication with the DUT for debugging

7.1 Control filter

To filter what is being shown in the control list, click the filter button just next to the control title:

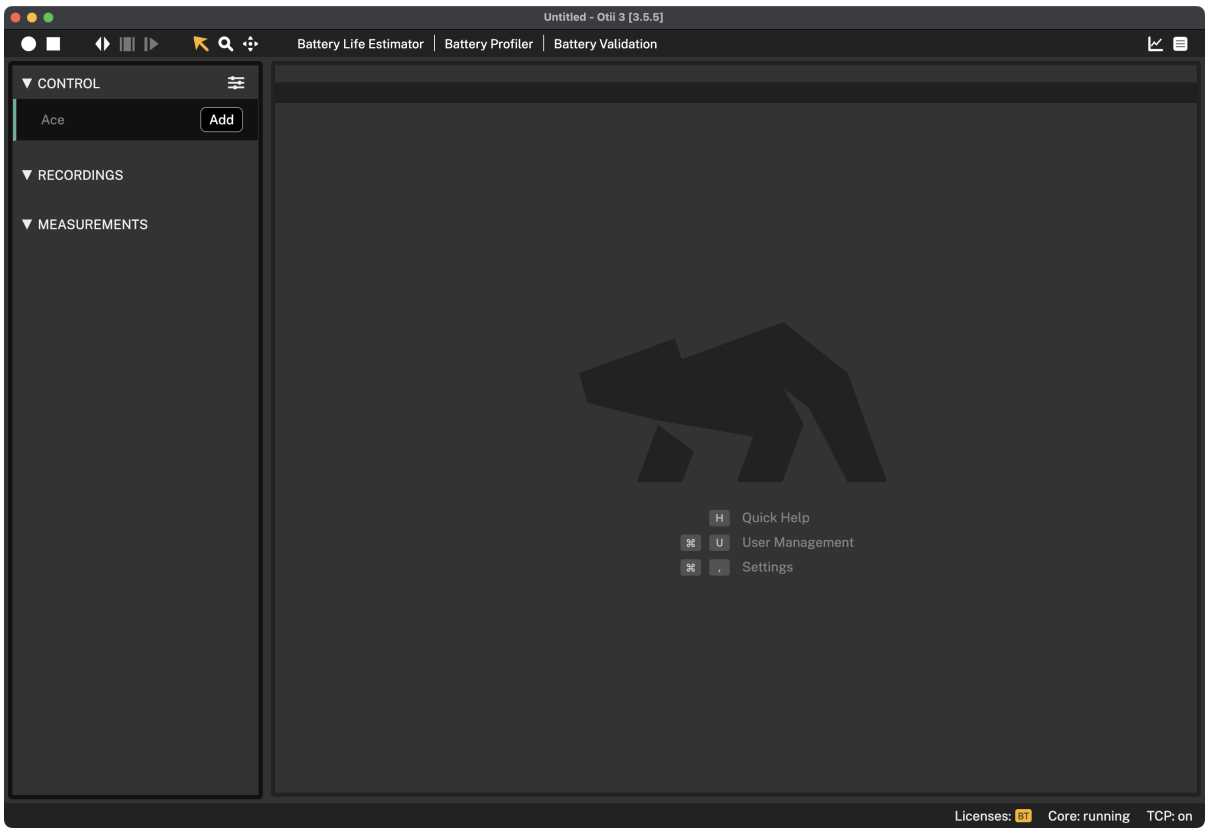


Where,

- **Unassigned**: When enabled, all devices, including those that are not added to the project, will be shown
- **Ignored**: When enabled, it will show ignored devices. UART devices can be ignored by right-clicking it and choosing *Ignore*

7.2 Otii Arc and Ace setup

After connecting the **Otii hardware** (either the **Otii Arc** or **Otii Ace**) to the computer is automatically recognized within the **Otii 3 Desktop App**. To add it, click the *Add* button.



By default, once the project is saved, the **Otii hardware** will be added to it each time it is opened. Once added, users can set up and/or customize the following parameters:

7.2.1 Name

By default, the name given to the device is the version of the **Otii hardware** used. To change it, double-click and assign the desired name, or right-click, select *Rename*, and assign the new name.

7.2.2 Main voltage

Just right next to the Otii hardware name, the measured voltage followed by the desired voltage will be found. Being the desired voltage, the one supplied through the **Otii hardware** main terminals when the unit is set in *Power Box* mode.

By default, the voltage assigned to the Otii hardware, either **Otii Arc Pro** or **Otii Arce Pro**, is 3.0 V. To change it, click on the value and enter the new desired voltage. Keep in mind that the Otii hardware's main output range for the Otii Arc Pro is 0.0 V — 5.0 V, and for the **Otii Ace Pro** is 0.0V — 25.0V; however, the system restricts the max output to 0.0V — 5.5V. When using **Otii Ace Pro**, the max voltage can go up to 25.0V. To change it, go to *Settings* > *Max supply voltage* and set the desired maximum voltage.

ⓘ The maximum voltage that Otii Arc Pro can supply depends on whether a power supply via the DC plug is connected to the Otii hardware and whether activating the "Auto range" option is

selected as the main current parameter. The maximum value will be automatically updated based on the current configuration.

⚠ Make sure to cap the max voltage based on the DUT's technical requirements to avoid accidental high voltages.

7.2.3 Power button

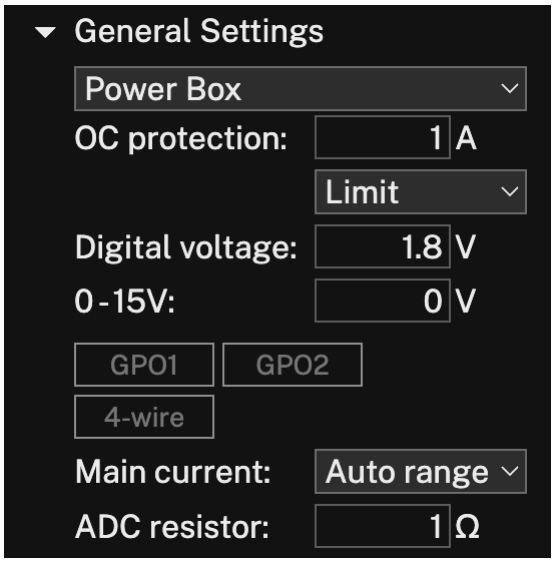
Click the power button to turn the main power supply ON/OFF.

7.2.4 Device attachment

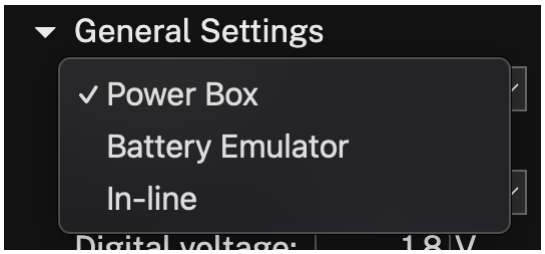
The drop-down box below the **Otii hardware** name enables the control settings to be reassigned to another **Otii hardware** of the same version. To do so, select and reassign the other **Otii hardware**(s) within the list.

⚠ An Otii Arc Pro cannot be reassigned with an Otii Ace Pro or vice versa.

7.3 General settings



7.3.1 Supply mode



- **Power Box**: The Otii hardware will supply a constant voltage to the Device Under Test (DUT)

- **Battery emulator**: Otii hardware will emulate a battery. (Only available with Battery Toolbox)
- **In-line**: This particular mode is exclusive to the Otii Ace Pro version. It works like an ampere meter by being in-line between the power source and the DUT. The Otii Ace Pro automatically switches between different measurement ranges but provides no power. To also measure voltage, 4-wire must be enabled

7.3.2 Over current protection

The OC current protection setting allows users to set a maximum current limit that the **Otii hardware** can supply. For the **Otii Arc**, if the current exceeds the assigned limit, the unit will switch OFF the main supply to protect the DUT.

Within the dropdown, choose among the available options how the current limits must be detected, along with expected device behavior:

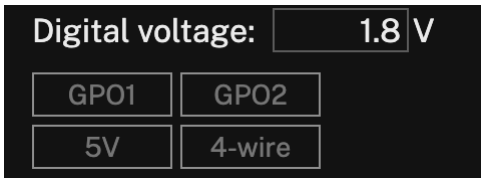
- **Limit**: enabled for **Otii Arc Pro** and **Otii Ace Pro**, it limits the output current to the set current, lowering the output voltage
- **Cut-off**: enabled only for the **Otii Arc Pro**, it detects when the current exceeds the set current to turn the power OFF and open the output relay.

7.3.3 Digital voltage

The digital voltage setting allows the output voltage desired for the pins available within the expansion port to be set up. Also, depending on the digital voltage setting, the level for deciding if it is a logical 0 or 1 is also changed.

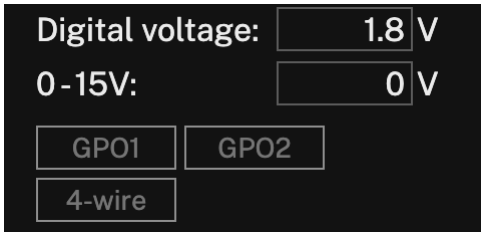
Some configurations in this section differ based on the version of **Otii hardware** used.

7.3.3.1 Otii Arc Pro



- **GPO1 & GPO2** - enable digital outputs
- **5 V** - enables an additional fixed 5 V output
- **4-wire** - enables 4-wire measurements to compensate for voltage drop in the power supply leads

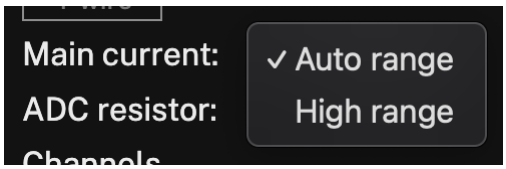
7.3.3.2 Otii Ace Pro



- **0 - 15 V** - Enables an additional voltage output ranging from 0 to 15 V
- **GPO1 & GPO2** - enable digital outputs
- **4-wire** - enables 4-wire measurements to compensate for voltage drop in the power supply leads

7.3.4 Main current

The main current enables the Otii hardware to switch automatically between the different measurement intervals.



- **Auto range**: when selected, the highest available measurement accuracy interval is selected
- **High range**: when selected, it prevents the **Otii hardware** from going to the lowest measurement range Therefore, there is no switch in or out of the shunt resistor
  - Use this setting if the device resets when the **Otii hardware switches measurement interval
  - Use this setting if higher voltage output is needed for **Otii Arc Pro**, as the voltage drop internally within this Otii hardware version is lower

ⓘ Otii Arc cannot supply above 3.75V in USB mode and 4.55V when supplied with DC plug when selecting auto range. In high range, it can supply 4.2V in USB mode and 5.0V when supplied with DC plug. Note, Otii Ace can supply the full voltage range both in USB mode and when supplied with DC plug.

7.3.5 ADC resistor

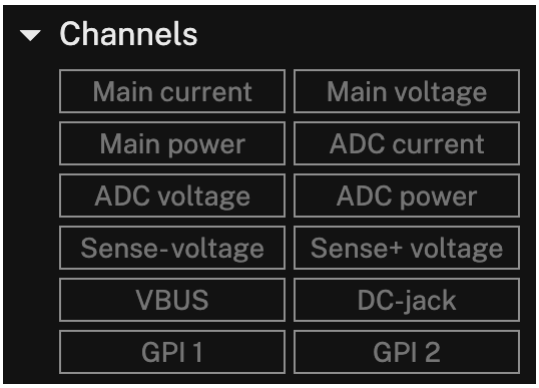
When using the ADC pins to measure the current of a specific system, an external shunt resistor is required. Therefore, the resistance must be set. To ensure that the resistor impedance is set correctly, refer to the Measuring a subsystem example; this example will guide you through connecting the DUT to the ADC connectors and setting the impedance of the resistor.

7.4 Channels

Channels in the **Otii 3 Desktop App** are different input/output data sources that can be monitored or controlled. These channels enable the measurement of multiple currents, voltages, and power consumption of the main system and sub-systems.

To enable a channel, simply click on the desired channel to be measured. Once enabled, the graphing section will display where the measured values will be logged for further analysis. If a channel is disabled for a new recording within the same project, the associated data will be kept, although the channel will not be included in upcoming recordings.

When selecting a current channel, the matching power channel will automatically be enabled as well, but the power measurement will be hidden by default. The power channel is needed to calculate the energy consumed in the statistics for the recordings.



Within the channels available for measurement, the following can be found:

- **Main current**: include a graph of the current measured at the Otii hardware main connectors in the recordings
- **Main power**: include a graph of the power measured at the Otii hardware main connectors in the recordings
- **Main voltage**: include a graph of the voltage measured at the Otii hardware main connectors in the recordings
- **ADC voltage**: include a graph of the voltage measured at the ADC connectors in the recordings. This is measured on ADC+ pin. Single-ended from ADC+ to AGND; see an example of how to connect the ADC to measure voltage
- **ADC current**: include a graph of the current measured at the ADC connectors in the recordings. This is a differential measurement between ADC+ and ADC- pin; see an example of how to set the ADC resistor
- **ADC power**: include a graph of the power measured at the ADC connectors in the recordings
- **SENSE +/- voltage**: include a graph of the voltage measured at the SENSE+ and/or SENSE- pins in the expansion port. Single-ended measured compared to AGND
- **VBUS**: include a graph of the VBUS voltage
- **DC-jack**: include a graph of the DC-jack voltage

- **GPI1 & GPI2**: include a graph of the signal measured at the GPI1 and/or GPI2 pins in the expansion port. These pins are digital, and their value depends on how you have configured your digital voltage level in the supply mode

7.5 UART

UART (Universal Asynchronous Receiver/Transmitter) is a device-to-device communication protocol that allows interface and talk to exchange among devices. Within the **Otii 3 Desktop App**, the UART settings allow engineers and developers to:

- monitor and analyze data transmitted directly from the device under test (DUT) to the **Otii 3 Desktop App**
- or transmit data directly from the **Otii 3 Desktop App** to the DUT; for example, send AT commands to the DUT while simultaneously observing the behavior and response via the console log.

By capturing and analyzing the UART data displayed within the Otii 3 console log, developers can gain insights into the DUT's behavior and precisely identify and troubleshoot any issues that might be causing the DUT's energy source-drain.

You can either use the built in UART of your **Otii hardware**, or you can use use a third party USB UART.

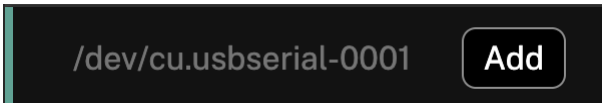
7.5.1 Enable the UART of your Otii hardware

1 While configuring the DUT in the *CONTROL* section of the **Otii 3 Desktop App**, navigate to the last setting option: Uart. 2 Set the baud rate on the respective field. To guarantee serial communication, assign the same baud rate as the one assigned in the DUT. 3 Enable the serial communication in the Channels option by clicking the *UART log* button. Upon enabling it, the console log will be automatically displayed at the bottom section of the Otii 3 Desktop App.

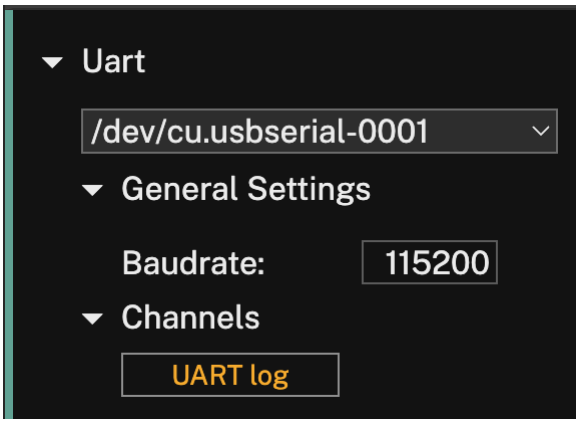
Once activated, with the additional connection required to establish serial communication between the DUT and the Otii hardware, the data transmitted will be displayed on the console log after the power supply is switched ON and the recording starts.

7.5.2 Enable an USB UART

If you have a third party USB UART connected to your computer it will be shown below the **Otii hardware** with the name of the computer device it is connected to. To add it, click the *Add* button.



Set the baudrate of the device and click in *UART log* to enable it for recording.



7.6 Firmware management

To ensure the proper operation of the **Otii hardware** with the latest version of the Otii 3 Desktop App, it is important to keep the hardware up to date with the latest firmware version available.

Once the **Otii 3 Desktop App** detects a new firmware version for the connected Otii hardware, a yellow exclamation mark icon shown in the *CONTROL* section indicates a new firmware is available. To upgrade it, you can:

- click on the exclamation mark icon, and choose *Upgrade firmware*,
- or, navigate to *Devices > Upgrade firmware to latest*.

ⓘ Note that the upgrade firmware option is only available if the connected Otii hardware requires an upgrade. Otherwise, it will be visible but disabled.

Once the firmware upgrade starts, the following behavior will be observed on the Otii hardware:

- **Otii Ace Pro**: the LED will start blinking red and blue quickly, then turn off for a few seconds, turn on yellow, and stay blue — the same state it was before the firmware upgrade. Then, it will automatically reconnect to the **Otii 3 Desktop App**
- **Otii Arc Pro**: It will restart, and when the upgrade is finished, it will automatically reconnect to the **Otii 3 Desktop App**

7.6.1 Upgrade firmware from file...

In case the Otii hardware needs to be upgraded to another firmware version, for testing and/or compatibility with particular features, it can be upgraded by:

1. Navigate to Devices > Upgrade firmware from file.
2. Choose the firmware file shared by the Qoitech team, and click open.
3. Once opened, the firmware is upgraded in a few seconds, following the same behavior previously described on this page.

7.7 Calibration

To guarantee accurate measurements, the Otii hardware, both the **Otii Arc Pro** and the **Otii Ace Pro** can be calibrated when required. It is recommended to calibrate before any measurement requiring the best accuracy, especially ultra-low currents.

7.7.1 Calibrate a single connected Otii Arc Pro/Otii Ace Pro

1. Open the **Otii 3 Desktop App**, and open/create a project.
2. In the **Otii 3 Desktop App** toolbar, navigate to *Device > Calibrate*.

Once the calibration starts, the following behavior will be observed on the Otii hardware:

- **Otii Ace Pro**: the LED will start blinking white slowly, then turn off for a few seconds and turn on blue - the same state it was before the calibration.
- **Otii Arc Pro**: It will restart when the calibration is finished.

7.7.2 Calibrate all the Otii hardware connected

1. Open the **Otii 3 Desktop App**, and open/create a project.
2. In the **Otii 3 Desktop App** toolbar, navigate to *Device > Calibrate all devices*

Once calibration starts, the same behavior on the **Otii hardware** previously described on this page can be observed.

8 Recordings

Measurements acquired within the **Otii 3 Desktop App** are represented as recordings. The measurements can be either single or multiple recordings, and are intended to compare different testing scenarios across the hardware development cycle.

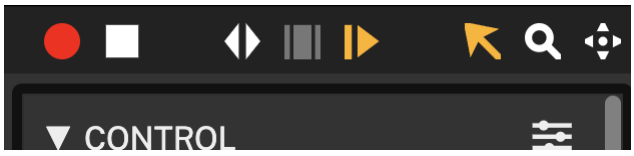
8.1 Start recording

Before starting a new recording, make sure to:

1. Have an **Otii hardware** connected and added to the **Otii 3 Desktop App**.
2. Select the channel(s) to be measured.

ⓘ If some of the points mentioned above are missing, the notification center notifies the user that these must be done to start the recording.

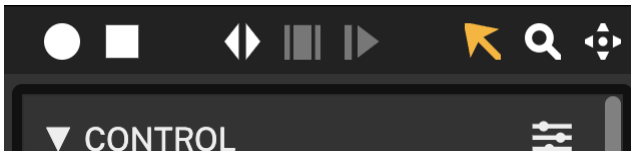
To start recording, click the record button at the top left of the Otii 3 Desktop App. Once clicked, it will change to red, informing the recording is active.



If multiple recordings are required, just one after another, click the record button while it is active, and a new recording will start automatically.

8.2 Stop recording

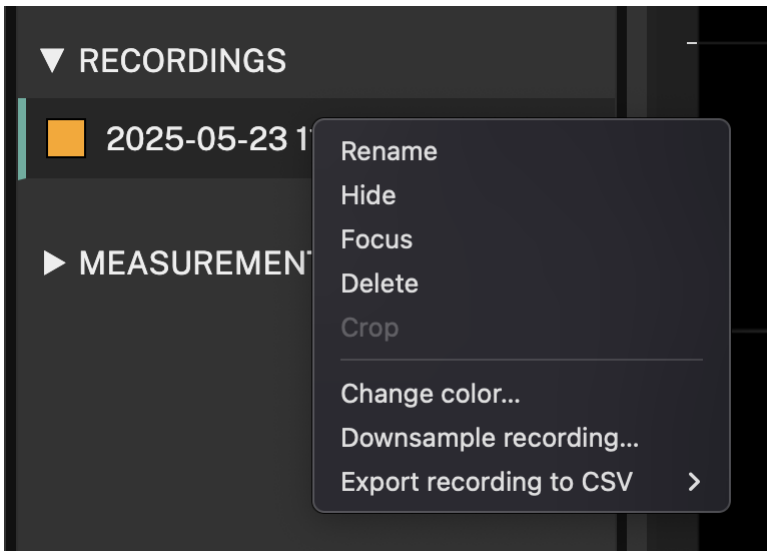
To stop recording, click the stop button located right next to the record button.



ⓘ It is not required to stop the recording to start a new one. Simply start a new recording by clicking the record button while it is active, as described in the previous step.

8.3 Recordings context menu

Under the *RECORDINGS* section, found on the left sidebar of the **Otii 3 Desktop App**, each of the recordings saved is listed separately. By right-clicking on a recording, a menu is displayed, enabling multiple options to further customize or explore the recording, among them:



8.3.1 Rename recording

Right-click on the recording, select the *Rename* option, and assign the new name. Alternatively, double-click on the recording's name to rename it.

8.3.2 Hide/Show recording

Right-click on the recording and select the *Hide* option. Once hidden, the recording is still shown in the saved recordings but will be hidden from the graphs. To display the graph recording again, right-click on the recording and select the *Show* option.

Alternatively, click on the colored box beside the recording name. Once clicked, the box becomes uncolored, indicating that the graph is hidden. By clicking again, it will be recolored, indicating that the graph is being shown.

8.3.3 Focus/Unfocus recording

This option allows the focus of the graph visualization on a single recording, while the other recordings are grayed out in the background instead of the color assigned to the recordings.

To focus on a specific recording, right-click on the recording and select the *Focus* option. To unfocus it, follow the same instructions but this time, click on the *Unfocus* option.

Once the focus option has been enabled on a recording, it can be switched between the different recordings by simply clicking on the desired recording to be focused on.

8.3.4 Delete recording

Right-click on the recording desired to be deleted and select the *Delete* option.

Alternatively, click on the recording desired to be deleted and press the delete key.

8.3.5 Crop recording

By default, the crop option is disabled, as a graph segment must be selected to enable cropping.

On the graph visualization section, select the graph segment to keep by clicking on the starting point, and holding the click until the desired end point for the graph to be cropped. After selecting the graph segment, right-click on the recording desired and select the *Crop* option.

8.3.6 Change color

To change the default color assigned to the recording graph, right-click on the recording and select the *Change color...* option. Once selected, a window with multiple colors to choose open. Choose the desired color, which will automatically apply to the graph. Click the *Close* button to close the window.

8.3.7 Downsample recording

In case of requiring a lower sample rate when analyzing the recordings, it can be modified it by right-clicking on the desired recording and selecting the *Downsample recording...* option. Once selected, a window will open listing all the channels enabled for recording, where one or several at a time can be selected. Define the new sample rate, and to set it, just click the *Downsample* button.

⚠ Downsampling removes data points from the measurement and replaces them with average values, so unless you undo the downsample, you cannot increase the sample rate after it.

The sampling rate resolution while recording differs according to the version of **Otii hardware** in use. For **Otii Arc Pro**, the sample rate is up to 4 ksps and is not adjustable. For **Otii Ace Pro**, it is adjustable up to 50 ksps Note that the sample rate differs depending on the channel,

The sample rate differs depending on the channel. Check the following docs for further info:

Otii Arc Pro sample rate

Otii Ace Pro sample rate

8.3.8 Export to CSV

To export recordings as CSV, right-click on the desired recording and click the *Export to CSV...* option. Assign the desired name for the recording, and click *Save*.

8.4 Recording tools

8.4.1 Graph tools

The graph tools within the Otii Product Suite are designed to facilitate the analysis and optimization of multiple channels, such as voltage, current, and power.

8.4.1.1 Selection tool

With the selection tool selected, the mouse can be used to select a specific time span. The statistics will be updated to show the statistics for the selected area. You can drag the edges to change the selection size and drag the entire selection along the time axis by pressing the command simultaneously.

To activate the Selection tool, press the icon or press S on the keyboard to draw the selection in the graphs.

8.4.1.2 Zoom tool

With the zoom tool enabled, you can draw an area in a specific to zoom into.

To activate the Zoom tool, press the icon or press Z on the keyboard and use the mouse to click and draw a zoom area. The mouse wheel + `Ctrl` / ⌘ can also be used to zoom in and out at any time. By default, the mouse wheel zooms on the X-axis. Hold `Alt` to zoom the Y axis instead.

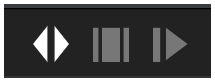
8.4.2 Drag tool

With the drag tool, you can drag a specific graph in any direction.

If you press `Alt` / ⌥ when the drag tool is enabled, only the currently active graph on the X-axis will be moved. This will offset the graph with respect to the other, which is very useful when comparing different recordings.

To activate the Drag tool, click on the icon located on the toolbar.

8.5 Plot tools



The plot tools enable the customization of the behavior of visualizations for further analisys. The tools available can be active, enabled, and disabled.

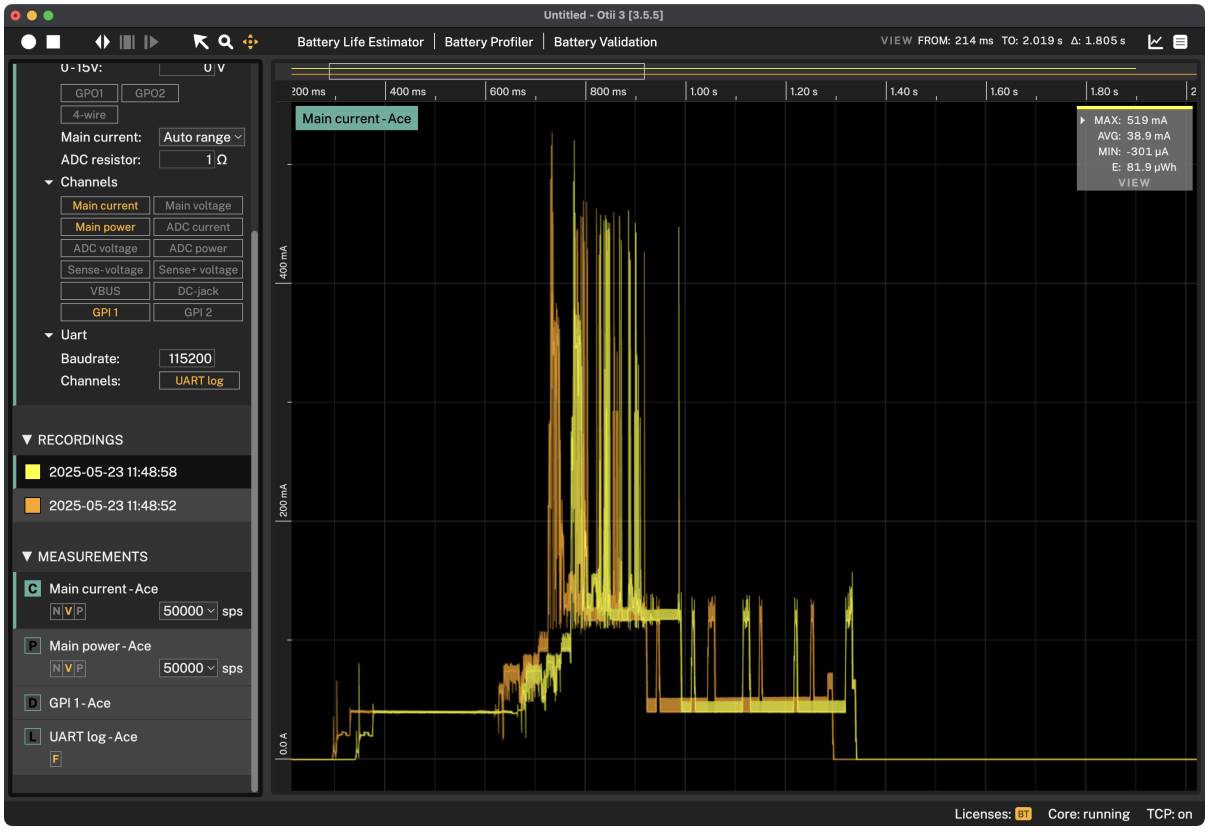
- **Fit width**: with this enabled, all the graphs will be zoomed out to show all the content
- **Zoom to selection**: if there is a selection, this makes the selection fill the entire view
- **Follow**: with this enabled, the tail of the current recording will be kept in the view

8.6 Recording viewers

Depending on the data type, the data is visualized in three different kind of viewers - analog, digital & log.

8.6.1 Analog

The analog viewer shows analog data recorded from the current, voltage and power channels. There is one analog view for each analog measurement, showing all the available recordings for each measurement in different colors.



8.6.1.1 Scroll bar At the top of the all viewers there is a scroll bar that show an overview of all the recordings in the project, and a white rectangle that represents the current view of the data.

You can scroll horizontally by dragging the white rectangle, and you can zoom horizontally by dragging the left and right border of the rectangle.

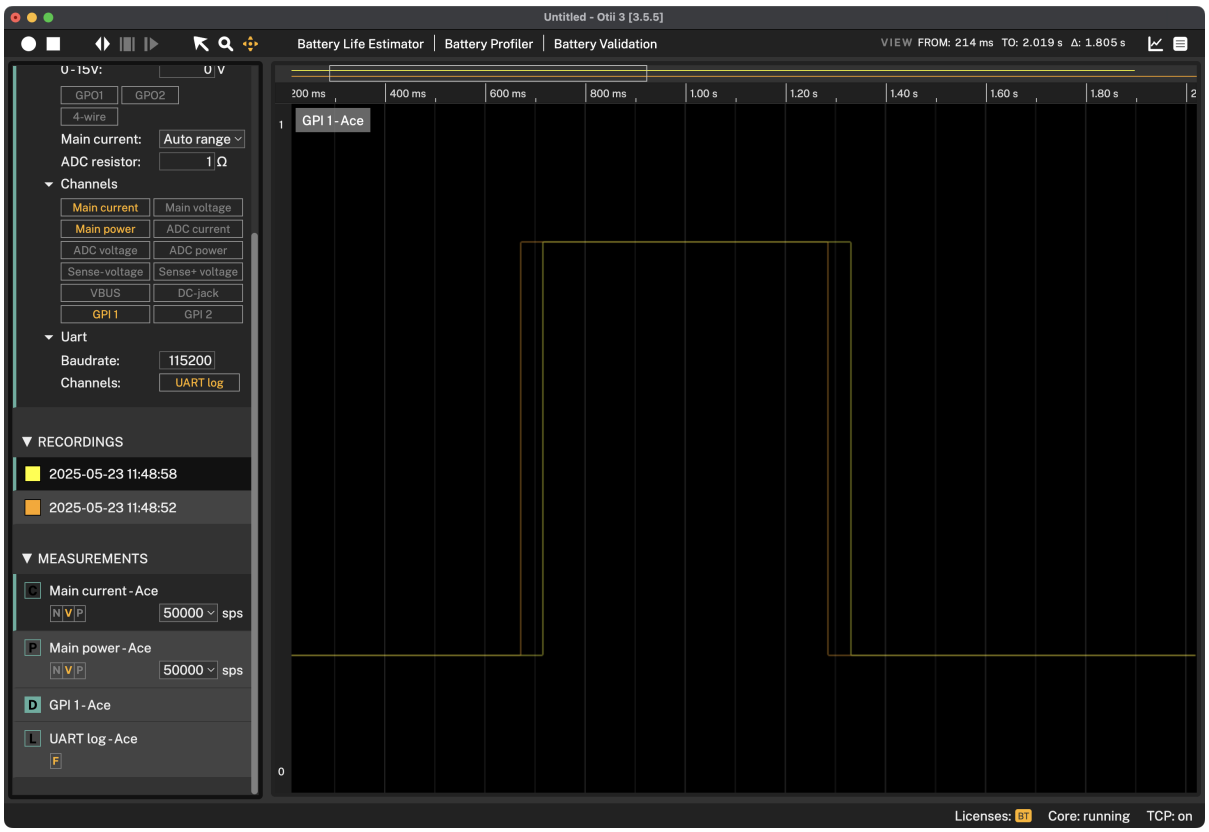
8.6.1.2 Graph Below the scroll bar is the graph showing part of the actual recordings. You can in addition to the scroll bar use the graph tools to navigate in the graph and to select data to analyse.

8.6.1.3 Context menu The context menu is displayed by right-clicking anywhere in the graph. The following options will be avaible the analog view:

- **Select all**: select all data
- **Clear selection**: clear the current selection
- **Zoom to selection**: when having a selection and right-click and select Zoom to selection, then the selection will fill the whole graph. This is the quickest way to zoom to the part that's desired to be analyzed
- **Set value range**: the time range to be evaluated can be adjusted from and to a specific time. This time will be the same for all the graphs shown
- **Set time range**: according to the channel graph selected, the time range to be evaluated can be adjusted from and to a specific time
- **Crop active recordings**: the selected recording will be cropped, keeping the selection, and all other recordings will remain unchanged. If the undo history hasn't cleared up, the crop can be undone by pressing `Ctrl-Z` / `⌘-Z`
- **Crop all recordings**: this will crop all the recordings, even the recordings that are not selected, keeping the data within the selection. If the undo history hasn't cleared up, the crop can be undone by pressing `Ctrl-Z` / `⌘-Z`.

8.6.2 Digital

The digital viewer shows digital data recorded from the GPI pins. There is one digital view for each digital measurement, showing all the available recordings for each measurement in different colors.



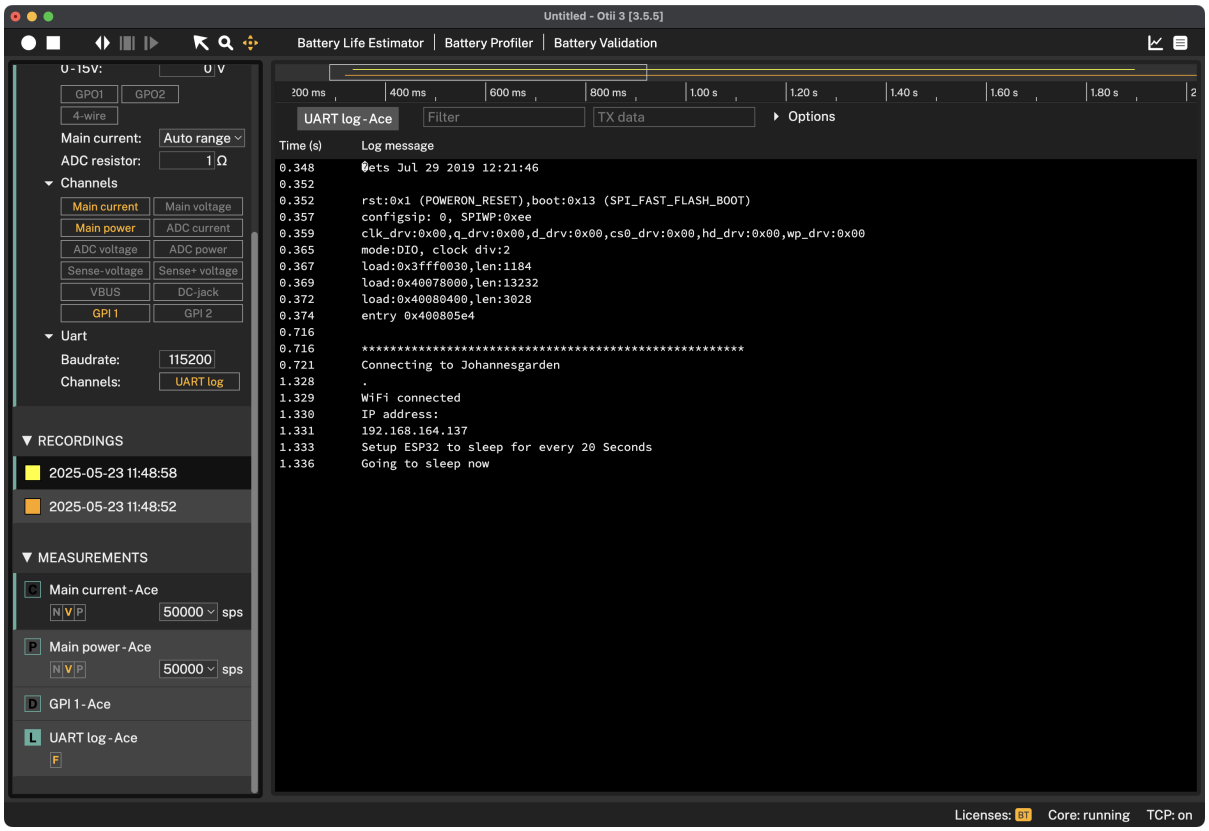
8.6.2.1 Graph You can use the graph tools to navigate in the graph and to select data to analyse.

8.6.2.2 Context menu The context menu is displayed by right-clicking anywhere in the graph. The following options will be avaible the digital view:

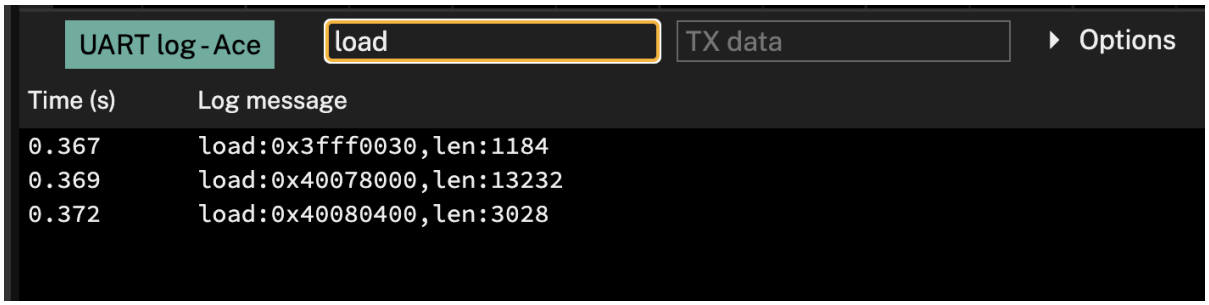
- **Select all**: select all data
- **Clear selection**: clear the current selection.
- **Zoom to selection**: when having a selection and right-click and select Zoom to selection, then the selection will fill the whole graph. This is the quickest way to zoom to the part that's desired to be analyzed.
- **Set time range**: according to the channel graph selected, the time range to be evaluated can be adjusted from and to a specific time
- **Crop active recordings**: the selected recording will be cropped, keeping the selection, and all other recordings will remain unchanged. If the undo history hasn't cleared up, the crop can be undone by pressing `Ctrl-Z` / `⌘-Z`
- **Crop all recordings**: this will crop all the recordings, even the recordings that are not selected, keeping the data within the selection. If the undo history hasn't cleared up, the crop can be undone by pressing `Ctrl-Z` / `⌘-Z`

8.6.3 Log

The log viewer shows log data recorded from either the UART pin of the **Otii hardware**, or from a third party USB UART. The log view only shows data from the active recording.



8.6.3.1 Filter The *Filter* field enables filtering the messages logged in the console log. Just write the desired message to be found, and it will filter out the messages that have been logged over time.

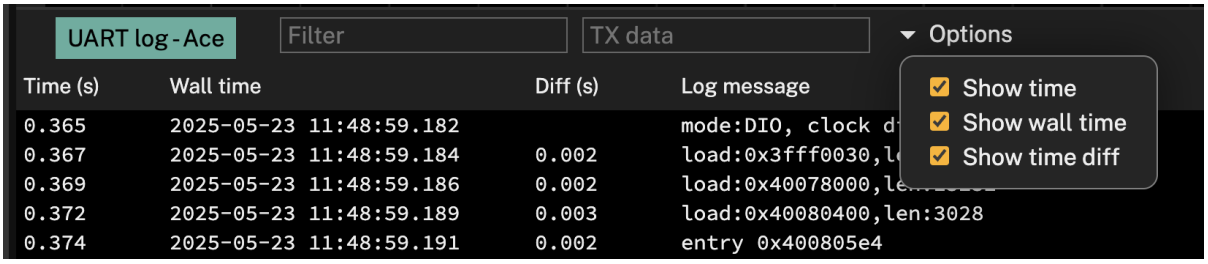


8.6.3.2 TX data The *TX data* field transmits data from the console to the DUT. Making it the ideal interface for sending AT commands or customized configurations to the DUT.

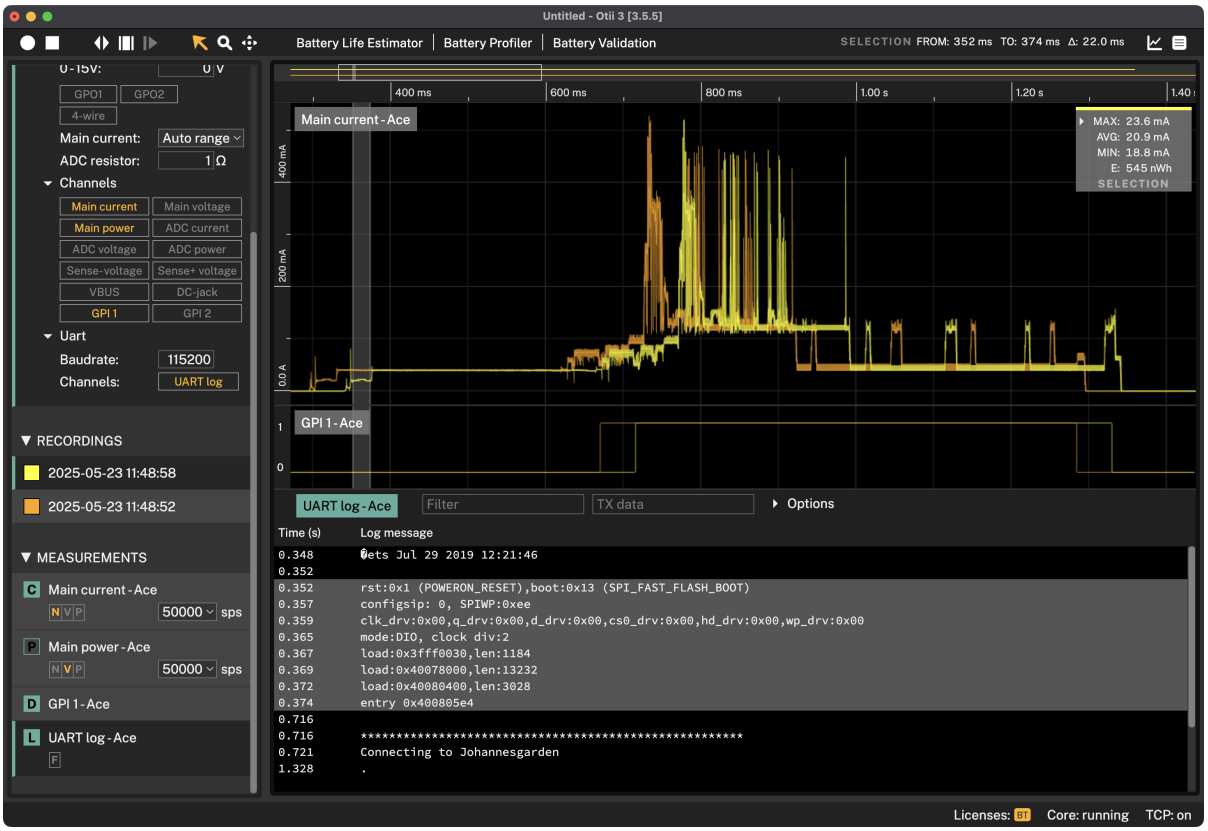
ⓘ A recording must be active to send the data.

8.6.3.3 Time traceability Logging time using the console logo is optional. To activate it, click on the options to select what is to be monitored:

- ** Show time**: It shows the execution time since the DUT was switched on
- ** Show wall time**: t shows the date, in YYYYY-MM-DD format, and the time, in HH:MM:SS format, on which the message has been transmitted. The timestamp displayed is based on the time zone configured on the computer.
- ** Show time diff**: Shows the time diff between the log message and the previous log message.



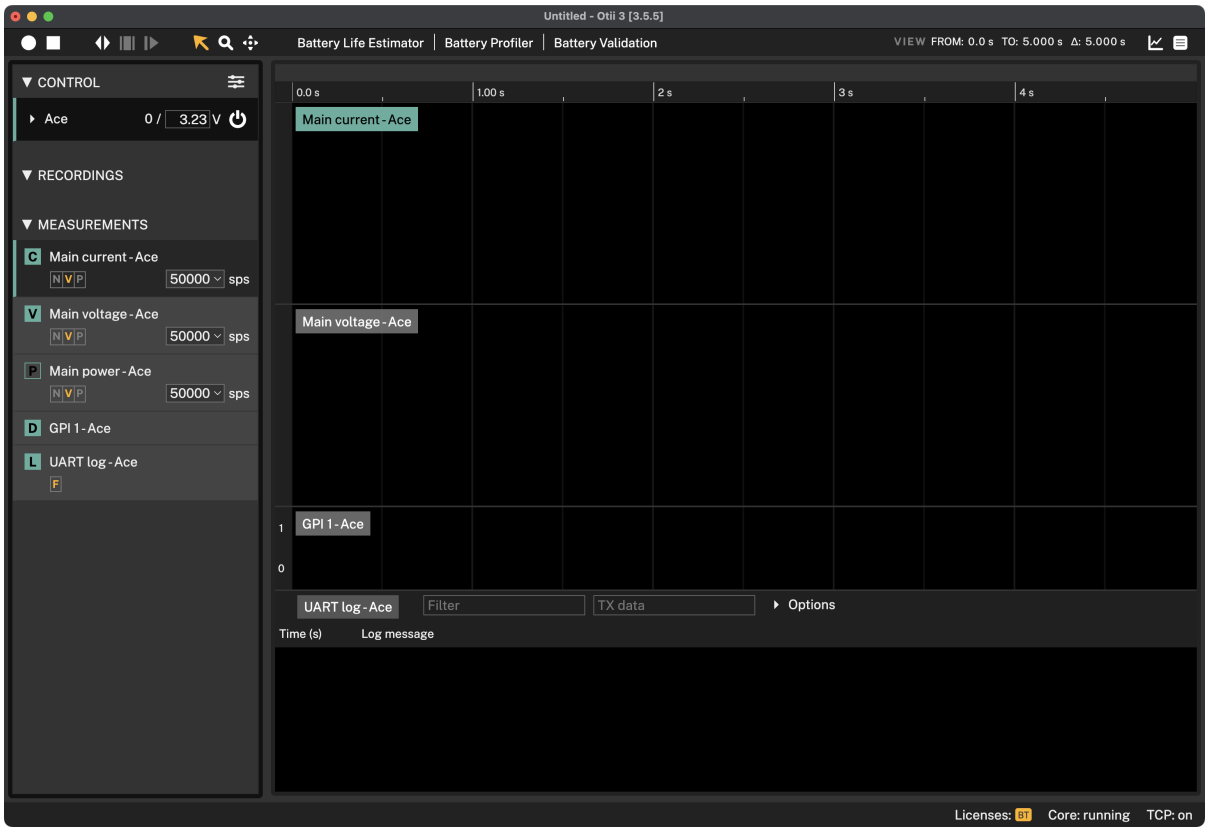
8.6.3.4 Mark messages Mark an area in the graph, and all the UART messages that occurred in this time frame are marked in the UART log. You can also do the opposite and mark one or several messages in the UART log. If you select one message, 1 ms will be highlighted on the graph. At the same time, the current graph for that time frame is marked.



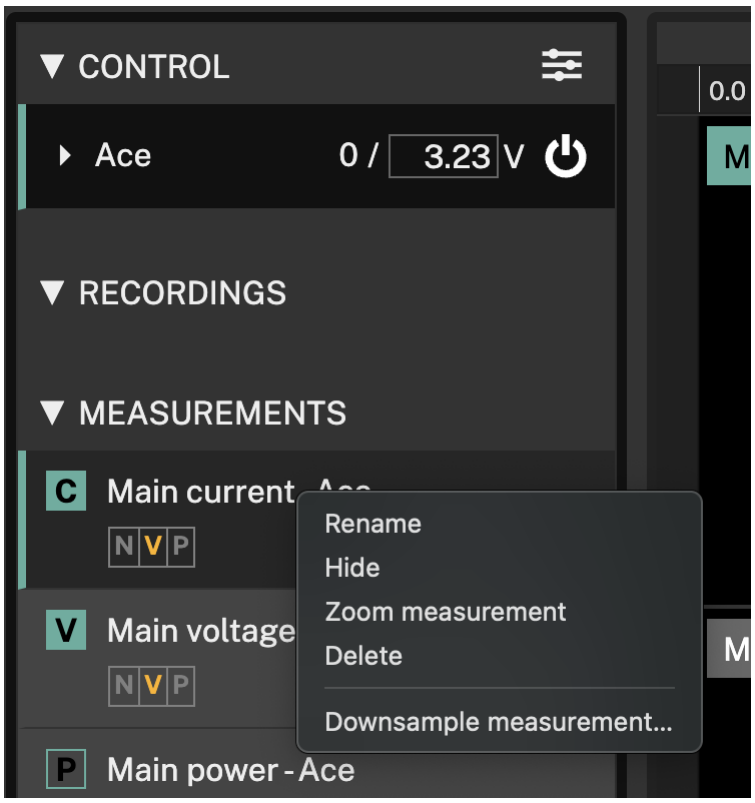
It is possible to select multiple messages using Shift-click or Shift-arrow up/down

9 Measurements

The measurement section includes all the channels that have been selected to be recorded in the general settings tab. These channels are listed under the *MEASUREMENTS* section in the left sidebar of the **Otii 3 Desktop App**.



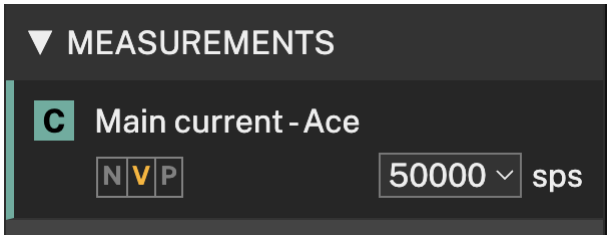
By right-clicking on any of the measurements listed, the following settings menu will be displayed:



Allowing the user to:

- **Rename**: Define a custom name to keep track of the measurement recorded. The measurement can also be easily renamed by double-clicking on the measurement name.
- **Hide**: Hide the measurement to focus the analysis on a different measurement(s). The measurement can also be easily hidden by clicking on the box with the initial letter of the measurement being recorded.
- **Zoom measurement/Show all**: Focus the analysis on one particular measurement. To display all measurements again under the same menu option, click the *Show all* option.
- **Delete**: Delete the recording. Once deleted, the recording can be restored with "undo" `Ctrl-Z` / `⌘-Z` if the undo history hasn't cleared up.
- **Downsample measurement**: Downsampling removes data points from the measurement with average values instead.

Furthermore, under the measurement name, there are three button options that enable the auto scale of the measurements in the Y direction.

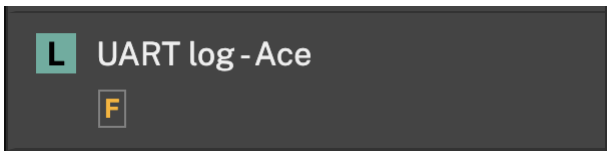


Where you can switch between:

- **N**: no autoscaling
- **V**: auto scaling according to the highest and lowest viewed values

- **P**: auto scaling according to the highest and lowest value in the entire project

Another option can be seen in the UART log.



Where: * **F**: follow UART log. This has the same function as follow recording but is for the UART window. The last message will be shown at the bottom of the log. This will be automatically disabled when scrolling in the UART log, so as a user, you need to select it again to have it follow the last log message.

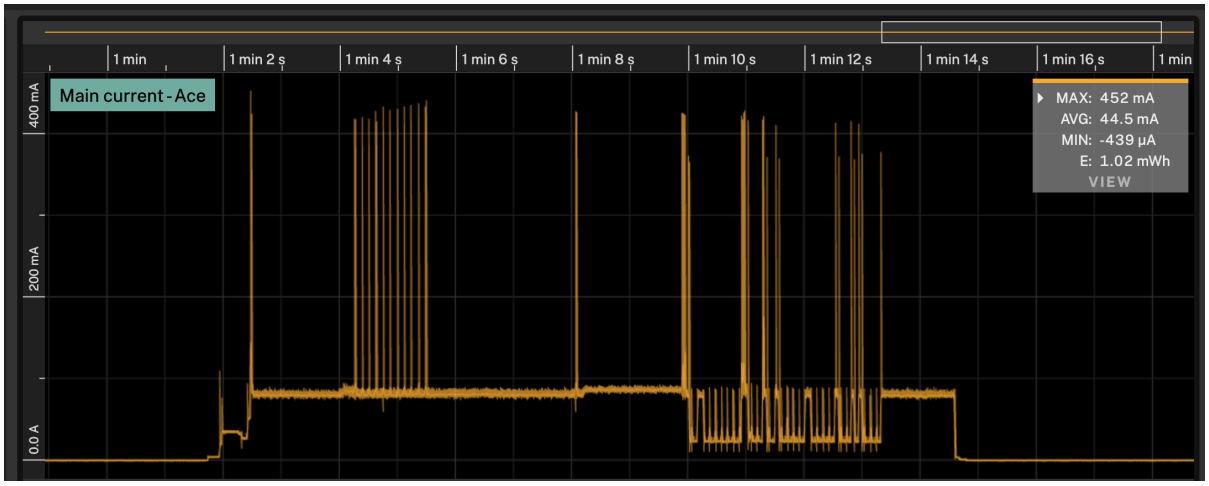
10 Analysis

10.1 Statistics

10.1.1 Statistics for the current recording

By default, with no selections the statistics view shows the statistics for all data in the project.

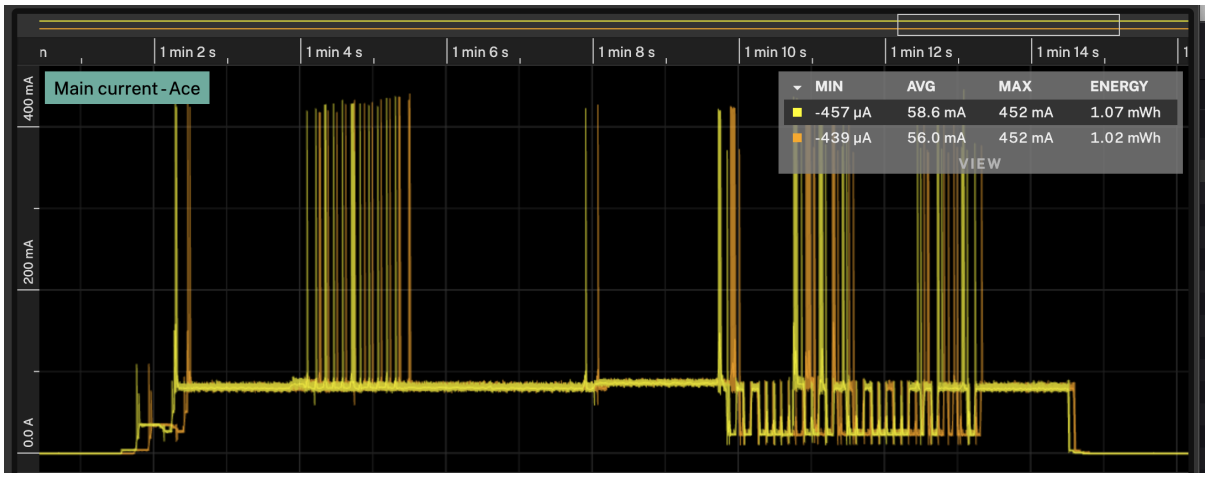
You can change it to be the statistics for the visisble data, by changing the `Statistics` setting in the settings to `View`.



When select a range of data, the statistics view will show the statistics for the selected range.

10.1.2 Statistics for multiple recordings

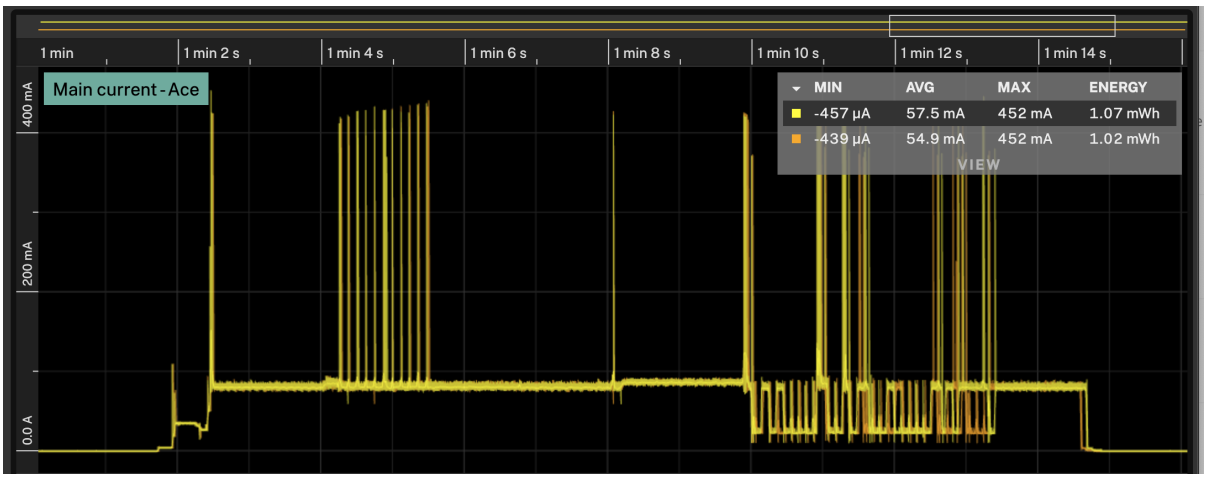
If you have multiple recordings you can compare the statistics for all the visible recordings by clicking the arrow in the top left corner of the statistics view.



10.2 Aligning recordings

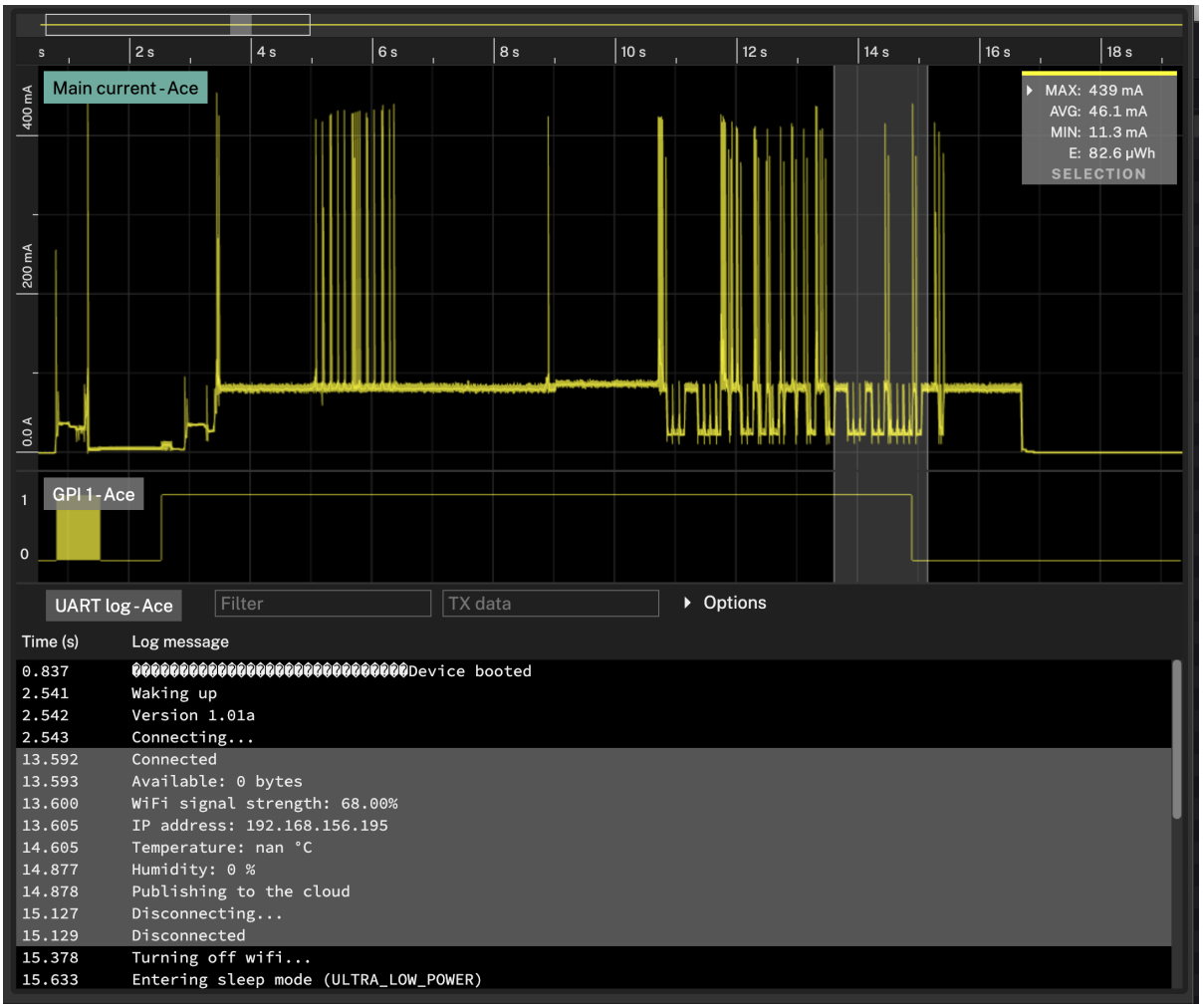
If you want to compare two recordings you can align them by:

- Select the recording you want to offset
- Select the drag tool in the toolbar
- Hold ⌥ while dragging the graph with the mouse
- You can also offset the graph with the keyboard with ⌥-`Left` or ⌥-`Right`

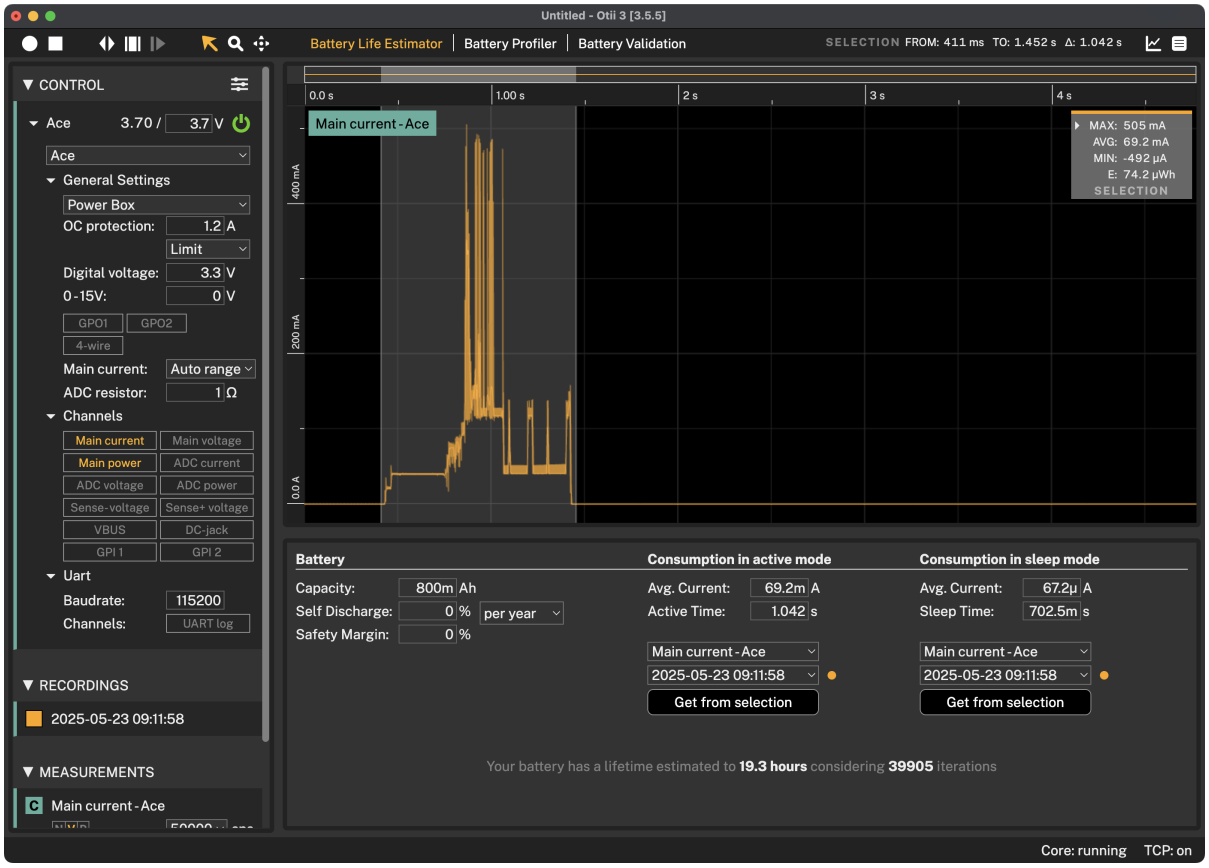


10.3 Sync with log

If you want to analyze a section between two events, you can make a selection in the log, and the corresponding time in the graph will be selected as well.



11 Battery life estimator



11.1 Battery

11.1.1 Capacity

The battery capacity of the battery you want to use. The supported range is from 1 mAh to 999 Ah.

11.1.2 Self discharge

Optionally you can enter a Self discharge in percent and choose the time interval for this value in the dropdown. The self discharge is calculated based on the remaining capacity, for each iteration.

11.1.3 Safety margin

You can also select a Safety margin in percent. This means that estimated battery life calculation will end with the safety margin left of battery capacity.

11.2 Consumption in active mode

11.2.1 Avg. current

The average current of the active period during the duty cycle.

11.2.2 Active time

The active time during the duty cycle.

11.2.3 Get from selection

Use this to get the average current from the current selection.

11.3 Consumption in sleep mode

11.3.1 Avg. current

The average current of the sleep period during the duty cycle.

11.3.2 Active time

The sleep time during the duty cycle.

11.3.3 Get from selection

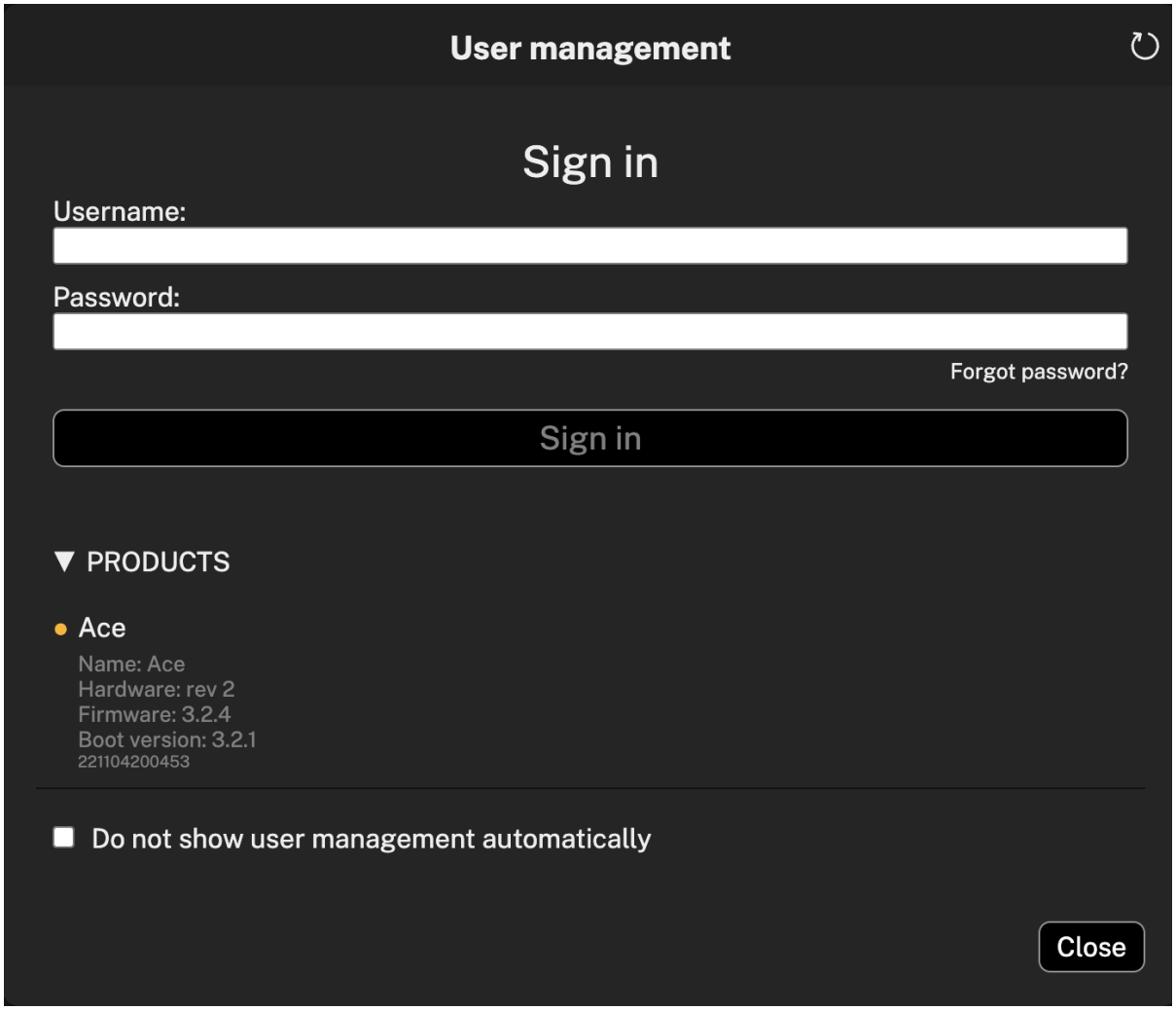
Use this to get the average current from the current selection.

12 Licenses

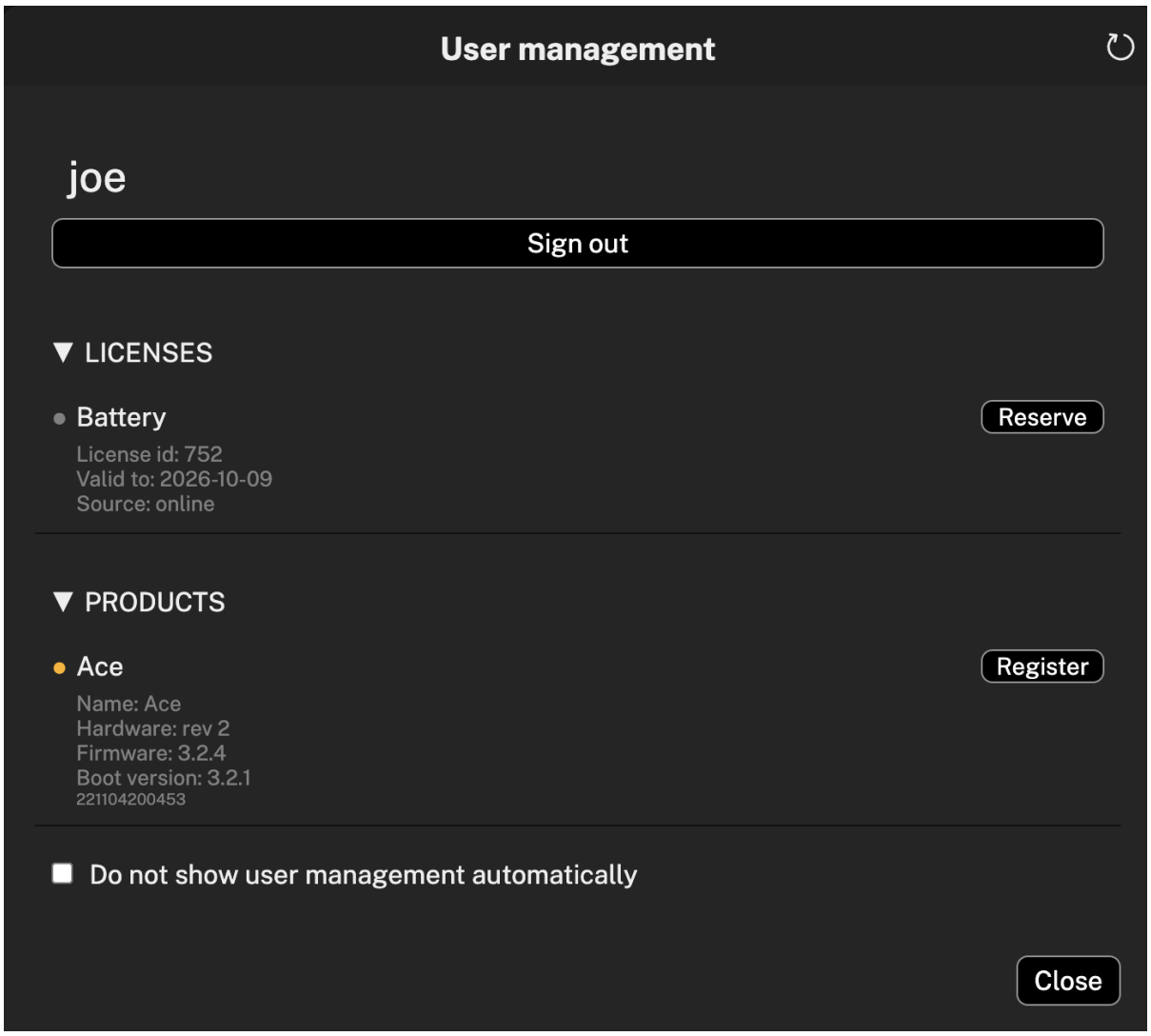
User management allows users to log in to their Qoitech account to access Otii Toolbox licenses and to register Otii hardware.

12.1 Sign in

To open user management, navigate to **Otii3 > User management** or press `Ctrl-U` / `⌘-U`.



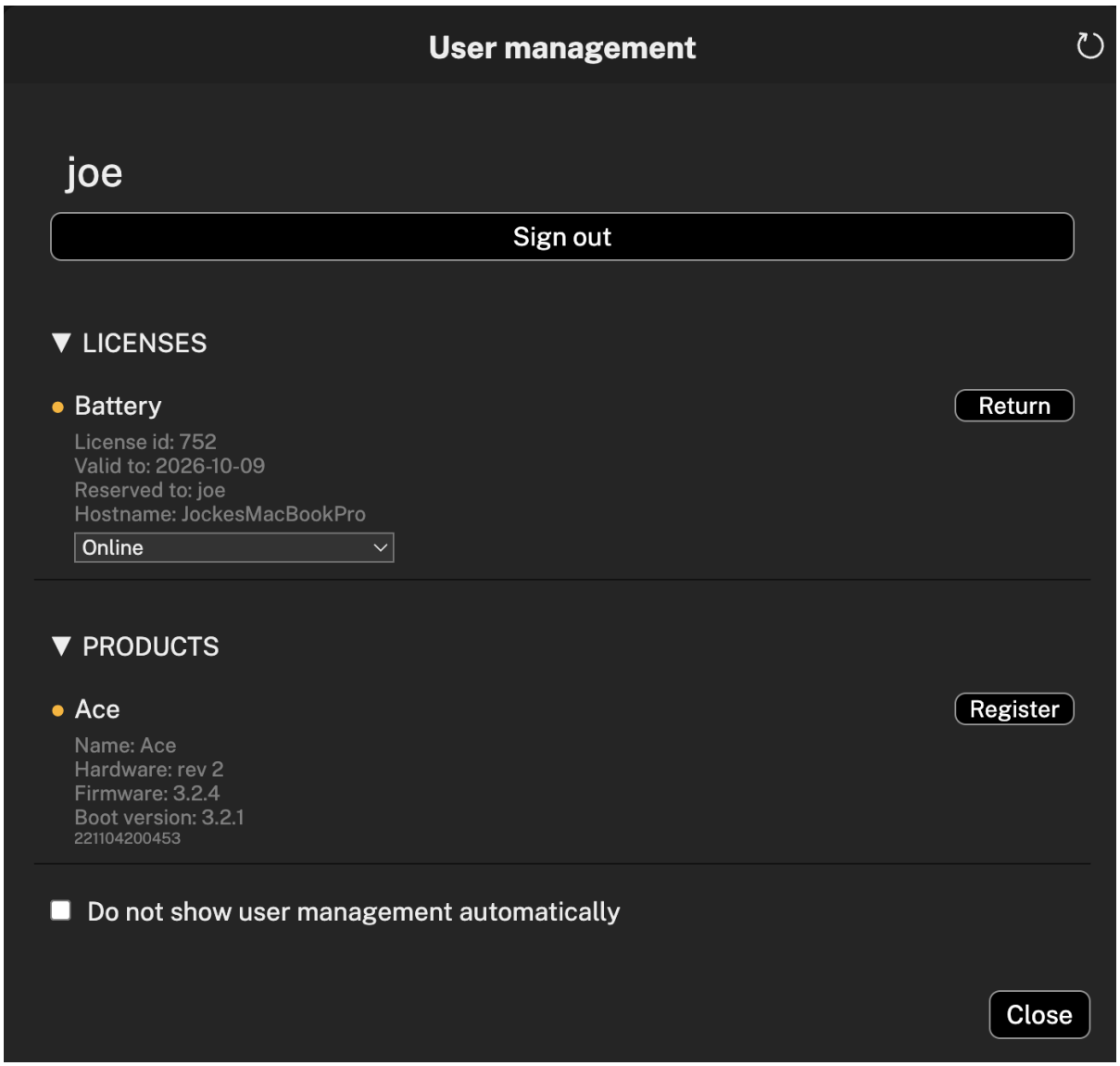
Otii hardware connected to the computer will be shown below the sign in form. Enter your username and password, and click sign in to log in. In case of not having an account, refer to this page to create and activate it.



Once logged in, the licenses associated with the account and the registered devices will be displayed as well as unregistered devices connected to the computer.

12.2 Reserve license

To use the features of a toolbox you need to reserve the license. You click *reserve* to reserve a license, and the license and its features will be available for this computer until you release it. You can reserve more than one license on a computer.



Reserved license will be indicated with a colored dot.

12.3 Return license

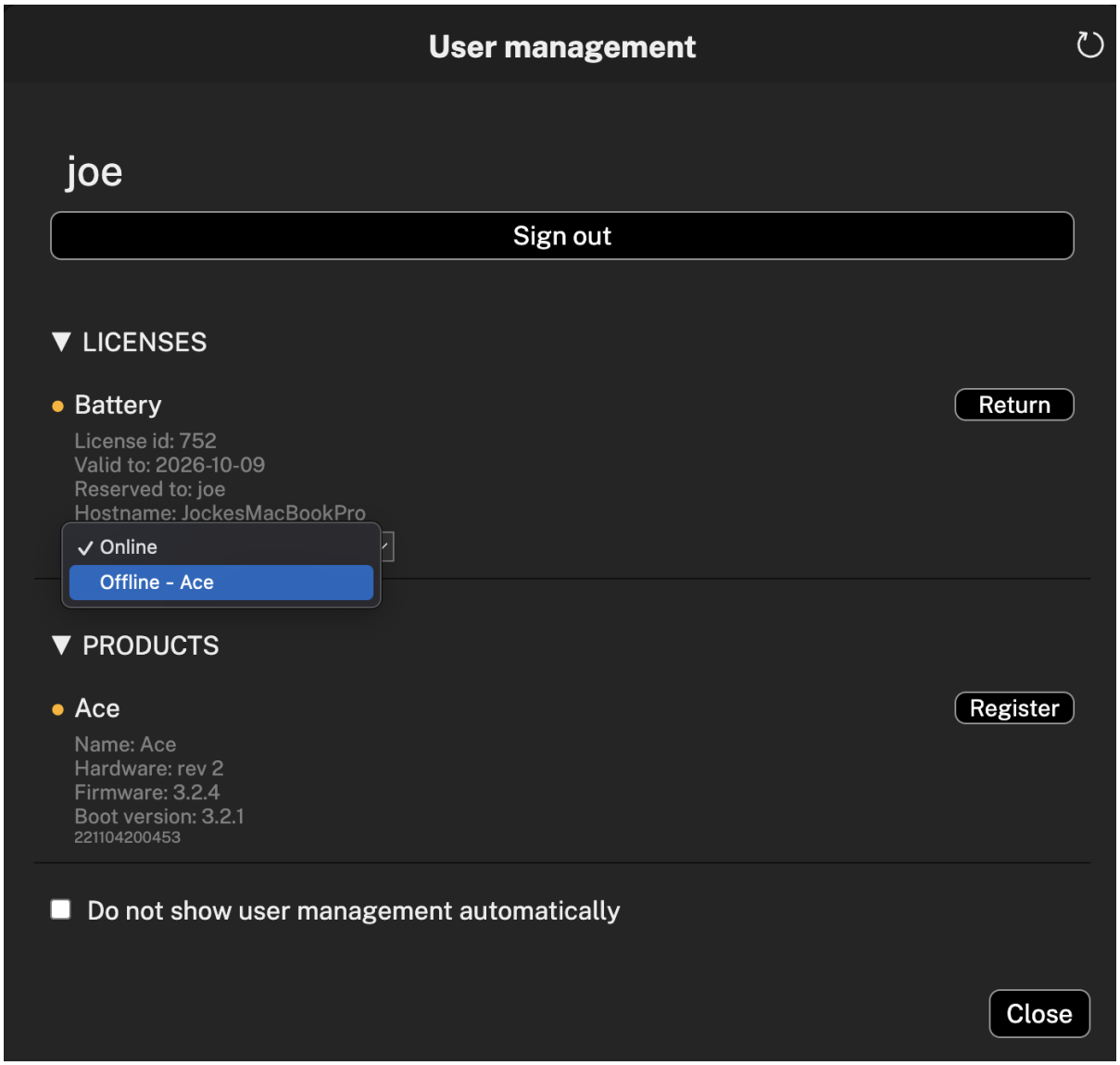
When you are done using the license and want to make it available for use on other computers, you click on *return* to release the license.

12.4 Offline licensing

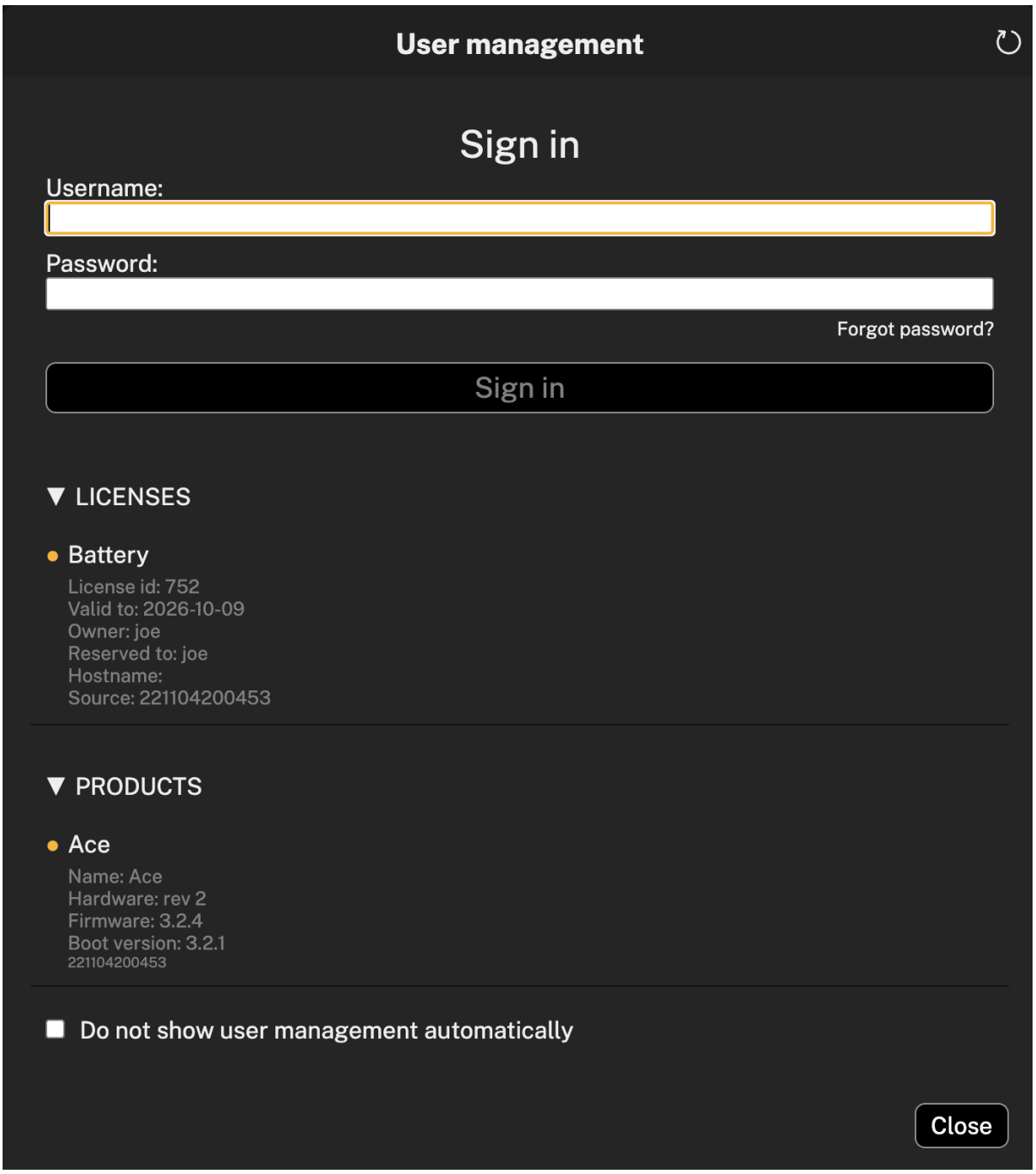
By default you need to be online to use the licenses. This makes it easy to use the license on several computers, one at a time, or to share the license with other users.

If you need to use Otii offline, you can move your online license to an Otii hardware. The license will then be available to the computer this Otii hardware is connected to, and unavailable online.

You need to reserve a license to move it to an Otii hardware. When you reserve it a dropbox will be shown at the bottom of the license stating **Online**. If you click the dropbox all connected devices will be listed, and you can select the device you want to move the license to.



When you sign out, the license will still be available as long as the Otii hardware with the offline license is connected.



To move the license from the hardware back online you need to be signed in, and then you connect the device to your computer and select **Online** again in the dropbox.

12.5 Register device

You can register your device by clicking *Register*, and the device will show up in your online account. You need to register your device if you want to request a calibration certificate.

ⓘ There is no need to be connected to the user management nor to have the devices registered to use the Otii 3 Desktop App.

12.6 License pool

You can easily setup a pool of licenses that are automatically shared between a group of test automation computers.

12.6.1 Create a shared account in the Qoitech User Management

In the User Management you should set up an account that will be shared between the automation computers. You then add any number of automation licenses to this account, either by moving them from another account, or by purchasing new ones.

The credentials to this account should be stored in a JSON-file with the following format:

```
{
    "username": "SHARED ACCOUNT NAME",
    "password": "SHARED ACCOUNT PASSWORD"
}
```

12.6.2 Prepare automation computers

On each automation computer you need:

- Otii 3 running (either Otii 3 Desktop or Otii Server)
- Otii TCP client for Python
- A copy of the credential file mentioned above

12.6.3 Setting up the test code

Since the license pool typically has fewer licenses than there are automation servers, you need to set up the code to wait for a license before running a test.

- You need to specify the location for the JSON-file with the shared credentials
- You need to specify how long the test script should wait to get a license from the license pool

```
from otii_tcp_client import otii_client

CREDENTIALS = './shared_credentials.json'
GET_LICENSE_TIMEOUT = 20 # Seconds

def my_test(otii: otii_client.OtiiClient) -> None:
    # Run test...

def main() -> None:
    client = otii_client.OtiiClient()
```

```
    with client.connect(credentials=CREDENTAILS, try_for_seconds=GET_LICENS
        my_test(otii)

if __name__ == '__main__':
    main()
```

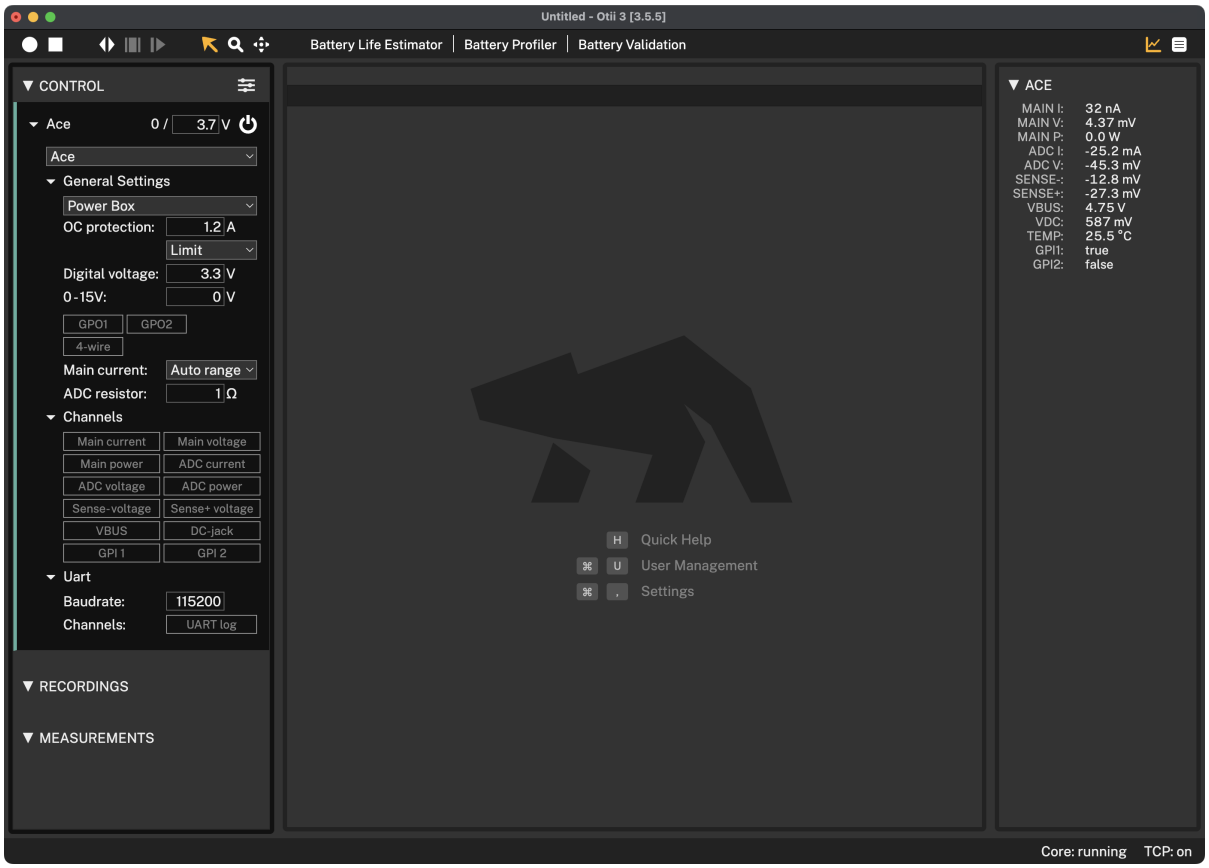
The `client.connect` method will automatically try to login to the license server and reserve a license. If it fails to reserve a license within the time specified it will throw an exception.

After the test is run, the license will be returned, and the client will be logged out.

13 Additional features

13.1 Show monitor

To check the status of your Otii hardware(s) in real-time, enable the “Monitor” feature by clicking on the monitor icon in the upper right corner of the **Otii 3 Desktop App** or by pressing `Ctrl-M` / `⌘-M`. Once enabled, the monitor statistics window will be automatically displayed.

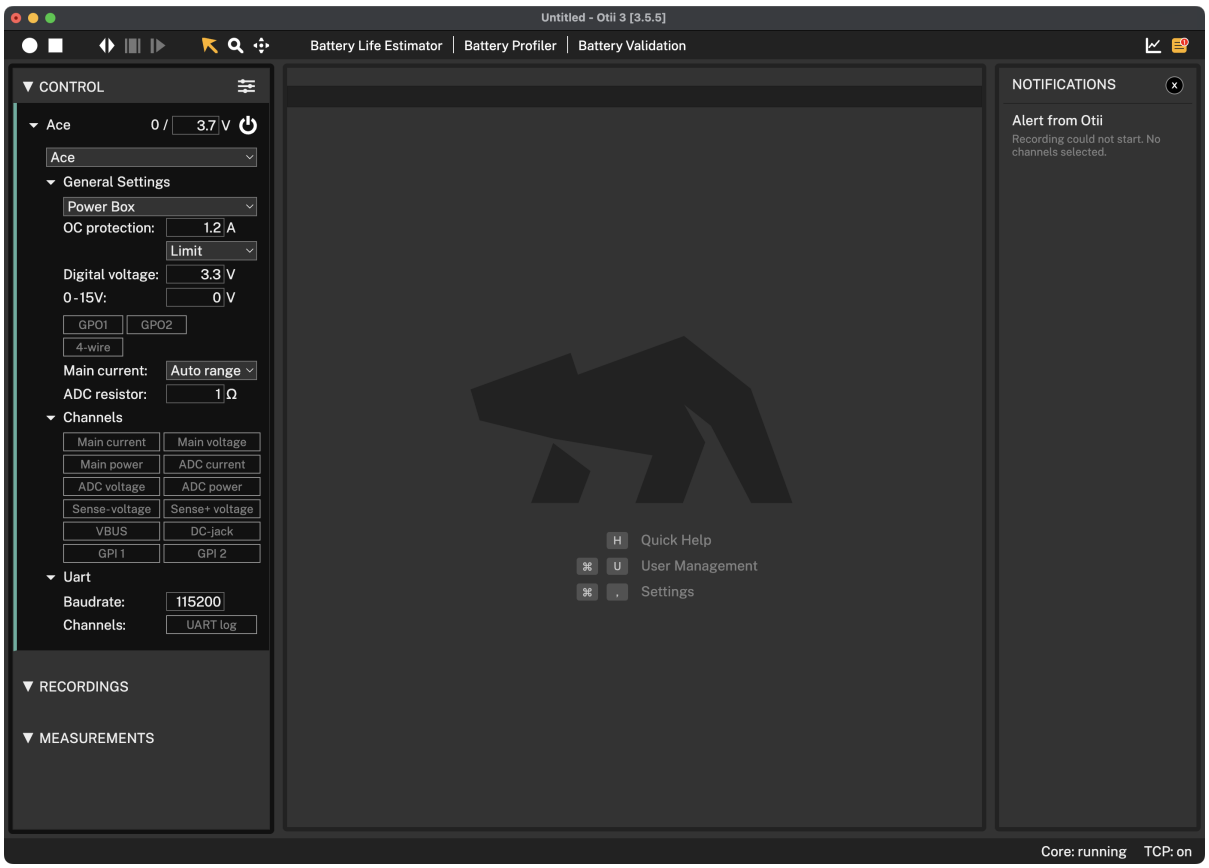


To disable the “Monitor” feature, click again on the icon.

13.2 Notifications

Otii 3 Desktop App features a notification system that updates users on their requests and actions, helping them notice things they might not otherwise, including software releases and relevant information.

To open the notifications panel, click the notification icon in the upper right corner of the app or press `Ctrl-I` / `⌘-I`.



To close a single notification, click the ⊗ next to it. To close all the notifications, click the ⊗ next to *Notifications* title, and then confirm the action by clicking *Clear*.

13.3 Register product

In the User Management dialog, you can register any connected device that hasn't been registered by anyone yet.

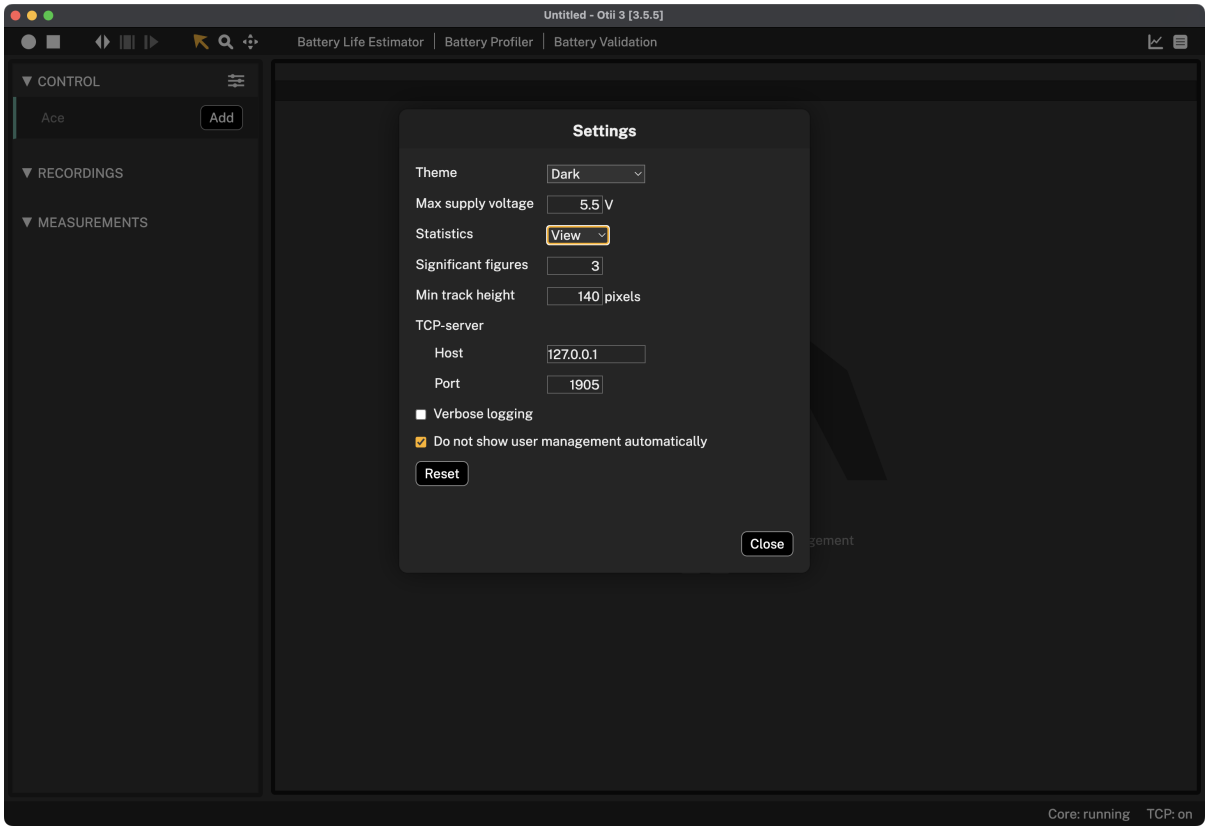
To open user management, navigate to **Otii3 > User management** or press `Ctrl-U` / `⌘-U`.

14 Settings

Otii 3 Desktop App can be easily customized and parameterized to either look or operate as you choose.

1. Open **Otii 3 Desktop App**
2. Navigate to:
    - **Windows & Ubuntu**: File > Settings, or by pressing `Ctrl-,`
    - **macOS**: Otii 3 > Settings, or by pressing `⌘-,`

The settings window will be open:



This section allows to configure the following settings:

- **Theme**: Customize the interface theme as you like. Choose between Dark, Dark - Blue, Dark - Green, and Dark - Yellow.
- **Max supply voltage**: As a safety practice, set the maximum supply voltage based on the device under test (DUT) technical characteristics. The default voltage is 5.5V.
- **Statistics**: Specify the mode in which the statistics will be calculated for the ongoing project when a selection is not made:
    - **Project**: Statistics are calculated for the entire recording.
    - **View**: Statistics are calculated only for the displayed part of the recording.
- **Significant figures**: Affects the number of figures within the statistics and in the monitor.
- **Min track height**: Minimum height of the window measurement before start scrolling it

- **Verbose logging**: When enabled, add more information to the otii3.log to improve debugging.
- **Do not show user management automatically**: When it is not enabled, and you are logged in and have no reserved licenses, User Management will automatically open when a new project is started.
- **Reset**: Resets any settings made and assigns the default settings.

15 Automation toolbox

ⓘ All these features require an Automation Toolbox License.

Otii Automation Toolbox is a software license that elevates Otii software with scripting capabilities to automate measurements based on the project's needs.

This toolbox features:

- Scripting in any language using the Otii TCP Server API
- Packaged scripting modules available from Github for Python, C#, Java and Matlab
- **Otii Server**, a headless server that easily can be integrated in test setups

The Otii Automation Toolbox license is: - Associated with a user, not a specific piece of hardware. - Sharable; a user can share it with another user, but it can only be used by one user at a time.

You can also configure a **license pool** that automatically shares the licenses among your automation setups. When using a shared pool, your test scripts can be configured to wait for a license to become available before execution.

15.1 Otii TCP Server

It is possible to control Otii from another application using the Otii TCP Server API.

Using this API you can control Otii from any application that includes support for standard TCP sockets.

Otii Server is available in both the **Otii 3 Desktop App** and in the **Otii Server** application. You can not run both at the same time on the same computer.

At Qoitech's Github page you will also find wrappers for a few popular languages like Python, C#, Java & Matlab.

15.2 Otii Server

15.2.1 Command Line Interface

You can also start the TCP server using `otii_server`, the command line version of Otii.

The server is packaged within the application and to find where it is located on your system, select **Help** › **Find otii_server** in **Otii Desktop App**.

If you run `otii_server` with the `--help` option you will see all available options:

```
# otii_server --help

Options:
  -V, --version                output the version number
  -v, --verbose                verbose output
  -c, --config <config-file>   configuration file
  -h, --help                   display help for command
```

To start a server run the command:

```
# otii_server
```

The server is by default listening on host address 127.0.0.1 and port 1905. If you want to change the default settings, or need to add settings for a proxy, create a **config.json** file:

```
{
    "tcpserver": {
        "host": "127.0.0.1",
        "port": 1907
    },
    "proxy": {
        "host": "192.168.1.1",
        "port": "8080",
        "auth": {
            "username": "johndoe",
            "password": "mypassword"
        }
    }
}
```

and start the server with:

```
# otii_server --config config.json
```

The server will run until you stop it by pressing `Ctrl-C`.

In a future release Otii will ship with tools that will make it possible to run `otii_server` as a Windows Service, or a daemon in Ubuntu and macOS.

15.3 User management

15.3.1 User management in Desktop application

Read more about logging in and reserving licenses in the **Otii 3 Desktop App** here:

- User management

15.3.2 User management in test script

It is possible to use the TCP-API directly in the test script for user management:

```
#!/usr/bin/env python3
'''
If you want the script to login and reserve a license automatically
add a configuration file called credentials.json in the current folder
using the following format:

    {
        "username": "YOUR USERNAME",
        "password": "YOUR PASSWORD"
    }

Alternatively you can set the environment
variables OTII_USERNAME and OTII_PASSWORD.

'''
from otii_tcp_client import otii_client

def my_test(otii):
    # INSERT TEST CODE HERE

def main() -> None:
    # Connect and login to Otii 3
    client = otii_client.OtiiClient()
    with client.connect() as otii:
        my_test(otii)

if __name__ == '__main__':
    main()
```

The `client.connect()` is used to connect, login and reserve licenses. It works like this:

- If the TCP server isn't already logged in, the credentials will be read from a `credentials.json` file in the current directory, or from the environment variable OTII_USERNAME and OTII_PASSWORD.
- If there is no Automation Toolbox license reserved and there is a license available, it will automatically be reserved.
- By default the system will only try to reserve an Automation Toolbox license. If you need to reserve another Toolbox as well add all the licenses you need to the licenses parameter:

```
client.connect(licenses = [ 'Automation', 'Battery' ])
```

- When disconnecting from Otii all licenses that were implicitly reserved will be returned, and if the connect method logged in to the system, it will log out again.
- If you want to manually reserve a specific license, you use an empty array for the licenses parameter:

```
#!/usr/bin/env python3
from otii_tcp_client import otii_client

# Connect and login to Otii 3
client = otii_client.OtiiClient()
with client.connect(licenses = []) as otii:
    # List all licenses
    licenses = otii.get_licenses()
    for license in licenses:
        items = [
            f'{license["id"]:4d}',
            f'{license["type"]:12}',
            f'{license["reserved_to"]:15}',
            f'{license["hostname"]}'
        ]
        print(' '.join(items))

    # Reserve a license
    otii.reserve_license(licenses[0]["id"])

    # INSERT TEST CODE HERE

    # Return license
    otii.return_license(licenses[0]["id"])
```

15.3.3 User management with Python tool

You can do the user management from the command line using our python module. You install the module with:

```
python3 -m pip install --upgrade otii_tcp_client
```

You can then use the otii_control tool for user management:

```
python3 -m otii_tcp_client.otii_control --help

usage: otii_control.py [-h] {login,logout,list-licenses,reserve-license,ret

Otii Control
```

```
  options:
    -h, --help            show this help message and exit

  commands:
    {login,logout,list-licenses,reserve-license,return-license}
      login               Log in to Qoitech server
      logout              Log out from Qoitech server
      list-licenses       List all available licenses
      reserve-license     Reserve license
      return-license      Return license
```

And here is an example bash script:

```
  #!/usr/bin/env bash

  python3 -m otii_tcp_client.otii_control login --username johndoe --passwor
  python3 -m otii_tcp_client.otii_control reserve-license --id 1234
  ./my_test.py
  python3 -m otii_tcp_client.otii_control return-license --id 1234
  python3 -m otii_tcp_client.otii_control logout
```

16 Battery toolbox

ⓘ All these features require a Battery Toolbox License.

The Otii Battery Toolbox is a software license that extends the capabilities of the **Otii Product Suite** to enable battery profiling, battery simulation, battery testing and validation for low-power IoT devices and electronics.

The Otii Battery Toolbox license is:

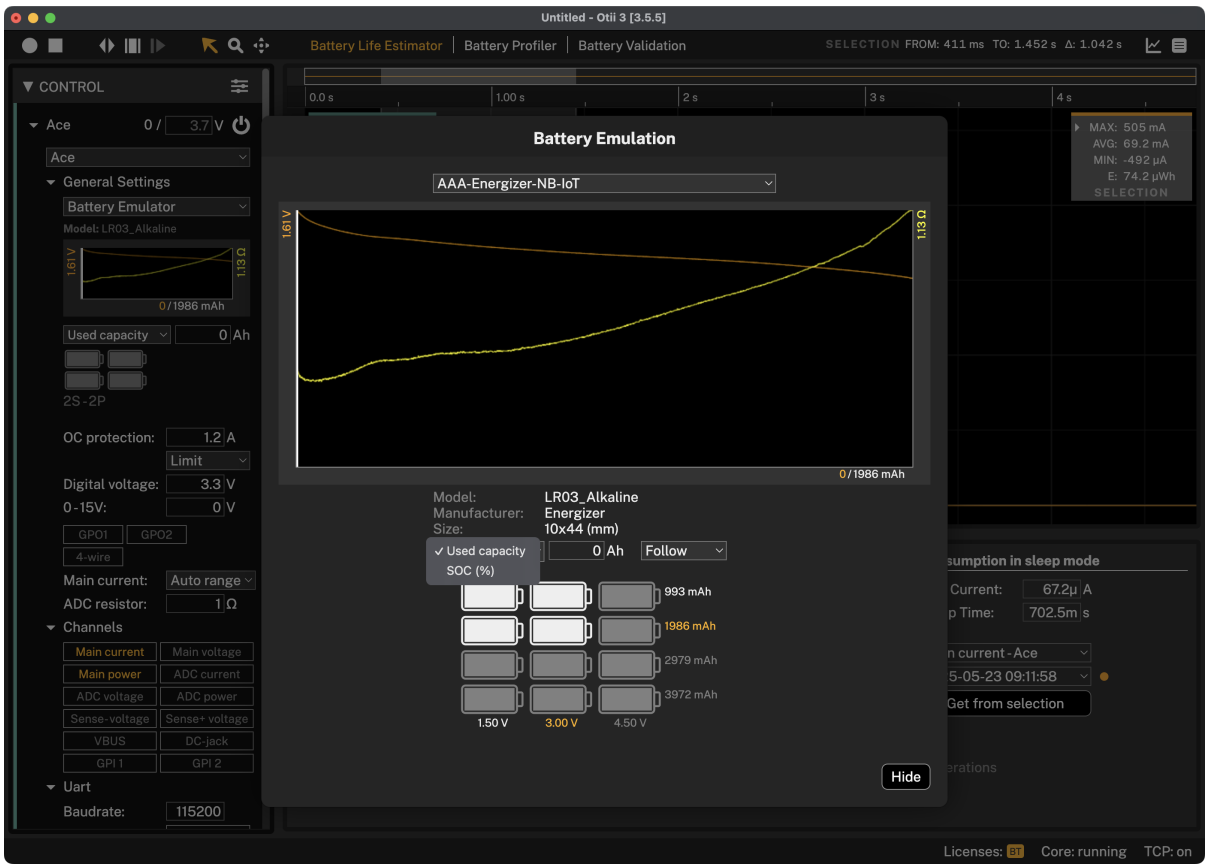
- Associated with a user, not a specific piece of hardware.
- Sharable; a user can share it with another user, but it can only be used by one user at a time.

Using Otii Automation Toolbox, you can also configure a license pool that automatically shares the licenses among your battery evaluations setups. When using a shared pool, your test scripts can be configured to wait for a license to become available before execution.

16.1 Battery emulation

Otii can act as a battery, following a discharge curve. If you have reserved a **Battery Toolbox** license, a Supply section will appear in the Arc/Ace control settings. Press the Emulate battery button to open the battery emulation dialog. Choose the battery profile you want in the drop down list. A discharge curve will be shown for the chosen battery.

There are two curves, one shows the unloaded battery voltage over the used capacity and the other curve shows the internal resistance. Otii will adjust the output voltage depending on the load, just like a real battery with internal resistance.



16.1.1 Used Capacity

Enter how many Ahs of the emulated battery's capacity that has been used if you selected Used capacity or how many percent if you selected SOC(%). To emulate a fresh battery, enter 0 for Used capacity and 100% for SOC(%).

Select Fixed to emulate the battery with a constant Used capacity as entered above. Follow will emulate a discharge over the time you are recording.

The rest of the settings are the same as when using the Arc as a constant power supply.

16.1.2 Series and Parallel

Click on a battery in the battery grid to set how many batteries in series and parallel to emulate.

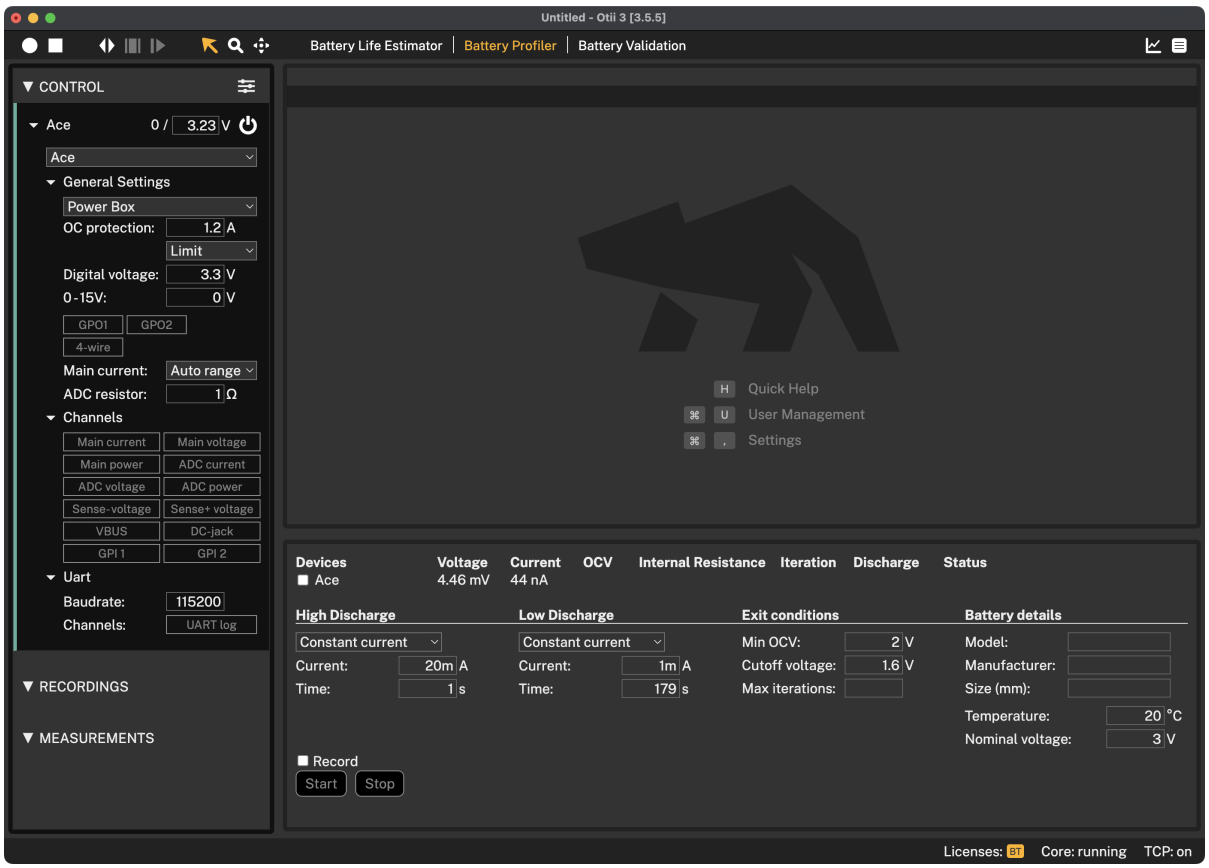
16.1.3 Hide

Press to hide the battery emulation dialog. Your current settings will still be visible in your Arc/Ace control settings, as displayed below. Here you can edit the used capacity or state of charge. To update other settings, press the curve to re-open the dialog.

16.2 Battery profiler

In addition to source current, the Arc can be set to sink current. E.g. by connecting a battery to the main connectors, you can test how the battery performs with different kind of loads.

To get started with battery profiling, choosing Battery Profiler from the toolbar will open the following pane:



16.2.1 Devices

Here you choose which of your connected devices you want to use for battery profiling. When the profiling is running you will see some information of the current status of the profiling for each device.

16.2.2 Profiling Settings

Here you configure the profiling parameters, see more below on some best practices in choosing them.

You can also choose to automatically start a recording of the profiling. Recording the profiling session is optional, and for longer profiling sessions will require a lot of free space on your computer.

16.2.3 Battery Details

Here you enter the model, manufacturer and size of the tested battery.

16.2.4 Output

Scroll down the dialog to see a more detailed output from the profiling.

16.2.5 Start

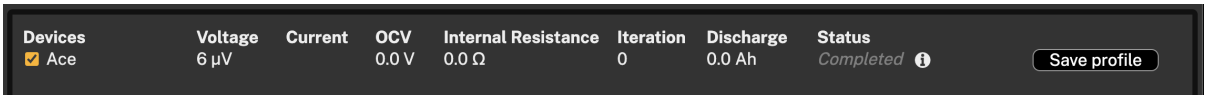
Start is used to start a new profiling.

16.2.6 Stop

Stop is used to stop an ongoing profiling.

16.2.7 Save profile

When any of the exit conditions has been fulfilled, or the stop button has been pressed, a save button will appear after each device allowing you to save the profile for use in the battery emulator.



16.3 Getting started with battery profiling

1. Connect the battery positive pole to the red banana connector (+) and the negative pole to the black banana connector (-)
2. Start **Otii 3 Desktop App**, log in and reserve a **Battery Toolbox license**
3. Open Battery Profiler tab
4. Select the **Otii Arc/Ace(s)** that is going to profile the battery/batteries. It is possible to have several Otii Arc/Aces connected to the same computer to be able to profile several batteries at the same time
5. Select if the discharge should be constant current/power or resistance for both the High Discharge and Low Discharge. This represent how your device draws energy from the battery. Typically, a LDO draws energy in constant current mode, a DC/DC in constant power mode and a resistive leakage in constant resistance mode.
6. Input data for the High discharge and the Low Discharge period. This could be the active period and the sleep period of your device, how if draws the energy from the battery. There must be a difference in the High discharge and Low Discharge and High Discharge must be higher than Low discharge. This to be able to calculate the internal resistance of the battery.
7. Input data for when the battery profiling is to be stopped, there are three possible exit condition and if one of them are met, then the profiling stops.
    1. Min OCV is the calculated Open Circuit Voltage (OCV) of the battery. It is not necessary to have a 0 load to be able to react on OCV as this is a calculated value based on the measured battery data.
    2. Cutoff voltage is the measured battery pole voltage
    3. Max iterations is if you would like to limit the number of iterations. If you do not want to limit the number of iterations, just leave this field empty

8. Input the Battery details data like the module, manufacturer and size of the battery. This is just for your information, to remember what battery that was profiled.
9. Input what temperature the battery was profiled in. It is good to have the battery profiled in the different temperatures that it will be subjected to in the real use case. Make sure to only put the battery inside the temperature chamber, not the Arc/Ace.
10. Input the battery datasheet nominal voltage of the battery.
11. Select if you want to have a recording ongoing during the battery profiling. This is not needed, the battery data is being measured anyways.
12. Enable Main current and Main voltage channels and set wanted sample rate if a recording is wanted during profiling. Typically, the sample rate is set to 1000 samples per second as batteries has slow response.
13. Click Start button to start the battery profiling and then wait for it to finish.
14. When battery profiling has met an exit condition, the relay opens and a button “Save profile” appears. Click this to save the generated battery profile. The measured battery profiling data is also stored in the active project.

16.4 Why do I get internal resistance warning when profiling a battery

The internal resistance warning is a warning, so the profiling continues regardless.

The warning is that the internal resistance calculation ended up with a negative resistance. To understand this, the theory behind the internal resistance calculation needs to be understood.

Battery profiler discharges the battery with a high discharge pulse and a low discharge pulse. When the switch from high to low happens, the voltage (VH) and current (IH) is measured . Then when the switch back from low to high happens, the voltage (VL) and current (IL) is measured. The internal resistance is then calculated by Ohms law Rint = (VL-VH)/(IL-IH) The currents are negative. If the battery has a lower voltage in the end of the low discharge, than the voltage at the end of the high discharge, Rin will be negative.

To avoid this do the following:

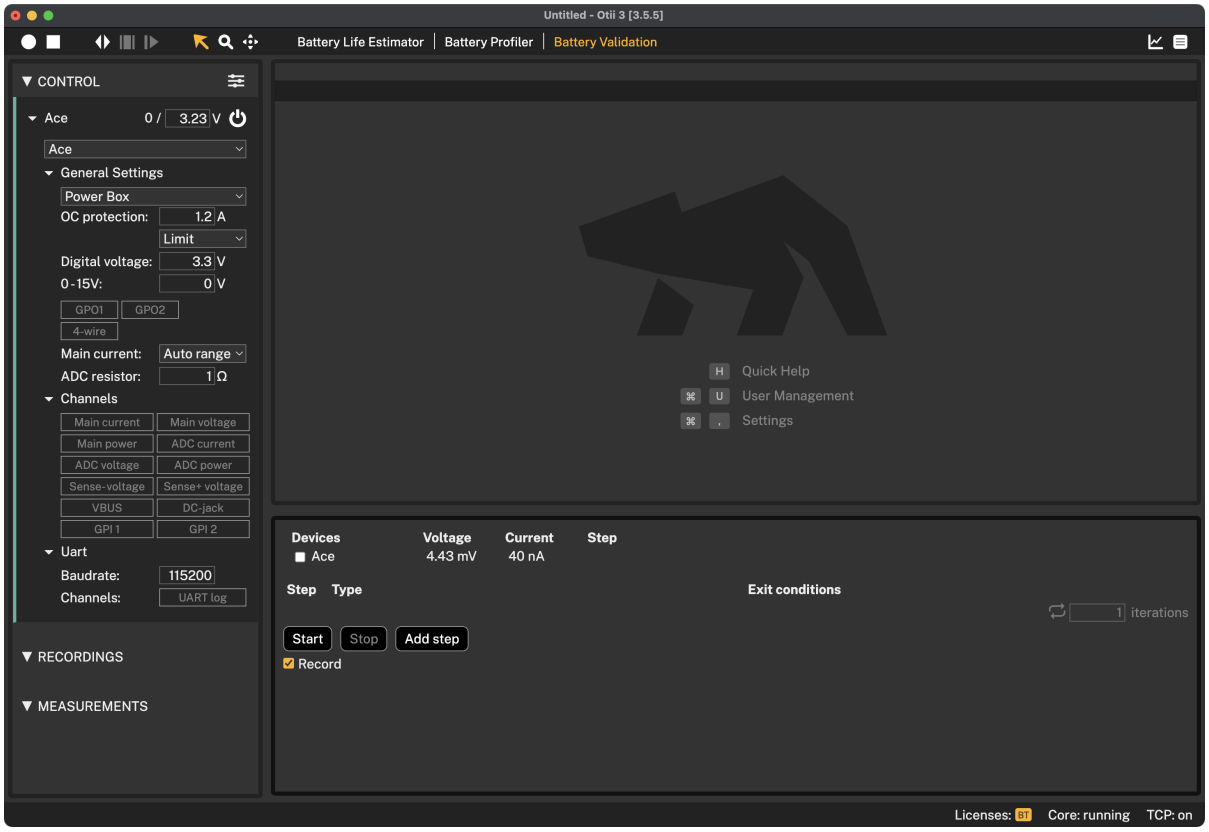
- Have a shorter cycle time
- Have a higher difference between high and low current

16.5 Battery Validation

ⓘ Battery Validation requires an Otii Ace Pro.

Battery Validation provides a comprehensive view of your battery by monitoring its performance in various scenarios. You can charge and discharge the battery in one or multiple steps, repeat discharge cycles, and create custom scenarios.

To start using Battery Validation, select Battery Validation in the toolbar or press `Ctrl-T` / `Cmd-T` combination. A panel will then appear where you can view and modify the validation configuration.



16.5.1 Devices

Here you will see a list of connected devices where you can pick which of them you want to use for battery validation.

16.5.2 Start

To start battery validation, press *Start*.

16.5.3 Stop

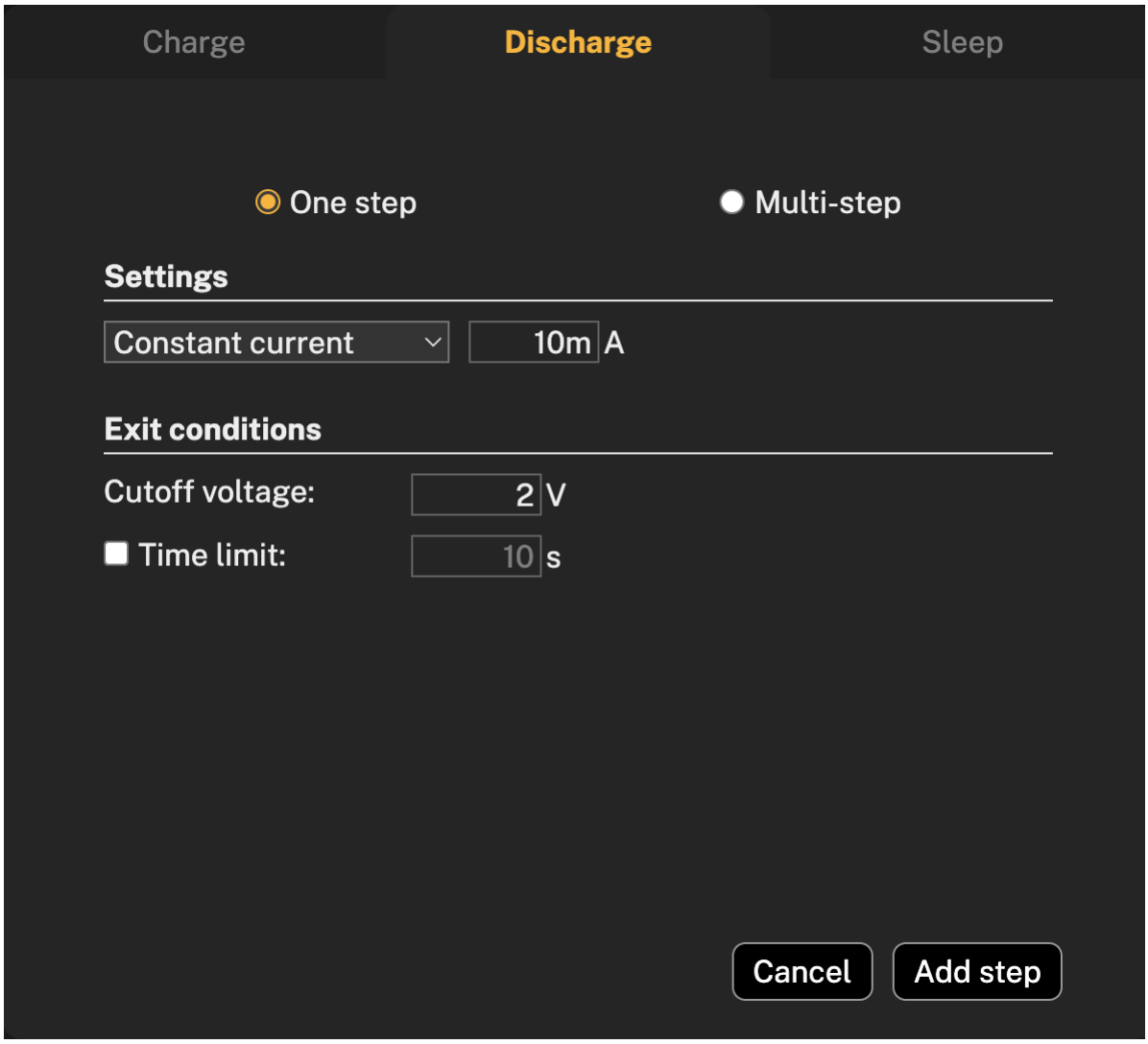
To stop battery validation, press *Stop*.

16.5.4 Record

You can optionally record the ongoing battery validation session by selecting Record.

16.5.5 Add step

Press Add step to open the battery validation dialog.



16.5.6 Discharge step

The discarge step function is splitted by One step and Multi-step.

16.5.7 One step

One step allows you to have a discharge step without specifying max time.

16.5.8 Settings

In the settings you can pick between Constant current, Constant power or Constant resistance and insert you value, according to the battery under the test.

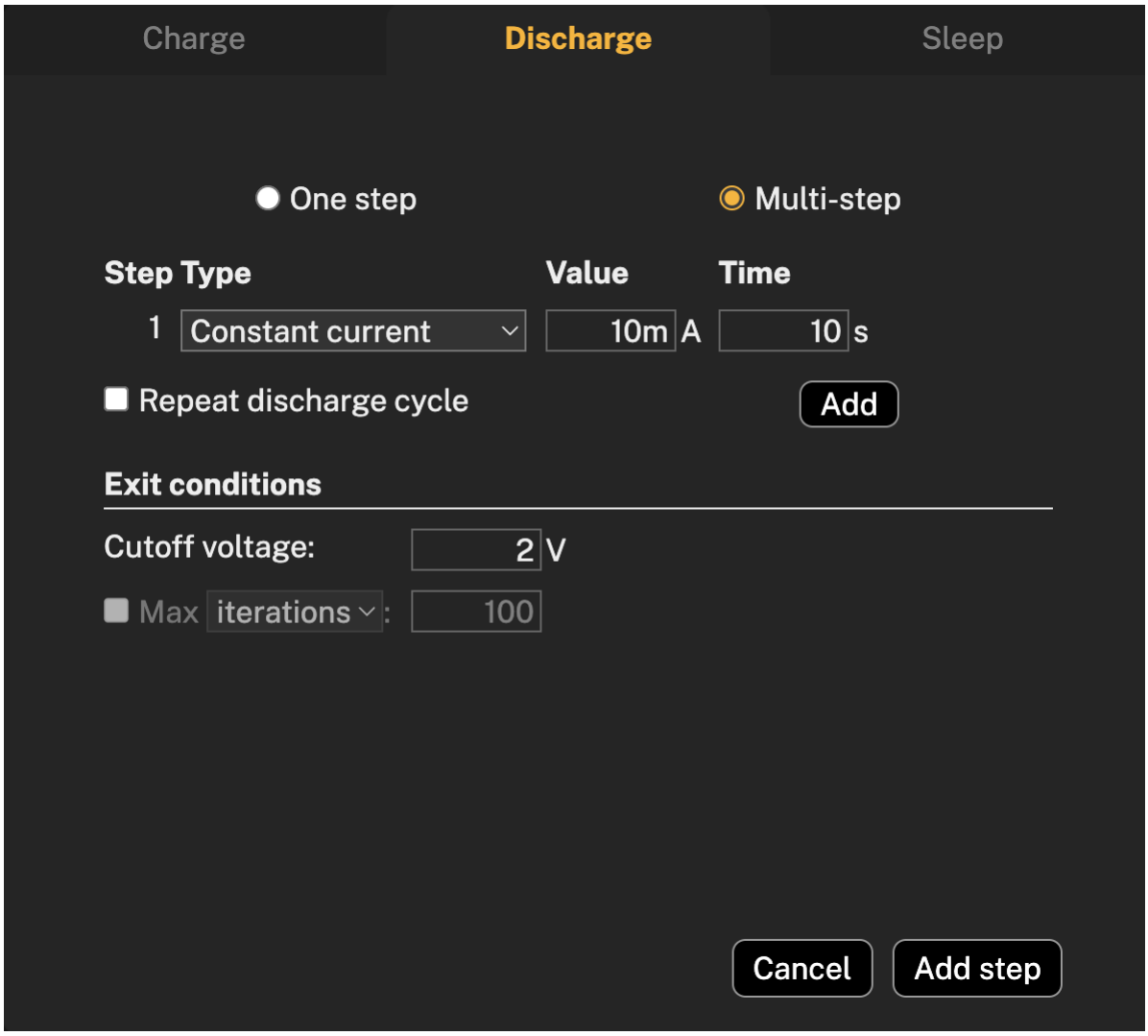
16.5.9 Exit conditions

The exit conditions for discharging a battery are the conditions under which the battery should be stopped from discharging to prevent damage to the battery. Cutoff voltage means setting a minimum voltage below which the batterry is not allowed to discharge. Check battery datasheet for this information. Time limit is optional in case of One step discharge, but recommends in case if the battery is discharging at a very low current and the voltage cutoff may not be triggered, also to prevent overdischarging or other unexpected behaviours.

To add step, press Add step button and the configured step will appear in the dialog.

16.5.10 Multi-step

A multi-step battery discharge involves discharging the battery in several steps, each with a different constant current, power, or resistance profile, depending on the desired outcome and battery type.



16.6 Step type

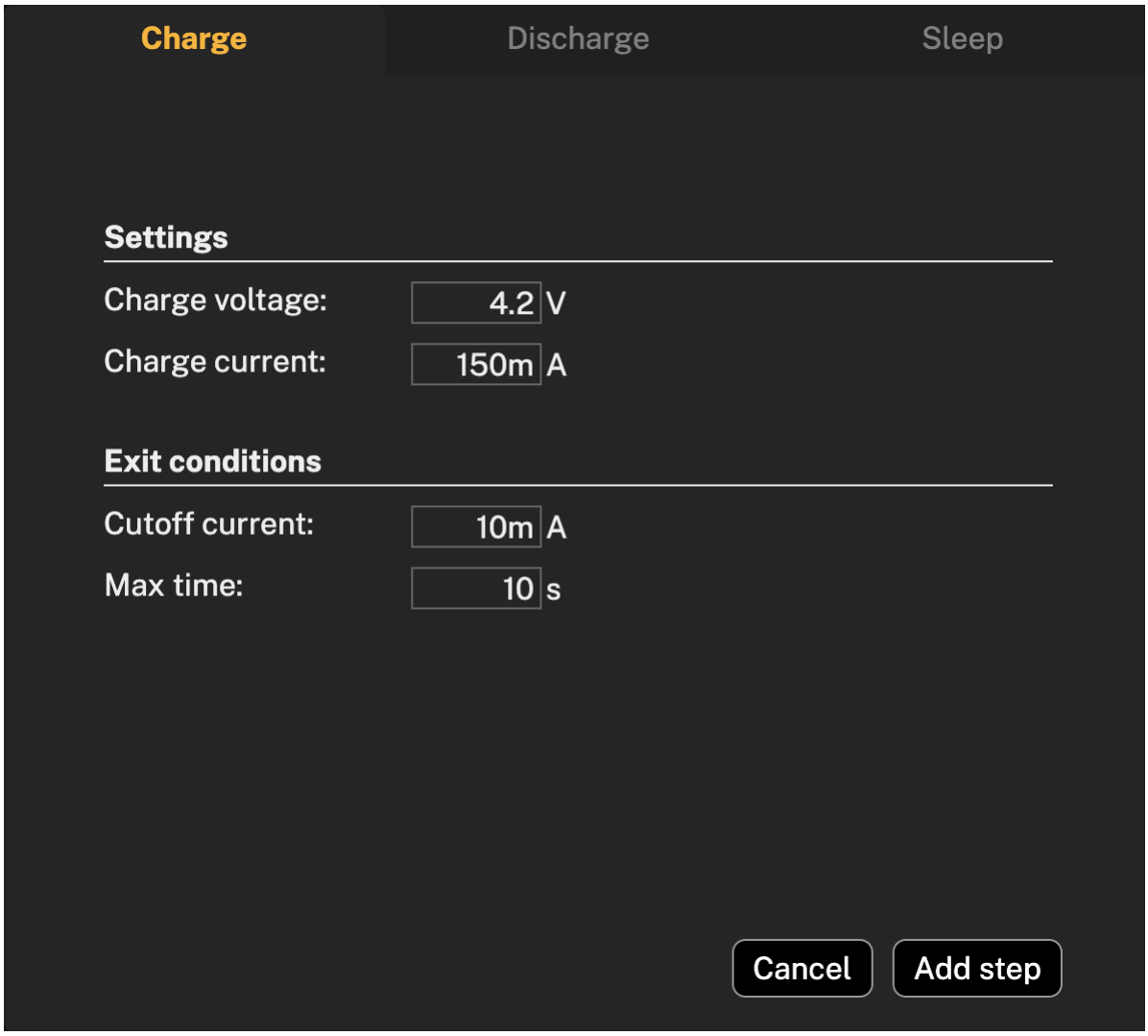
To create a profile with several step types for battery discharge, select the desired step types and configure the parameters for each step, including time, and press Add to add more steps. It is also possible to repeat discharge cycle by selecting Repeat discharge cycle.

The exit conditions in the multi-step discharge, besides cutoff voltage that works same principle as in one step discharge, contains a checkbox where in case if you decide to have repeated discharge cycle, you can choose between Max iterations or Max time.

To add multi-steps, press on Add step button and configured multi-steps will appear in the dialog.

16.6.1 Charge

A battery charging test is often used to evaluate a battery design to see if it meets its performance and safety specifications.



16.6.2 Settings

Before charging the battery, make sure to set up the settings according to the documentation of the tested battery, which is includes Charge voltage and Charge current.

Charge voltage is the voltage applied to a battery to overcome its internal resistance and charge it. It is typically higher than the battery's nominal voltage.

Charge current is the electric current that flows into a battery while its being charged. It's also very dependent on the type of battery, and before setting up the value, check the datasheet of the battery under test.

The exit conditions of battery charging are the Cutoff current and Max time of charging the battery.

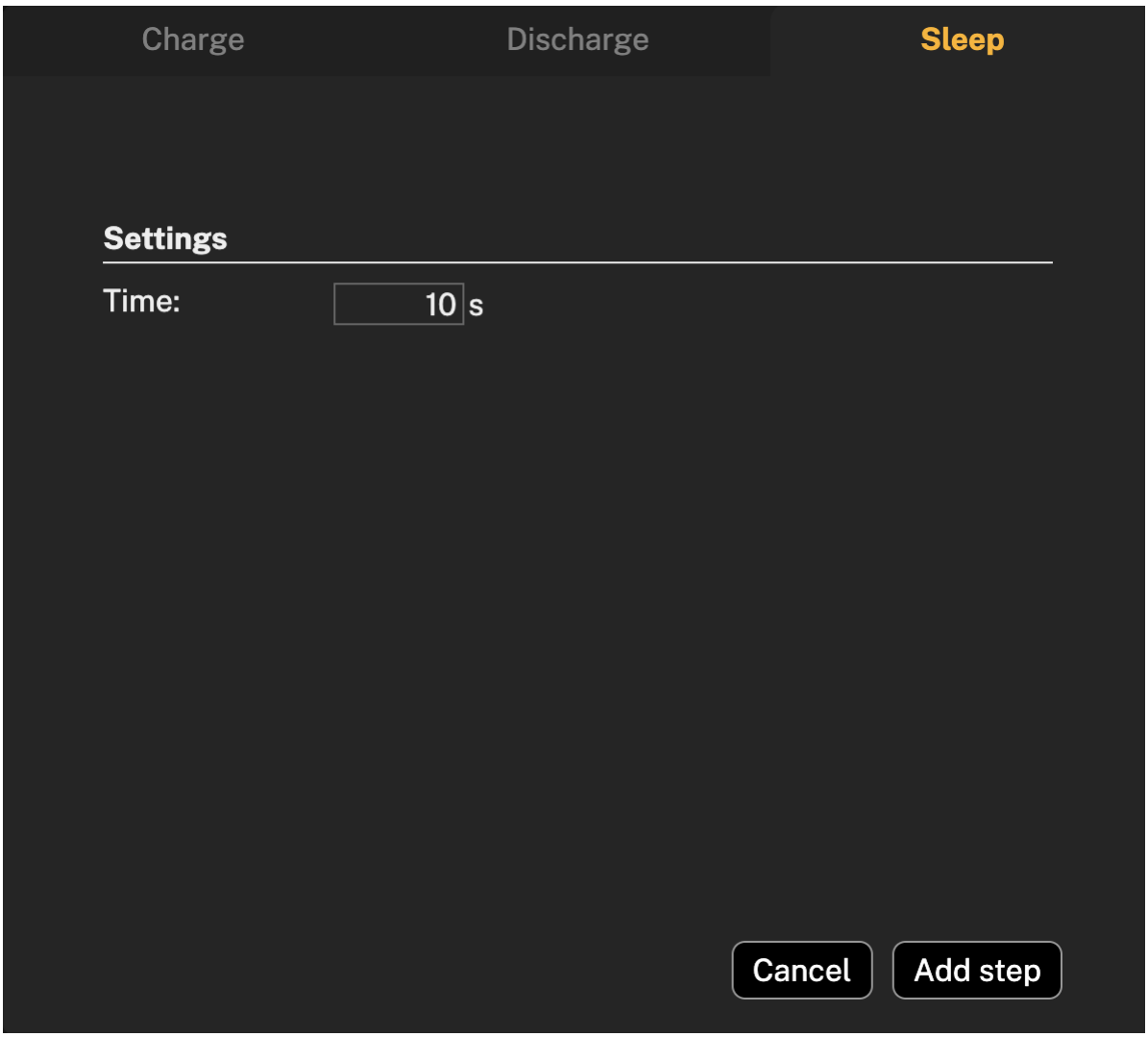
Cutoff current is the current at which a battery will stop charging; this prevents battery's over-charging.

Max time is the longest amount of time that a battery will be charging. This ensures that the battery is not being overcharged.

Battery types will have a different cutoff current and maximum charging time, so make sure to read the documentation first.

16.6.3 Sleep

Due to different battery types and their chemistry, battery temperature and protection security it is important to have a time sleep between charging/discharging steps. To set up sleep time between selected steps, swich to the Sleep tab and set up desired sleep time. Press Add step.



16.7 Battery model parameters

Battery model parameter	Otii Arc Pro	Otii Ace Pro
Points in Emulation	as many as no of iterations	as many as no of iterations
ESR Range[28]	up to 5 kohm	up to 5 kohm
ESR Resolution	down to 1 mohm	down to 1 mohm
Voc Range	0.5 V to 5 V	0 V to 25 V
Voc Resolution	1 mV	1 mV
Capacity Range	no limit	no limit
Capacity Resolution	1 µAh	1 µAh

[28] Otii Battery Toolbox emulates the total ESR.

17 Python scripting

ⓘ These features require an Automation Toolbox license.

Otii has a build in TCP Server that can be used to control the application from another application, e.g. from a script running in your C.I. environment.

You can write your own TCP client using the Otii TCP Server API, or use one of the clients available at the Qoitech Github.

In this guide we will show you how to control Otii from a Python application using the Otii TCP Client for Python.

You can find out more about how to get started with the TCP server here:

- Automation Toolbox
- TCP Server API

17.1 Installing the Python client

You can install the python client using pip:

```
python3 -m pip install otii_tcp_client
```

17.2 Running the TCP Server

You need to have Otii TCP Server running, either using the Otii desktop client or the Otii command line interface. You can read about it here:

- Otii TCP Server

For this example we also expect the server to be logged in, and that an Automation Toolbox license is already reserved. You can read more about logging in and reserving licenses here:

- User management

17.3 Automate measurements

Let us create a python script, e.g. `otii_measurement.py`. The first thing we need to do is include the Otii python client:

```
#!/usr/bin/env python3
import time
from otii_tcp_client import otii_client
```

The next step is to establish a connection with Otii TCP Server. Here we are assuming the TCP Server is running on the same computer and using the default TCP port.

```
client = otii_client.OtiiClient()
otii = client.connect()
```

After we are connected, we want to make sure that there is exactly on device connected to Otii, and get the handle to this device. We also make sure the device is added to the current project.

```
devices = otii.get_devices()
if len(devices) != 1:
    raise Exception(f'Expected to find exactly 1 device, found {len(devices
device = devices[0]
device.add_to_project()
```

For this script we want to make one 10 second recording of the main channels of the Arc connected. We want each run to add the recording to the existing project, so that we later on can compare them.

```
project = otii.get_active_project()
```

Next step is to configure the Otii Arc/Ace for our project. In this case the device is normally running on a coin-cell battery, so we set the main voltage to 3.0. We want to use the high accuracy measurement (low range), and we want to measure the main current (mc) and the main voltage (mv).

```
device.set_main_voltage(3.0)
device.set_max_current(0.5)
device.enable_channel("mc", True)
device.enable_channel("mv", True)
```

Now we can start a recording, turn on the power and sleep for 10 seconds, and then turn off the power and stop the recording.

```
project.start_recording()
device.set_main(True)

time.sleep(10.0)

device.set_main(False)
project.stop_recording()
```

And finally we disconnect from Otii TCP Server.

```
otii.disconnect()
```

And that's it, a short script to automate recordings.

You find more examples on the Qoitech Github.

18 C# scripting

ⓘ These features require an Automation Toolbox license.

Otii has a build in TCP Server that can be used to control the application from another application, e.g. from a script running in your C.I. environment.

You can write your own TCP client using the Otii TCP Server API, or use one of the clients available at the Qoitech Github.

In this guide we will show you how to control Otii from a C# application using the Otii TCP Client for C#.

18.1 Installing the C# client

The Otii TCP Client for C# is avaliable as a NuGet package from nuget.org.

In Visual Studio, create a C#/.NET Framework project, right click on References in the Solution Explorer, and select Manage NuGet Packages....

In the NuGet Package Manager install OtiiTcpClient from the package source nuget.org. (If nuget.org not is listed as a package source, see here how to fix it.

18.2 Start the TCP Server

You need to start the Otii TCP server before connecting to it.

You can find out more about how to get started with the TCP server here:

- Automation Toolbox

18.3 Getting started

Make sure you have Otii running with the TCP server started, and an Arc connected to the computer without any devices connected to its main output.

We start with adding code to turn on and off the power of the Arc.

```
using System;
using System.Threading;
using Otii;

namespace Test {
    class Program {
        static void Main(string[] args) {
```

```
            // Calling Connect without parameters will connect to a local i
            var client = new OtiiClient();
            client.Connect();

            // Create a local reference to the Otii property for convenienc
            var otii = client.Otii;

            // Get a list of all Otii devices available
            var devices = otii.GetDevices();
            if (devices.Length == 0) {
                throw new Exception("No available devices");
            }

            // Get a reference to the first device in the list
            var arc = devices[0];

            // Turn on and off the power
            arc.SetMain(true);
            Thread.Sleep(2000);
            arc.SetMain(false);

            // Close the connection
            client.Close();
        }
    }
}
```

Run the program, and the Arc connected to your computer should turn on the power, and after a second turn it off again.

18.4 Handling projects

Before starting a measurement tou need to either create a project, connect to an already opened project, or open a previously saved project.

Here we connect to an open project if it exists, otherwise we create a new one.

```
            ...

            // Get a reference to the first device in the list
            var arc = devices[0];

            var project = otii.GetActiveProject();
            if (project == null) {
```

```
        project = otii.CreateProject();
    }

    ...
```

18.5 Configuring the project

Before measuring, we set the main output voltage, overcurrent protection, and we confiure and enable the UART.

Then we add the channels we want to records. In this case we enable Main Current (mc), Main Voltage (mv), UART logs (rx) and the digital input GPI1 (i1).

If you hover above the EnableChannel method in Visual Studio you can see what channels are available.

```
    ...

    var project = otii.GetActiveProject();
    if (project == null) {
        project = otii.CreateProject();
    }

    // Configuration
    arc.SetMainVoltage(3.3);
    arc.SetMaxCurrent(0.5);
    arc.EnableUart(true);
    arc.SetUartBaudrate(115200);
    arc.EnableChannel("mc", true);
    arc.EnableChannel("mv", true);
    arc.EnableChannel("rx", true);
    arc.EnableChannel("i1", true);

    ...
```

18.6 Do a timed recording

Replace the code in the section “Turn on and off the power” with the section Record shown below. You also need to add an constant in the main class defining the recording time.

```
    private const int RecordingTime = 30000;

    static void Main(string[] args)
    {
```

```
            ...

            // Record
            project.StartRecording();
            arc.SetMain(true);
            Thread.Sleep(RecordingTime);
            arc.SetMain(false);
            project.StopRecording();

            // Close the connection
            client.Close();
        }
```

You can find the Basic Measurment example at the Qoitech GitHub.

19 Jenkins integration

ⓘ These features require an Automation Toolbox license.

If you have a valid license for the automation toolbox you can enable a TCP server in Otii, making it possible to control Otii from another application. Using this feature you could e.g. use Otii in a continuous integration environment to automatically keep track of how firmware changes affects the energy consumption.

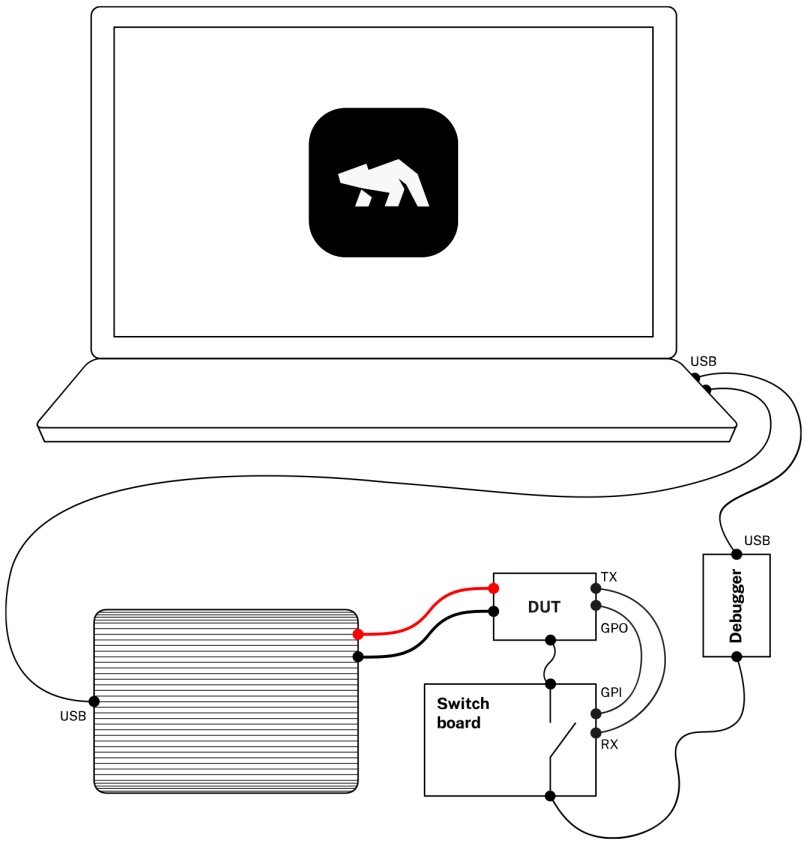
The following example shows how to add a test job that uses Otii in combination with Jenkins to make sure that a firmware change doesn't affect the energy consumption in a negative way.

19.1 Our test system

In this example we are using a ST32 Cortex M4 programmed with a ST-Link debugger. The board is powered by the Otii, and is connected to the ST-Link using the SWD interface to make it possible to flash the device with new firmware. To get a realistic measurement of the energy consumed, the ST-Link needs to be disconnected during the actual measurement.

For this reason we have developed a simple switch board that is connected to the expansion port of the Otii Arc, and is controlled by the GPO of the Arc. This makes it possible to connect the SWD when flashing the device, and then disconnect it when doing the energy measurements.

The device connects to the RX and the GPI1 of the Otii Arc. These are used in this example to mark the start and stop of parts of the measurements we want to verify.



19.2 The Python Otii Client

You need to install the Otii python client first, read more about it in the section Scripting with Python.

19.3 A first start

Create a test python script named otii_test.py. We will use the built in python unit test framework unittest to run our tests. Add the following code to otii_test.py:

```
#!/usr/bin/env python3
import unittest

class OtiiTest(unittest.TestCase):
    def test_energy_consumption(self):
        pass

if __name__ == '__main__':
    import xmlrunner
    unittest.main(testRunner=xmlrunner.XMLTestRunner(output='test-reports')
```

19.4 Running the test script

Make the script executable:

```
chmod a+x otii_test.py`
```

And run the script:

```
./otii_test.py

Running tests...
----------------------------------------------------------------------
.
----------------------------------------------------------------------
Ran 1 test in 0.000s

OK
```

19.5 Configure the Otii TCP Server

See the TCP Server for information about how to configure and start the TCP server, either using the Otii desktop client, or using the Otii command line tool.

If you want to automatically start the server from your test script, read more in the section Python scripting.

19.6 Connecting to the Otii TCP Server

The next step is to connect to the Otii TCP server from the test script:

```python
#!/usr/bin/env python3
import sys
import unittest
from otii_tcp_client import otii_connection, otii as otii_application

HOSTNAME = '127.0.0.1'
PORT = 1905

class OtiiTest(unittest.TestCase):
    def test_energy_consumption(self):
        connection = otii_connection.OtiiConnection(HOSTNAME, PORT)
        connect_response = connection.connect_to_server()
        if connect_response["type"] == "error":
            print("Exit! Error code: " + connect_response["errorcode"] + ",
```

```
            sys.exit()
        otii = otii_application.Otii(connection)

if __name__ == '__main__':
    import xmlrunner
    unittest.main(testRunner=xmlrunner.XMLTestRunner(output='test-reports')
```

19.7 Configuring the Otii Arc

Now we query Otii for all the available devices, and try to find the correct Arc to use.

By giving each Arc a unique name, you will be sure that you are using the correct one.

```
...
PORT = 1905
ARC_NAME = "TestArc1"
...
class OtiiTest(unittest.TestCase):
    def test_energy_consumption(self):
        ...
        devices = otii.get_devices()
        if len(devices) == 0:
            print("No Arc connected!")
            sys.exit()
        devices = [device for device in devices if device.name == ARC_NAME]
        if len(devices) != 1:
            print("Expected to find exactly 1 device named {0}, found {1} d
            sys.exit()
        arc = devices[0]

        arc.set_main_voltage(3.3)
        arc.set_exp_voltage(3.3)
        arc.set_max_current(0.5)
        arc.set_uart_baudrate(115200)
        arc.enable_uart(True)
        arc.enable_exp_port(True)
        arc.enable_5v(True)  # The switch board is powerd by the Otii +5V p
```

We configure the main out and the expansion port to 3.3 V, we set the max current to 500 mA and configure the UART to a baudrate of 115000, and then enable the UART and expansion port.

19.8 Measurement

Now we are ready to start a measurement.

```
...
import time
...
MEASUREMENT_DURATION = 5.0
...
class OtiiTest(unittest.TestCase):
    def test_energy_consumption(self):
        ...
        project = otii.get_active_project()

        arc.enable_channel("mc", True)
        arc.enable_channel("i1", True)
        arc.enable_channel("rx", True)
        arc.set_main(True)

        project.start_recording();
        time.sleep(MEASUREMENT_DURATION)
        project.stop_recording();

        arc.set_main(False)
```

19.9 Analyzing the result

The demo system using the UART to send out a start and stop indicator for a part where the system does a temperature measurements and it using the digital input to mark the interval of each temperature measurement.

19.10 Using RX to analyze data

In this system, the DUT sends the log message “Getting temperature” when starting a temperature measurement. We use two of these messages to extract the energy consumed for a complete cycle.

```
...
class OtiiTest(unittest.TestCase):
        ...
        recording = project.get_last_recording()
        index = 0
        count = recording.get_channel_data_count(arc.id, "rx")
```

```
        data = recording.get_channel_data(arc.id, "rx", index, count)
        values = data["values"]
        timestamps = [value["timestamp"] for value in values if value["valu
        self.assertGreaterEqual(len(timestamps), 2, "Need at least two \"Ge

        statistics = recording.get_channel_statistics(arc.id, 'mc', timesta
        self.assertLess(statistics["energy"], 0.0004, "One interval consume
        self.assertGreater(statistics["energy"], 0.0002, "One interval cons
```

19.11 Using GPI1 to analyze data

In this system, the DUT sends a small pulse when it wakes up to start a temperature measurement. We try to find the first two pulses, and extracts the energy consumed for this interval.

```
...
class OtiiTest(unittest.TestCase):
        ...
        index = 0
        count = recording.get_channel_data_count(arc.id, "i1")
        gpi1_data = recording.get_channel_data(arc.id, "i1", index, count)[
        timestamps = [gpi1_value["timestamp"] for gpi1_value in gpi1_data]
        self.assertGreaterEqual(len(timestamps), 4, "Need at least four GPI

        statistics = recording.get_channel_statistics(arc.id, 'mc', timesta
        self.assertLess(statistics["energy"], 0.0004, "One interval consume
        self.assertGreater(statistics["energy"], 0.0002, "One interval cons
```

19.12 Building and uploading the firmware

This Jenkins job is going to be triggered when new firmware is merged to the master branch of the code repository. The first thing we need to do is to build the firmware, and then upload the new firmware to the device. We only need to do this once, so we add the class method setUpClass to the OtiiTest class, this will be called once before running the unit tests.

Since we have a switch board that is used to enable the debugger that is controlled by the Otii Arc, we have to move the connection and setup part from the test to this method. We make otii and arc class atrributes, making them accessible by all tests in this class.

```
...
class OtiiTest(unittest.TestCase):
    otii = None
    arc = None
```

```
@classmethod
def setUpClass(cls):
    # Connecting to Otii and setup Arc
    connection = otii_connection.OtiiConnection(HOSTNAME, PORT)
    ...
    arc.enable_5v(True)  # The switch board is powered by the Otii +5V

    # Turn on the main power, and give the DUT time to startup.
    arc.set_main(True)
    time.sleep(1.0)

    # Enable the USB, and give the ST-lINK time to startup
    arc.set_gpo(1, True)
    time.sleep(3.0)

    # Upload new firmware
    result = subprocess.call("cd ../firmware; make; make upload", shell
    if result != 0:
        print("Failed to upload new firmware")
        sys.exit()
    time.sleep(3.0)

    # Disable the USB, and turn off the main power
    arc.set_gpo(1, False)
    time.sleep(1.0)
    arc.set_main(False);

    def test_energy_consumption
    ...
```

20 User account

You need to create an acount at www.qoitech.com to:

- Download the software
- Buy a Otii Toolbox license
- Manage your existing licenses
- Manage your products
- Create a support case

20.1 Register account

You can create a new account at www.qoitech.com/register.

Fill in your user information, accept the terms of use, and click on *Create*.

A verification code will be sent to the mail address you provided.

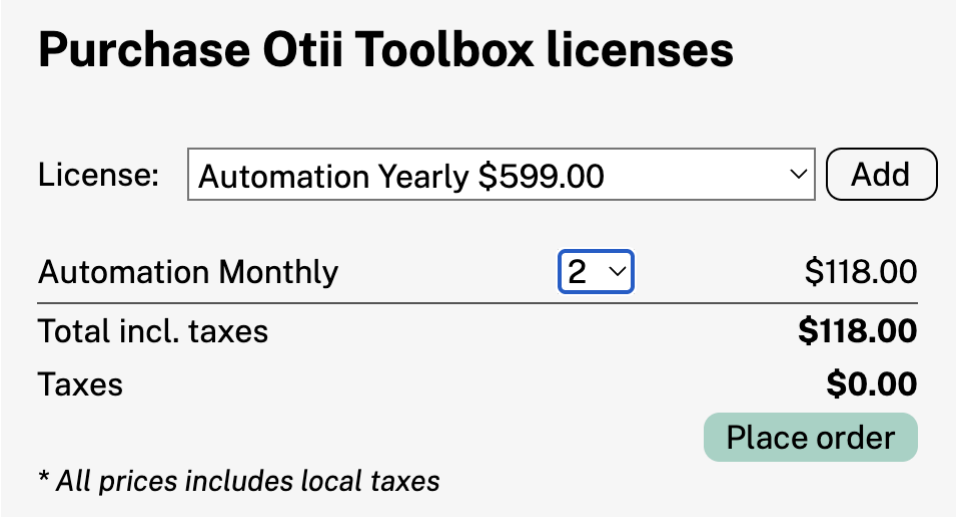
20.2 Forgot password

If you forgot your password, you can request a password reset at www.qoitech.com/forgotpassword.

21 Purchase licenses

You can purchase new licenses at www.qoitech.com/licenses.

⚠ All prices includes local taxes



To buy new licenses, you choose the license you want from the **License** dropdown, and press *Add*. You can more than one license.

To update the number of licenses you want to buy, you use the dropdown box shown after each license.

ⓘ The license count is the number of licenses, not the number of years or months.

When you press *Place order*, you will be redirected to our payment processor Fastspring.

22 Manage licenses

You can manage your existing licenses at www.qoitech.com/licenses.

22.1 Move to user

To move your license to another user, you press *Move to user* and enter the user name.

This will move the access to the licens to this user, but not change any of the payment information.

22.2 Share with user

The license can be share to one or more users. Only one user at a time can reserve a license.

22.3 Return license

If it for some reasons not possible to return the license from the software, you can return the license here.

23 Legal

23.1 Product safety

Download the product safety information available in English, German, French, Spanish and Italian.

- Product Safety Information for Otii Arc Pro
- Product Safety Information for Otii Ace Pro

23.2 Liability disclaimer

Qoitech AB reserves the right to make changes to the desktop application without further notice, to improve reliability, function and/or design. Any liability arising out of the desktop application and its usage is herein not assumed by Qoitech AB.