Moku Data Logger

User Manual

LIQUID INSTRUMENTS

Table of Contents

Introduction

The Moku Data Logger is a versatile instrument designed for continuous data capture and logging. It supports multiple independent channels with configurable acquisition rates, making it suitable for a wide range of applications, including environmental monitoring and sensor data recording. The Data Logger can write data to internal memory, a solid-state drive, or an SD card, depending on the hardware platform. It offers flexible export options, allowing users to export data directly to a computer or cloud-based storage in various formats, such as CSV or MATLAB files. The instrument also features an embedded Waveform Generator, enabling users to generate signals on up to four output channels. Advanced scheduling and triggered start capabilities allow for long-term measurements, limited only by the storage capacity.

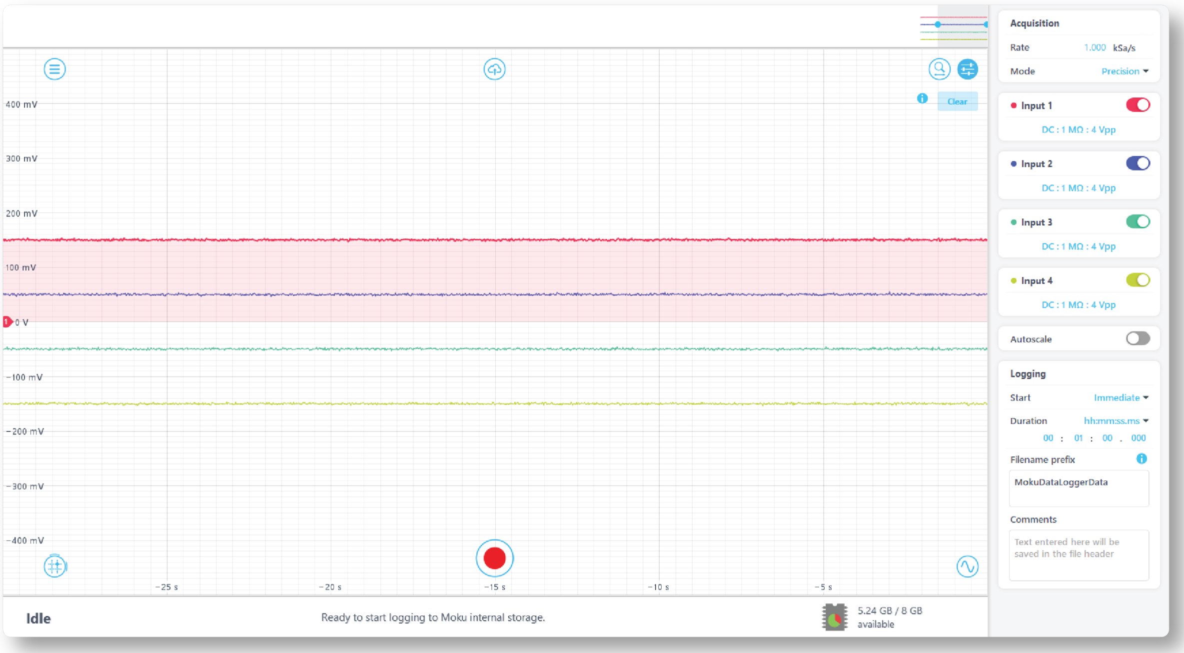


Figure 1. Moku Data Logger user interface showing signal display area (left) and settings panel (right).

These user manuals are tailored to the graphical interfaces available on macOS, Windows, iPadOS, and visionOS. If you'd prefer to automate your application, you can use Moku API; available for Python, MATLAB, LabVIEW, and more. Refer to the API Reference to get started.

AI-powered help is available to aid both workflows. AI help is built into the Moku application, and provides fast, intelligent answers to your questions, whether you're configuring instruments or troubleshooting setups. It draws from Moku manuals, the Liquid Instruments Knowledge Base, and more, so you can skip the datasheets and get straight to the solution.

Access AI help from the main menu

For more information on the specifications for each Moku hardware, please refer to our Product Documentation, where you can find the Specifications and the Data Logger Datasheets.

Quick start guide

Here we outline how to log data, download the file, and convert it to a CSV in order to highlight the typical workflow of a measurement in the Data Logger.

Figure 2. Data Logger interface and signals for the quick start guide example

In this example we will log 5 seconds of data from four input signals, download the data and convert it to a CSV file to use for post-processing. The input signals will be a sine, square, ramp and pulse wave. These signals will have amplitudes 1 Vpp, 2 Vpp, 500 mVpp, and 200 mVpp, respectively, with frequencies 1 Hz, 2 Hz, 500 mHz, and 1 Hz, and the pulse wave will have an offset of 100 mV.

- **Step 1: Configure the analog front end settings for the inputs.**
  - Open the side-panel by clicking the settings icon .
  - Enable the input signals and set the analog front end settings for the inputs. In this case, all inputs have a 1 MΩ input impedance, 4 Vpp attenuation to accommodate the largest signal, the square wave of amplitude 2 Vpp, and use DC coupling to observe to offsets.
- **Step 2: Configure the acquisition settings.**
  - Set the acquisition rate to 1 MSa/s.
  - Set the acquisition mode to "Precision".
- **Step 3: Generate signals with the embedded Waveform Generator.**
  - Generate your output signal in the Waveform Generator embedded in the Data Logger. Generate a 1 Vpp, 10 kHz, sine wave.
- **Step 4: Monitor the signals.**
  - Clear the history , toggle on the "Autoscale", and adjust the span to -12 seconds using the preview bar at the top of the interface.
- **Step 5: Enter the logging settings.**
  - Set the logging to start immediately for a duration of 5 seconds.
  - Set the filename prefix to "UserManualExample" and the comments to "This is the quick start guide example".

- **Step 6: Start the logging session.**
  - Start the logging session by clicking the start logging button .
  - Wait for the logging session to complete, monitor the signals and status of the run using the graphing window and the status bar.

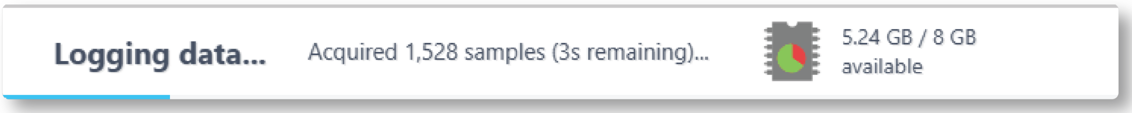


Figure 3. Status bar: Logging data...

- **Step 7: Export data.**
  - Once the logging session has successfully completed,

Figure 4. Status bar: Completed

- Open the File manager .
- Select the file named *UserManualExample_YYYYMMDD_HHMMSS.li*,
- Select convert to CSV,
- Select "Browse..." and select the appropriate local directory,
- Select "Download" and wait for the download and convert to complete.

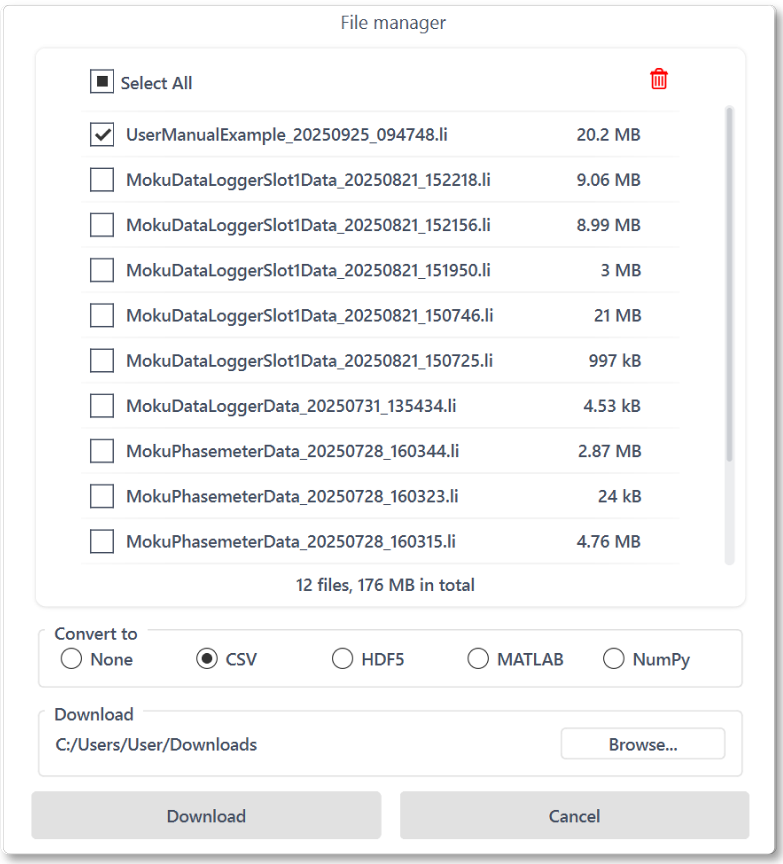


Figure 5. Export data with the file manager

- **Step 8: Open the data.**
  - Open the converted data in your preferred editor to perform your analysis or visualization.

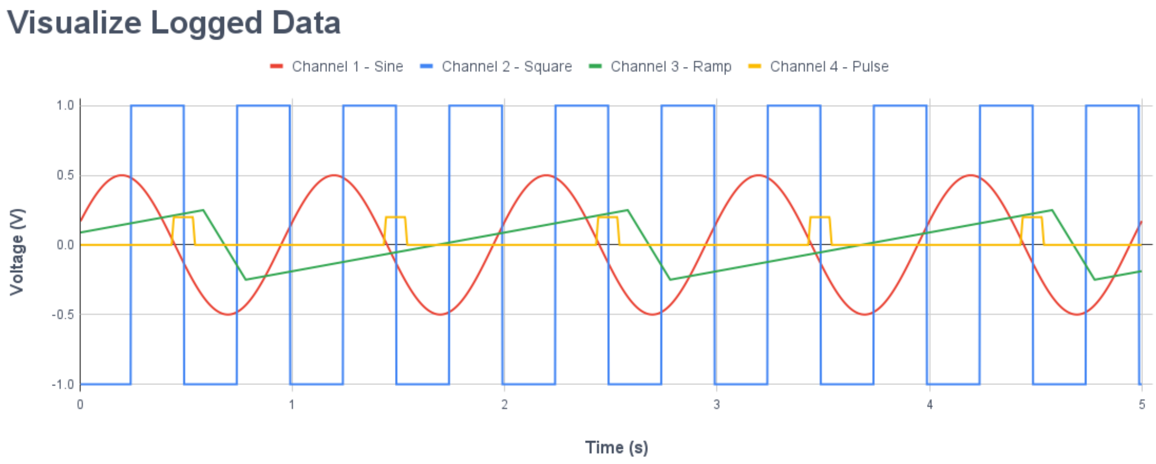


Figure 6. Plot of the logged data

Principle of operation

The Moku Data Logger is used for capturing and logging continuous data with no gap in a single session. The log duration and acquisition mode can be adjusted to best fit the logging requirements.

Acquisition modes

The Moku Data Logger supports different acquisition modes to suit various signal analysis needs.

Normal mode: This mode performs direct downsampling by discarding extra data points. It is suitable for signals with components at multiple frequencies or as a general-purpose mode. Normal mode, also called “Direct Downsampling” or “DDS”, selects every nth data point in the downsampling window. For example, if the Analog-to-Digital Converter (ADC) has over-sampled 100 points, and only needs one point to reconstruct the signal, then Normal mode will take every 100th sampled point. This can cause a high frequency input signal to alias (appear as a different frequency), but if you have a signal present, there's almost always something shown on the screen which can assist in debugging (in contrast with precision mode below).

Precision mode: This mode samples at the full rate and averages excess data points to improve precision. It is ideal for small or noisy signals that can benefit from improved resolution. Precision mode takes the average of all points in the downsampling window. This operation can remove noise and improves the voltage resolution of the measurement. If the signal is too fast for the selected acquisition rate, Precision mode may average the signal down close to zero, making it appear as if the signal is not present. Precision mode is not suitable for capturing fast glitches or transients because of the averaging it performs on the downsampling window.

Peak Detect mode: This mode is used to capture fast glitches or transients in the signal. For each channel that has been logged, there will be two columns of data, the maximum and minimum values in each sampling window are recorded. This appears as a shaded area in the graphing interface, extending from the minimum value to the maximum value in the downsampling window. This means that it will capture any glitches or transients present in the input but will also capture all noise which can make it more difficult to see the signal of interest. Peak Detect not only improves the visibility of small glitches but makes it easier to trigger on them as well.

Note: Peak Detect mode is not available on Moku:Lab.

Choosing between the Oscilloscope and Data Logger

The Moku Oscilloscope is primarily used to capture snapshots of fast signals and transient waveforms. The instrument captures and displays a segment of the signal once it's triggered. The Oscilloscope can capture very fast features, but the collected traces are not continuous. The maximum sampling rate for the Oscilloscope is higher than the Data Logger but is automatically set based on time span. The Moku Data Logger is primarily used to capture and log continuous data at a lower sampling rate than the Oscilloscope. There is no gap within a single logging session, and the logging duration and sampling rate can also be configured.

Consider using the Oscilloscope when quick, high-speed snapshots are required, and opt for the Data Logger when continuous, lower-speed recording is needed.

Logged data

Logged data is always saved in the LI file format. This is a compressed format that is optimized for speed and size. It can be converted to CSV or MAT format with our converter tool. Alternatively, it can also be converted from binary format to CSV, MAT, NPY, or HDF5 in the file manager of the Moku app.

Data that has been recorded in the Data Logger can be exported from the icon at the top of the interface, or from the file manager in the main instrument menu. From this window, you can download files that have been saved on to the device, and optionally convert them to CSV, HDF5, MATLAB, or Numpy formats.

Using the instrument

The controls options can be accessed by clicking the icon, allowing you to reveal or hide the control drawer. The controls drawer gives you access to channels, acquisition, and logging settings.

Display

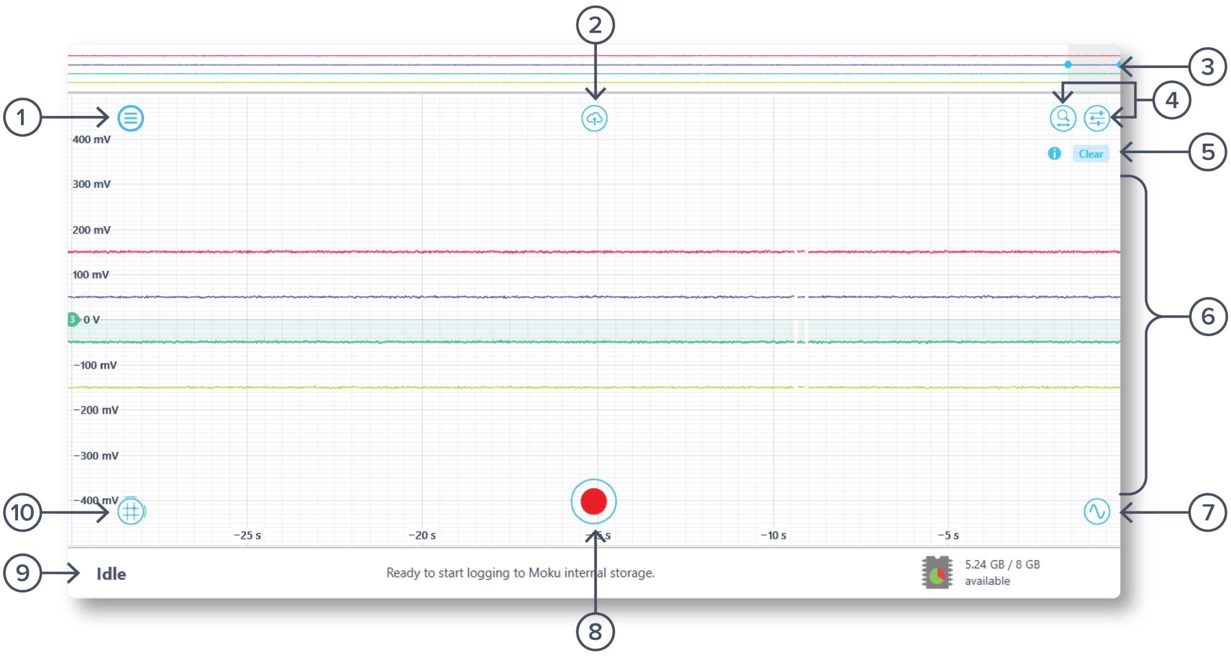


Figure 7. Data Logger display graph

① Main menu
② Export saved data
③ Preview pane
④ Magnifying glass and toggle side panel
⑤ Clear history
⑥ Display graph
⑦ Output waveform settings
⑧ Start logging
⑨ Status bar
⑩ Cursors

Clear

To clear the history of the displayed traces from the graphing area and preview bar, select Clear, or right-click and select "Clear history" from the context menu.

Autoscale

Toggle continuous autoscaling on/off. Continually track and scale the vertical axis of all visible input channels to remain on the display.

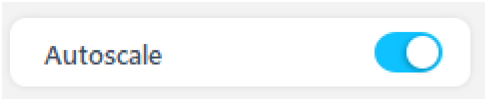


Figure 8. Autoscale traces

On the desktop app, this can be toggled from the context menu, by right-clicking the graph and selecting "Always autoscale". To manually resize waveforms to scale the vertical axis of all visible input channels to remain on the display, double-click the grid in the graphing area, or right-click and select "Autoscale traces".

On the iPad app, this can be toggled from the context menu, by pressing and holding the graph and selecting "Always autoscale". To manually resize waveforms to scale the vertical axis of all visible input channels to remain on the display, double-tap the grid in the graphing area, or press-and-hold and select "Autoscale traces".

Magnifying glass

Figure 9. Magnifying glass

Vertical scale

The vertical scale can be controlled using pinch and pan actions (on iPad) or click and scroll (on desktop) in the graph area while the magnifying glass is set to "Vertical zoom" .

Horizontal scale

The horizontal scale can be controlled using pinch and pan actions (on iPad) or click and scroll (on desktop) in the graph area while the magnifying glass is set to "Horizontal zoom" .

Rubber-band zoom

Both the vertical and horizontal axes can be set simultaneously, allowing you to select the area you would like to zoom to by selecting "Rubber-band zoom" and drawing a rectangle around the area.

Preview pane

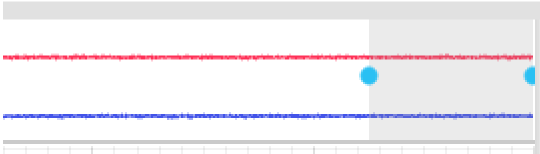


Figure 10. Data Logger preview pane

The span and time offset can be controlled using pinch and pan actions (on iPad) or click and scroll (on desktop) in the graph area while the magnifying glass is set to "Horizontal zoom" . The span and offset can also be controlled by the preview pane. Click and drag the highlighted pane to adjust the offset and click and drag the edge of the highlighted pane to adjust the span.

Reference trace

Capturing a reference trace can be utilised in applications where a comparison of a signal to a baseline is required. This helps to identify changes or deviations in the signal, which is particularly useful in testing. A reference trace can aid in visualizing changes in noise levels, amplitude variations, and frequency deviations.

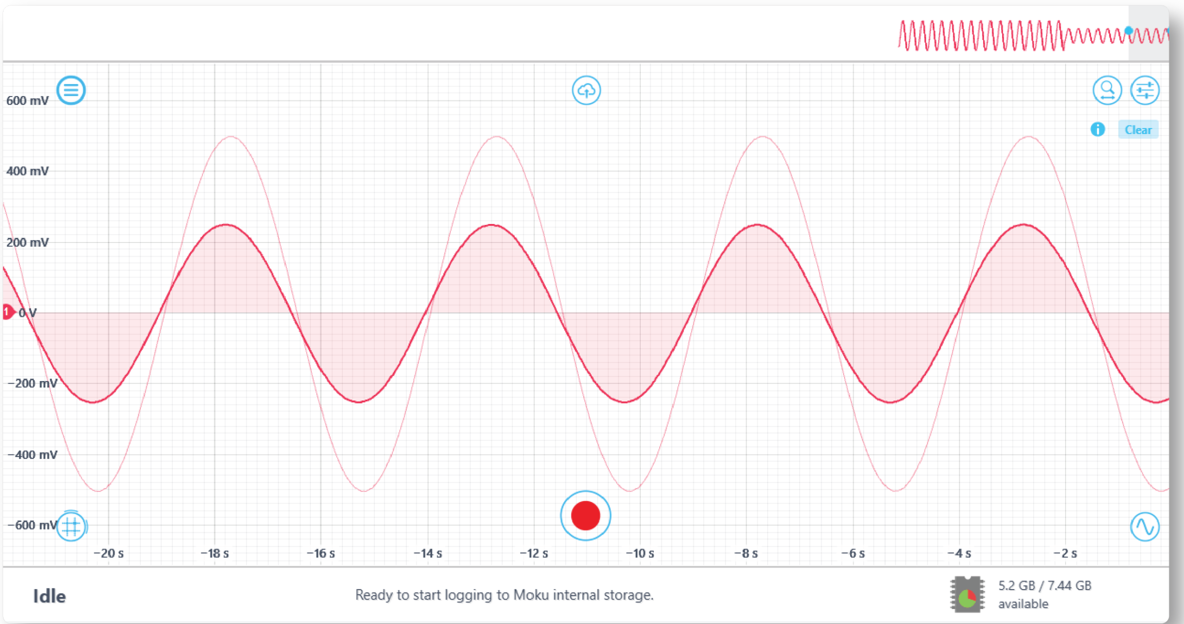


Figure 11. Data Logger reference trace example

To capture a reference trace, right-click on the trace in the graphing window, then select "Capture reference trace". Each trace can be updated at any time by right-clicking on the trace and selecting "Update reference trace". To remove the reference traces, either right-click on the trace and select "Remove reference trace" to remove an individual reference trace or right-click anywhere in the graphing area and select "Remove all reference traces".

Cursors

The cursors can be accessed by clicking the icon, allowing you to add voltage or time cursors on the display. The color of the cursor represents the channel of the measurement

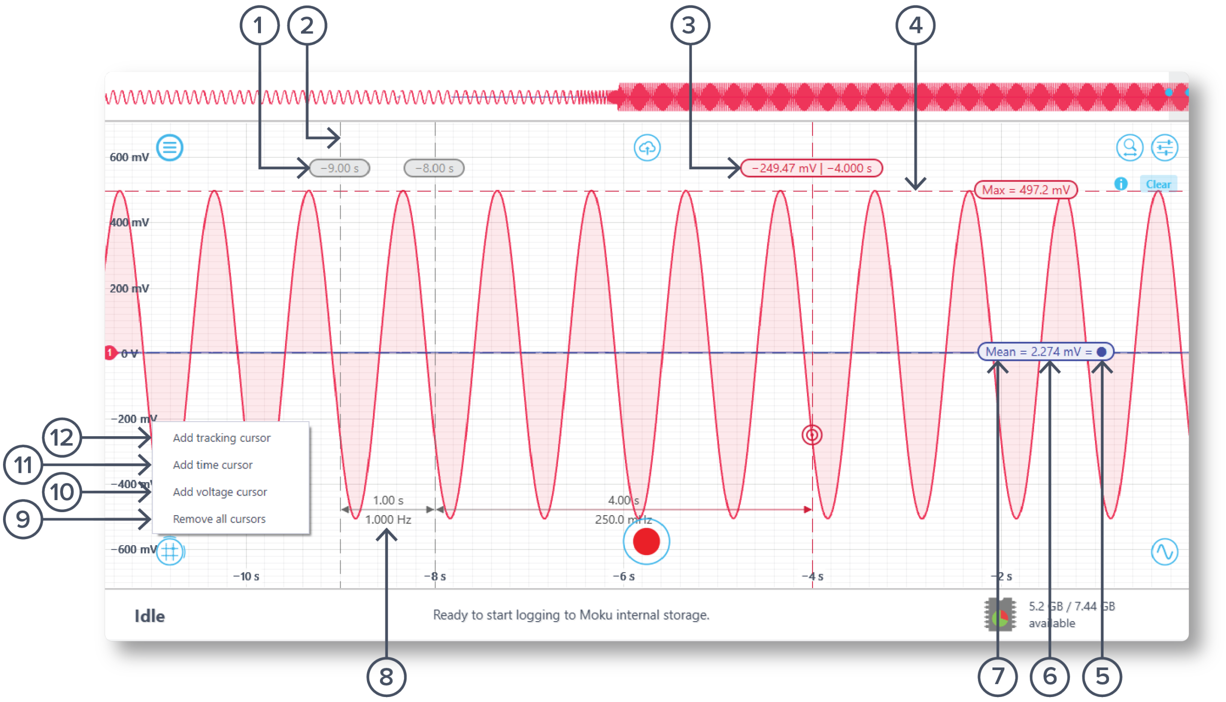


Figure 12. Cursor actions and context menu.

① **Time reading** Click to enter time, right-click to reveal time cursor options.
② **Time cursor** Drag left or right to set position.
③ **Tracking cursor** Drag left or right to measure the voltage at a given time.
④ **Voltage cursor** Drag up or down to set position.
⑤ **Reference indicator** Indicates the cursor is set as reference.
⑥ **Amplitude reading** Click to enter amplitude, right-click to reveal amplitude cursor options.
⑦ **Cursor function** Indicates the current cursor function (max, min, max hold, etc).
⑧ **Time difference** Represents the time difference between two cursors. This will show up automatically when you have two time cursors placed.
⑨ **Remove all cursors** Click to remove all voltage and time cursors.
⑩ **Add voltage cursor** Click to add a cursor measuring vertical position.
⑪ **Add time cursor** Click to add a cursor measuring horizontal position.
⑫ **Add tracking cursor** Click to add a cursor measuring the horizontal position, time, whilst tracking the vertical position, voltage value, at that time.

Time cursor

Select the cursor reading to enter a time value, right-click the cursor reading to reveal additional time cursor options.

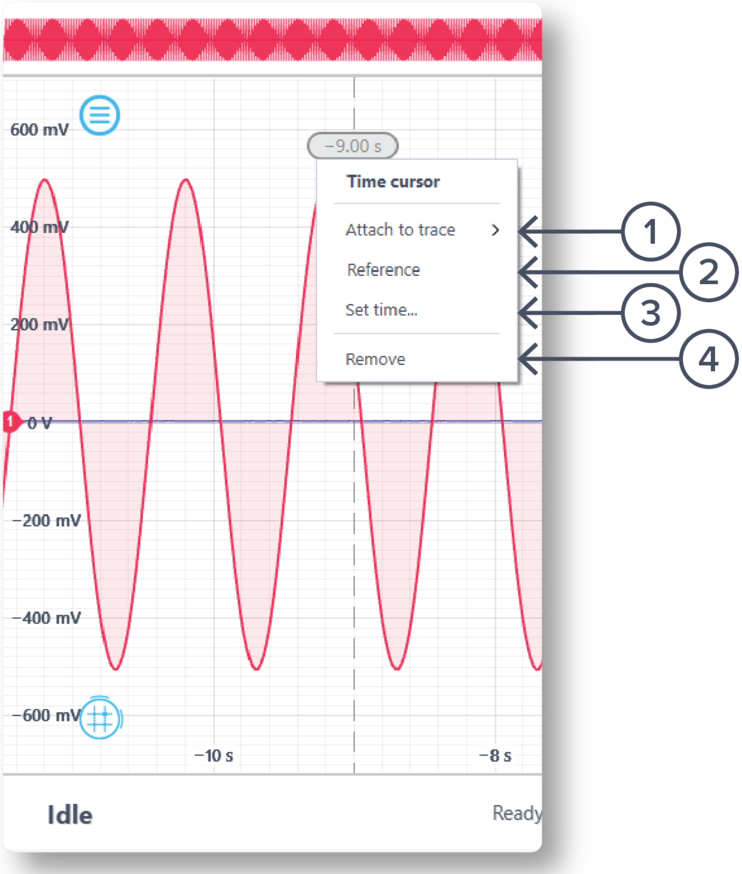


Figure 13. Time cursors and context menu

① **Attach to trace** Create a tracking cursor that displays the voltage of the selected trace at the specified time

② **Reference** Click to set the cursor to act as a horizontal reference value. When this option is selected all other cursors will display the difference between the cursor and the reference cursor's value.

③ **Set time** Click to position the cursor at the time entered.

④ **Remove** Click to remove the cursor from display.

Voltage cursor

Select the cursor reading to enter a voltage value, right-click the cursor reading to reveal additional voltage cursor options.

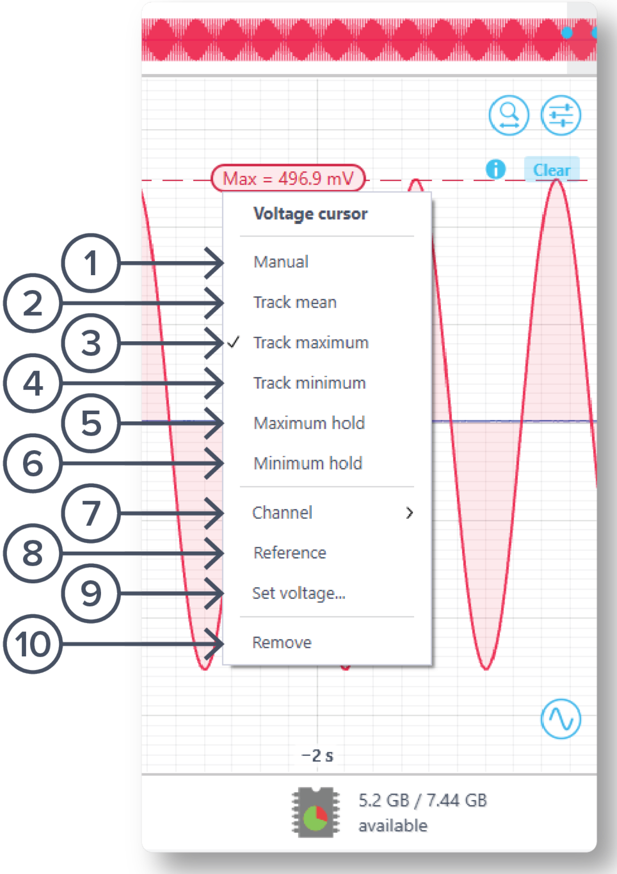


Figure 14. Voltage cursors and context menu.

① **Manual** Click to manually set the vertical position of the cursor.
② **Track mean** Click to set the cursor to track the mean voltage.
③ **Track maximum** Click to set the cursor to track the maximum voltage.
④ **Track minimum** Click to set the cursor to track the minimum voltage.
⑤ **Maximum hold** Click to set the cursor to the maximum voltage of previous traces.
⑥ **Minimum hold** Click to set the cursor to the minimum voltage of previous traces.
⑦ **Channel** Select to change the channel that the cursor measures.
⑧ **Reference** Click to set the cursor to act as a vertical reference value. When this option is selected, all other cursors will display the difference between the cursor and the reference cursor's value.
⑨ **Set voltage** Click to set the voltage cursor at a specific voltage.
⑩ **Remove** Click to remove the selected cursor.

Acquisition

The channel sub-panels allow you to toggle channels ON/OFF and change the analog frontend settings for each input. You can change the view and measurement of the input by changing the range, impedance, and coupling. The acquisition sub-panel allows you to set the acquisition rate and the acquisition mode.

Tip: The channel and acquisition sub-panels may be dragged into the main display by tap/hold and dragging on iPad.

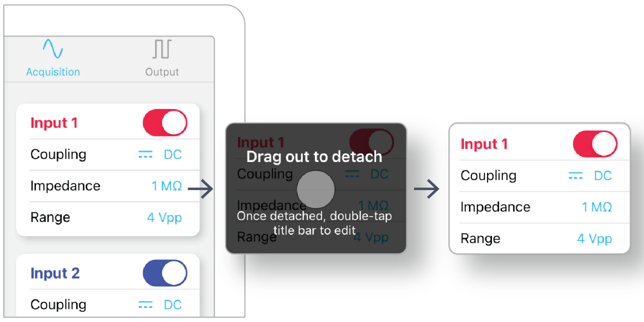


Figure 15. Drag acquisition side-panels

Analog front end settings

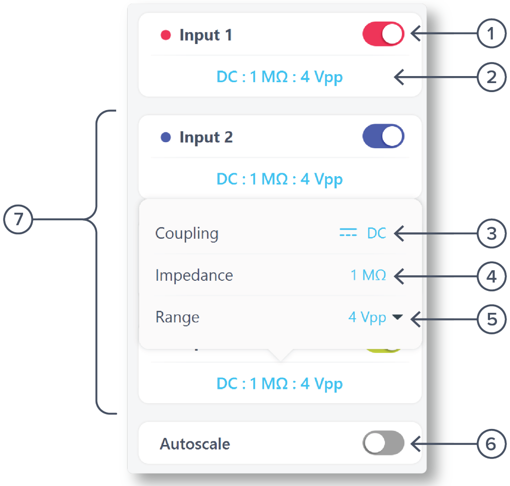


Figure 16. Data Logger front end settings

① Toggle to enable/disable the data for the input.
② Click to change the frontend settings of the input.
③ Select between AC and DC coupling of the input.
④ Select the input impedance (50 Ω* or 1 MΩ).
⑤ Select the input range of the input.
⑥ Toggle autoscaling.
⑦ Remaining inputs.
* Note: Moku:Go only supports 1 MΩ frontend impedance, it does not support 50 Ω

Coupling

Input coupling is a small circuit used to connect the signal to the instrument, this can be alternating current (AC) or direct current (DC) coupling. For most applications, DC coupling is the preferred option; this does not filter or modify the signal in any way. AC coupling is best used when your signal is relatively small compared to its DC offset, such as a small ripple along a DC power supply, so you can utilize the full range and resolution of the ADC.

- **DC coupling** shows all of the input signal, including any alternating waveforms as well as any DC offsets present in the measured signal.
- **AC coupling** will remove any DC offset present in the signal with a high-pass filter, showing only the signal's AC component. This can attenuate low frequency components in the process, such as square-wave-like signals and low frequency signals.

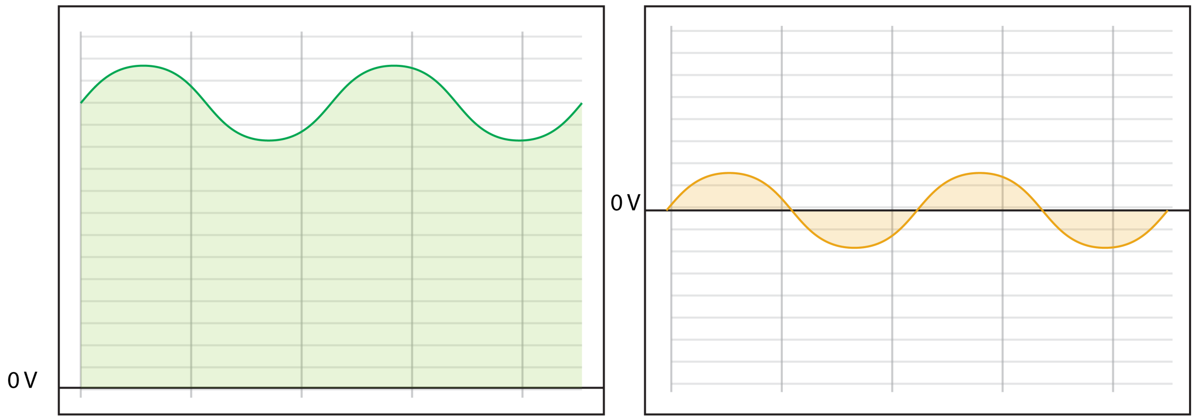


Figure 17. DC (left) and AC (right) coupled signals

Input impedance

Input impedance is effectively the resistance from the signal line to ground. For high-frequency measurements, it is important that Moku device's impedance matches that of the device under measurement. This is to maximize power transfer and minimize signal reflection. If the impedances are mismatched, some signal power will reflect off the input, interfering with the measurement.

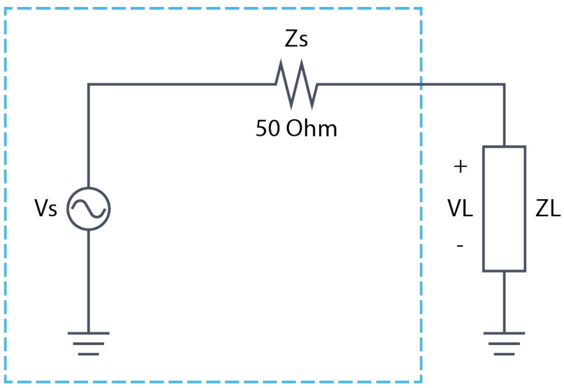


Figure 18. Impedance matching circuit

High-impedance is the default setting for many Data Loggers that are not targeted at RF frequencies, while 50 Ω is most common for RF devices and signal generators.

High-impedance mode (1 MΩ) is used when Moku is observing a signal that is externally terminated (e.g. between two high-speed devices), or when absolute voltage measurement accuracy is required and the signal is low-enough frequency that reflections from the measurement equipment do not significantly interfere with the original signal.

The input impedance also forms a resistive divider with the signal's source impedance, reducing the apparent voltage.

Note: Moku:Go only supports 1 MΩ frontend impedance, it does not support 50 Ω. It can be externally terminated using a 50 Ω terminator if necessary.

Range

The range settings are used to ensure the ADC utilizes all its bits of resolution, giving you the most accurate signal reconstruction at all times. This will apply attenuation to your signal so that it is always sending an appropriately scaled signal to the ADC, irrespective of the input signal amplitude.

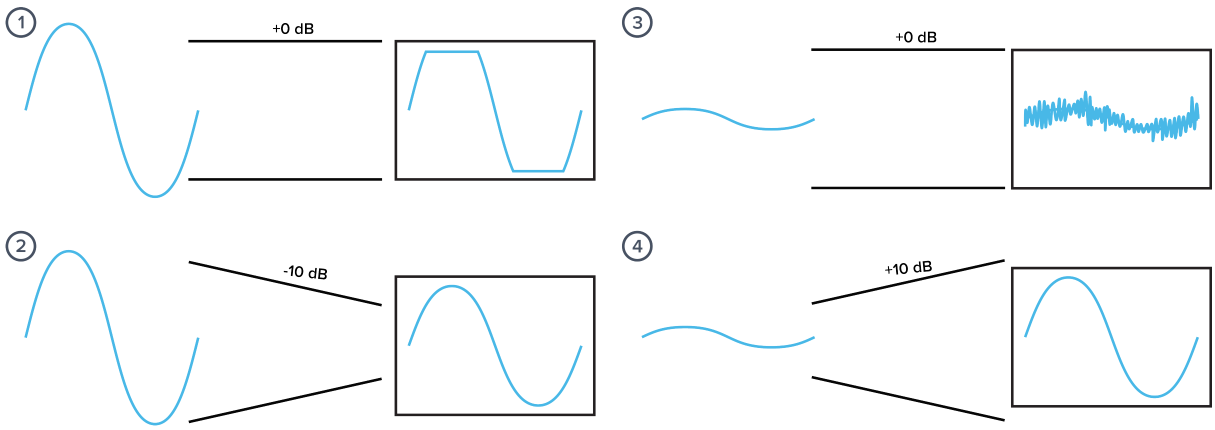


Figure 19. Range scaling diagram comparing an attenuated signal (top) and clipped signal (bottom)

For example, when you input a 1 Vpp signal into a Data Logger that utilizes a 400 mVpp ADC, this peak-to-peak amplitude of this signal needs to be attenuated down to 400 mV (divided by a factor of 2.5) so it is not clipped by the ADC, see Figure 19.

The range on the Data Logger can be set manually or automatically. If using "Auto", the Moku will select the range based on the signal extents at its current zoom level. To ensure the range is set properly in this case, double-click on the signal trace. This can sometimes lead to unstable behaviour if the signal amplitude is at one of these range crossovers. To manually set the input range, select the closest range value that is larger than the signal amplitude.

Acquisition settings

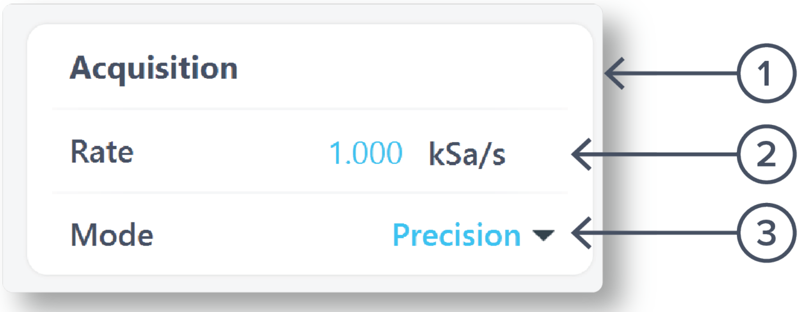


Figure 20. Acquisition settings

① Acquisition sub-panel
② Enter the desired acquisition rate or sampling rate
③ Select acquisition mode; precision, normal or peak detect

The acquisition rate refers to the speed at which the Data Logger samples and records data. This rate is measured in samples per second (Sa/s) and is a critical factor in determining the resolution and accuracy of the data captured.

The acquisition mode refers to the process of capturing the data and storing it in the Moku device's internal memory. This is always down-sampled, not up-sampled. See Principle of operation for more information on selecting the acquisition mode.

Logging

Configure the logging session in the logging sub-panel.

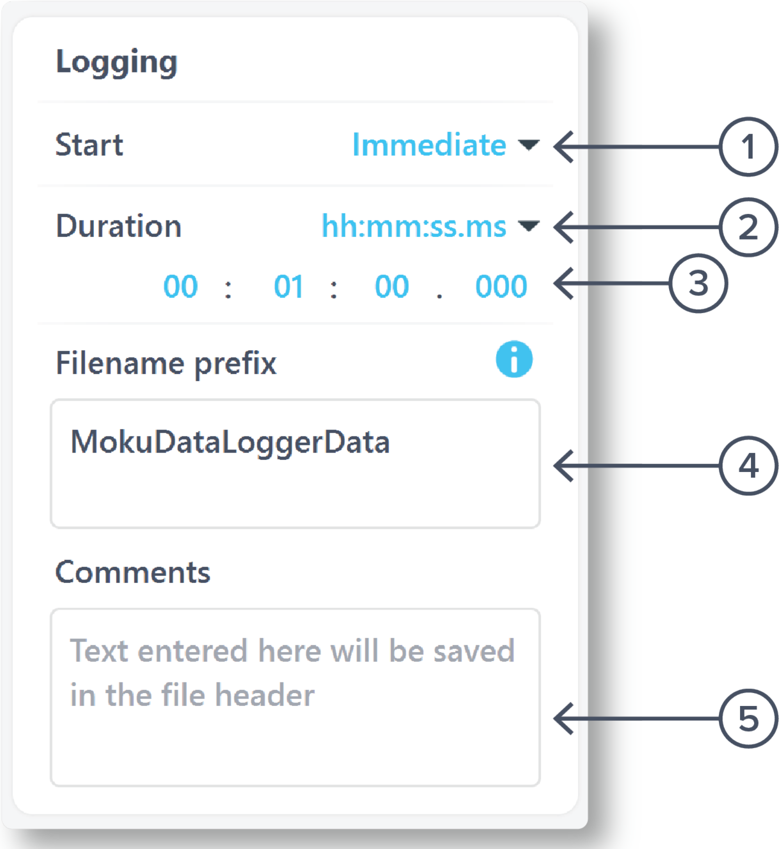


Figure 21. Data Logger control panel for configuring the logging session

① Select logging start condition; immediate, delayed or triggered
② Select duration format; hh:mm:ss.ms, hours, minutes, seconds, milliseconds
③ Enter the duration of logging session.
④ Enter a prefix for the log filename.
⑤ Enter any comments to be saved in the log file header.

Start modes

Select the logging start condition from Immediate, Delayed, or Triggered to begin the logging session.

Immediate

The logging session begins immediately after selecting ● and will log for the duration entered.

Delayed

After starting the logging session by selecting ●, the instrument will wait for the entered delay, then begin logging the data for the duration entered.

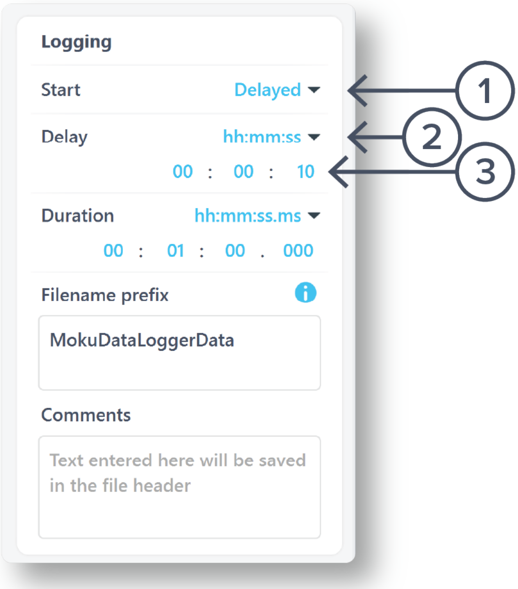


Figure 22. Delayed logging start condition

① Select delayed for the start condition.
② Select duration format; hh:mm:ss.ms, hours, minutes, seconds, milliseconds
③ Enter the duration of the delay.

Triggered

After starting the logging session by selecting ●, the instrument will wait for the trigger condition to occur to begin logging the data, then will log for the duration entered.

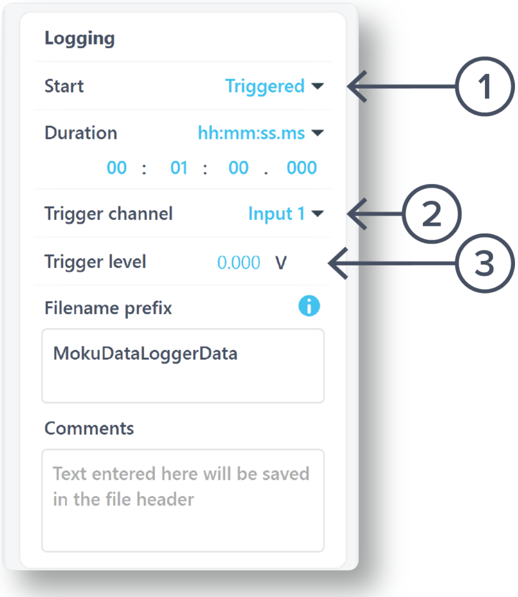


Figure 23. Triggered logging start condition

① Select triggered for the start condition.
② Select the channel from which the trigger condition signal will be found. Select the source for the trigger channel. This can be one of the Data Logger inputs, or external trigger input (if available) independent of what is displayed on the graphing interface.
③ Enter the trigger level. This sets a threshold voltage, such that when the signal crosses this value on a trigger edge or pulse it causes a trigger event and begins logging. The trigger watches for the input signal to cross the level threshold.

Duration

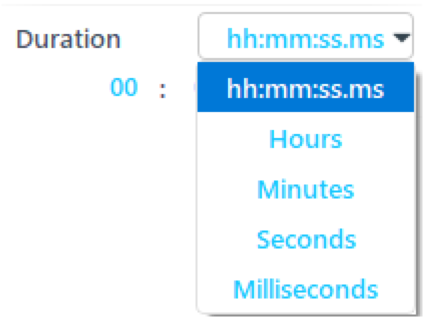


Figure 24. Time format options for the duration of the logging session

Select from the time formats displayed in Figure 24, then enter the desired duration to log the data for. The maximum duration will be limited by the storage available, the acquisition rate, and the acquisition mode.

Filename prefix and comments

These optional fields can prove useful when determining the difference between test runs. The filename prefix can provide a useful filtering option for logged data. The entered filename may contain any alphanumeric characters and underscores. A timestamp will be appended to ensure that the name is unique, for example: MokuDataLoggerData_20250321_161913.li. The comments will be saved within the heading of the file.

Status bar

The status bar of the Data Logger displays important information about the state of the Data Logger including the state, message, and storage.

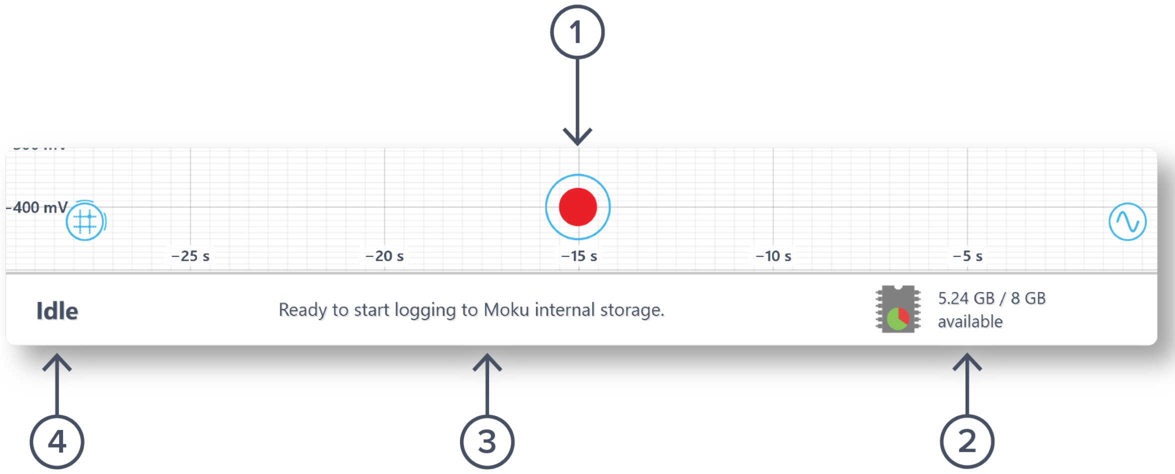


Figure 25. Data Logger status bar

① Start logging session icon
② Storage
③ Message
④ State

States

Idle	Before a logging session has started.
Logging data...	During the logging session.
Completed	After a logging session has successfully completed without interruption, this will be followed by a message with the number of samples and the file name, for example, *Successfully logged 10,000 samples to "MokuDataLoggerData_20250324_172053.li"*.
Starting...	The moment a logging session has been started by selecting the start logging icon.
Waiting...	Waiting for a delay to complete or a trigger condition to be met.
Stopping...	This state will occur while a logging session is interrupted by selecting the stop icon.
Cancelled	This state will occur after a logging session is interrupted by selecting the stop icon.
No SD card	No internal storage was detected.
Error	An overflow error occurred.

Messages

A completed log will be followed by a message with the number of samples and the file name, for example, *Successfully logged 10,000 samples to "MokuDataLoggerData_20250324_172053.li"*.

Idle	Ready to start logging to Moku internal storage.
No SD Card.	Please insert an SD card to start logging data.
Error	An overflow error occurred. This usually means that the storage is full, and the logging session exceeded the storage limits, open the File manager and clear out some space to log more data.
Completed	Successfully logged 10,000 samples to "MokuDataLoggerData_20250324_172053.li".
Logging data...	Acquired # samples (#s remaining).
Waiting...	Logging will begin in #s/when trigger occurs.

Storage

This shows the total storage and the available storage remaining. If it displays "no SD card", the external storage is not connected. If there is not an external storage option for the hardware type, this is an issue with the internal storage of the device.

Share data

To share and export data, open the File Manager with one of the following options:

1. Select the share button to open the file manager,
2. Select the main menu, scroll down, and select "File Manager...",
3. Right-click anywhere on the graphing window, select "Export data...".

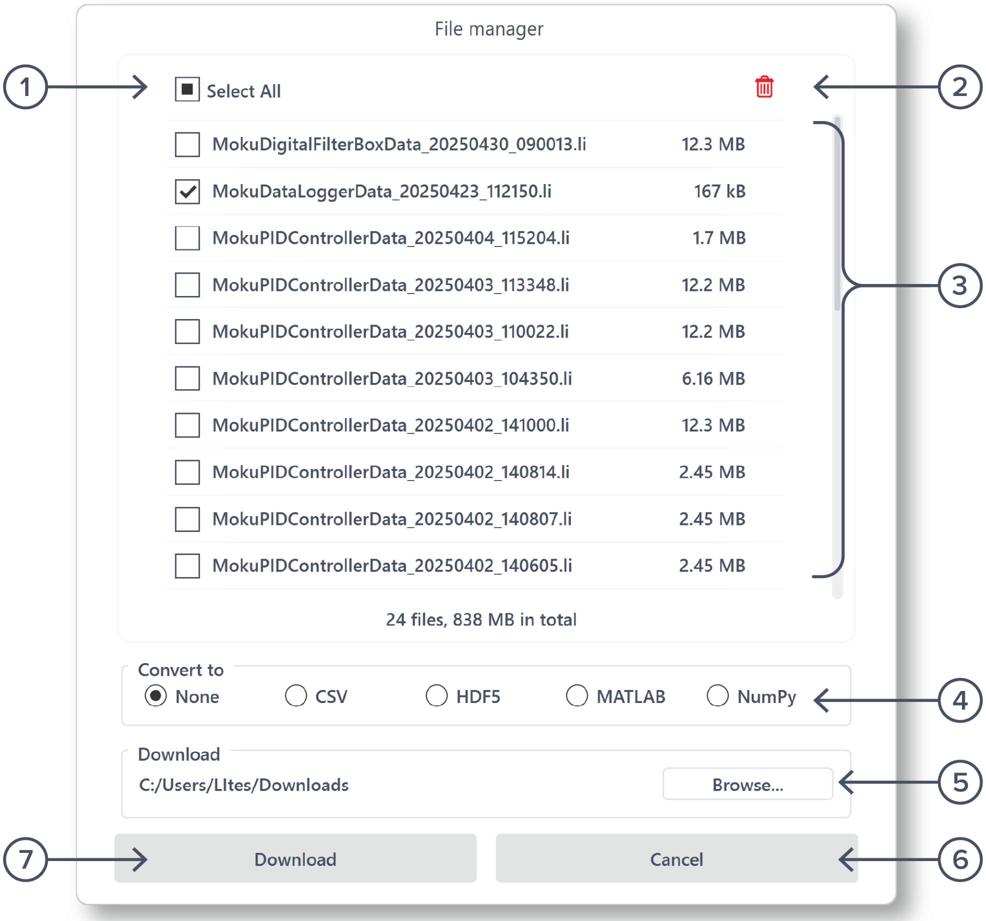


Figure 26. Download logged files with the file manager

① Select all files.
② Delete the selected file(s).
③ Select file(s) on the Moku to download or delete.
④ Convert the binary file(s) to the selected file type upon downloading.
⑤ Browse and select a local folder to download the selected file(s) to.
⑥ Click to exit the file manager.
⑦ Click to download the selected file(s).

Output settings

The Data Logger has an embedded Waveform Generator. It has all the same functions as the Moku Waveform Generator instrument, though does not support modulation.

Controls for the embedded Waveform Generator are found in the output side panel on iPad or revealed from the button in the graph area on the desktop app.

Figure 27. Data Logger output panel

① Click to reveal or hide output settings panel
② Click through available outputs
③ Enable / disable output
④ Select the waveform shape (sine, square, ramp, pulse, noise, or DC)
⑤ Toggle the output termination
⑥ Preview of the output waveform
⑦ Click to show context menu to sync the channels or copy settings across outputs
⑧ Select the units of displayed parameter
⑨ Enter the value of displayed parameter
⑩ Label of displayed parameter
⑪ Switch parameter representation
⑫ Click to configure parameter
⑬ Hide output settings panel

Examples

Streaming

In this example, we outline how to use the Python API to stream time and voltage data from the Moku Data Logger directly to your computer. In this example we will stream 10 seconds of data from four input signals and plot the signals in real-time. The input signals will be a sine, square, ramp and pulse wave, configured in Multi-instrument Mode. The equivalent set up is shown in the figure below.

Figure 28. Validating the streaming set up in the Moku desktop app

The script will follow the steps laid out below. For a full description of features and commands, visit the API documentation at apis.liquidinstruments.com/api/

- **Step 1: Connect to your device and configure Multi-instrument Mode**
  - Using your Moku IP address and the platform id of your hardware, connect to your Moku in Multi-instrument Mode.
  - Place the instruments in the slots; place the Waveform Generator in Slot 1 and place the Data Logger in Slot 2.
  - Connect the outputs of both instrument slots to each input of the Data Logger slot.
- **Step 2: Generate waveforms to observe in the Data Logger**
  - For each waveform, set the frequency to 20 Hz and the amplitude to 1 Vpp.

  - Generate a sine wave and a square wave with a 50% duty cycle in the Waveform Generator.
  - Generate a ramp wave with 50% symmetry and a pulse wave with 20 ms pulse width and 10 ms edge time in the Data Logger.
- **Step 3: Configure the Data Logger for streaming**
  - Enable each input of the Data Logger to observe each signal (this is the default setting).
  - Set the acquisition rate of the Data Logger to 10 kSa/s.
- **Step 4: Start the streaming session**
  - Set the duration of the stream to 10 seconds and start the streaming session.
  - Set up a plot to observe the streamed signals in real-time.
  - While the Data Logger is streaming, retrieve data and update the plot.
- **Step 5: Close off the session**
  - Once the streaming session has completed, disconnect from the Moku.

```
#
# moku example: User Manual Data Logger streaming
#
# This example demonstrates use of the Data Logger instrument to
# stream time-series voltage data and plot it using matplotlib
#
# (c) Liquid Instruments Pty. Ltd.
#
import matplotlib.pyplot as plt
from moku.instruments import MultiInstrument, Datalogger, WaveformGenerator

# Replace with your Moku IP address and platform id for your hardware
# force_connect will overtake an existing connection
mim = MultiInstrument('192.168.###.###', platform_id=4, force_connect=True)
wg = mim.set_instrument(1, WaveformGenerator)
i = mim.set_instrument(2, Datalogger)

try:
    # These connections are setup for the Moku:Pro, tailor to your hardware
    connections = [dict(source="Slot1OutA", destination="Slot2InA"),
                   dict(source="Slot1OutB", destination="Slot2InB"),
                   dict(source="Slot2OutA", destination="Slot2InC"),
                   dict(source="Slot2OutB", destination="Slot2InD")]
    mim.set_connections(connections=connections)

    # Generate waveforms on the output channels of the Waveform Generator
    wg.generate_waveform(1, "Sine", frequency=20, amplitude=1)
    wg.generate_waveform(2, "Square", frequency=20, amplitude=1, duty=50)

    # Generate waveforms on the output channels of the Data Logger
    i.generate_waveform(1, "Ramp", frequency=20, amplitude=1, symmetry=50)
    i.generate_waveform(2, "Pulse", frequency=20, amplitude=1,
                        pulse_width=20e-3, edge_time=10e-3)

    # Enable each input as we want to stream data from all inputs
    i.enable_input(1, enable=True)
    i.enable_input(2, enable=True)
    i.enable_input(3, enable=True)
    i.enable_input(4, enable=True)

    # set the sample rate to 10 kSa/s
    i.set_samplerate(10e3)

    # stream the data for 10 seconds..
    i.start_streaming(10)

    # Set up the plotting parameters and labels
    plt.ion()
    plt.show()
```

```
    plt.grid(visible=True)
    plt.ylim([-1, 1])
    plt.title("User Manual Streaming Example")
    plt.xlabel("Time (s)")
    plt.ylabel("Voltage (V)")

    line1, = plt.plot([])
    line2, = plt.plot([])
    line3, = plt.plot([])
    line4, = plt.plot([])

    # Configure labels for axes
    ax = plt.gca()

    # This loops continuously updates the plot with new data
    while True:
        # get the chunk of streamed data
        data = i.get_stream_data()
        if data:
            plt.xlim([data['time'][0], data['time'][-1]])
            # Update the plot
            line1.set_ydata(data['ch1'])
            line2.set_ydata(data['ch2'])
            line3.set_ydata(data['ch3'])
            line4.set_ydata(data['ch4'])
            line1.set_xdata(data['time'])
            line2.set_xdata(data['time'])
            line3.set_xdata(data['time'])
            line4.set_xdata(data['time'])
            plt.pause(0.001)

except Exception as e:
    i.stop_streaming()
    mim.relinquish_ownership()
    raise e
finally:
    mim.relinquish_ownership()
```

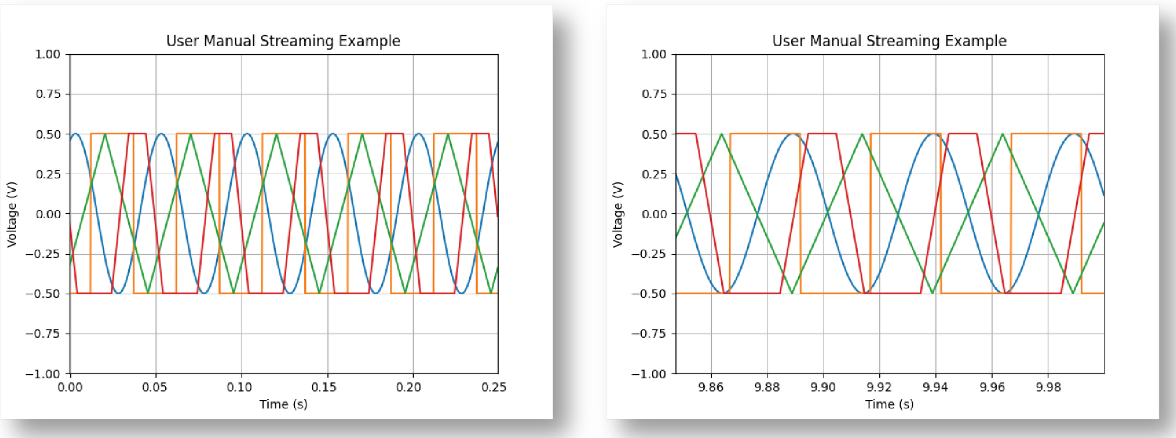
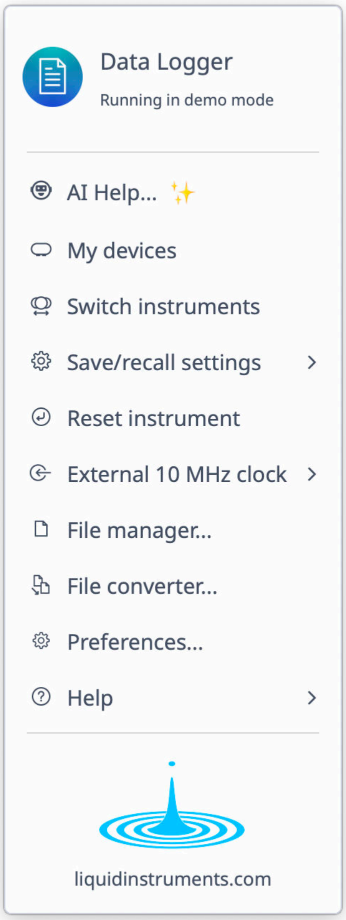


Figure 29. Plotting real-time data from the streaming example; first frame (left) and final frame (right)

Additional controls

Main menu

The main menu can be accessed by clicking the icon on the top-left corner.



AI Help... Opens a window to chat to an AI trained to provide Moku-specific help (Ctrl/Cmd+F1)
My Devices returns to device selection screen
Switch instrument to another instrument
Save/recall settings

- Save current instrument state (Ctrl/Cmd+S)
- Load last saved instrument state (Ctrl/Cmd+O)
- Show the current instrument settings, with the option to export the settings

Reset instrument to its default state (Ctrl/Cmd+R)
Sync Instrument slots in Multi-Instrument Mode*
External 10 MHz clock selection determines whether the internal 10 MHz clock is used.
Clock blending configuration opens the clock blending configuration pop-up *
Power Supply access panel*
File Manager access tool
File Converter access tool
Preferences access tool
* If available using the current settings or device.

Help

- **Liquid Instruments website** opens in default browser
- **Shortcuts list** (Crtl/Cmd+H)
- **Manual** Open the user manual in your default browser (F1)
- **Report an issue** to the Liquid Instruments team
- **Privacy Policy** opens in default browser
- **Export diagnostics** exports a diagnostics file you can send to the Liquid Instruments team for support
- **About** Show app version, check for updates or licence information

File converter

The File converter can be accessed from the main menu.

The File converter converts a Moku binary (.li) format on the local computer to either .csv, .mat, .hdf5 or .npy format. The converted file is saved in the same folder as the original file.

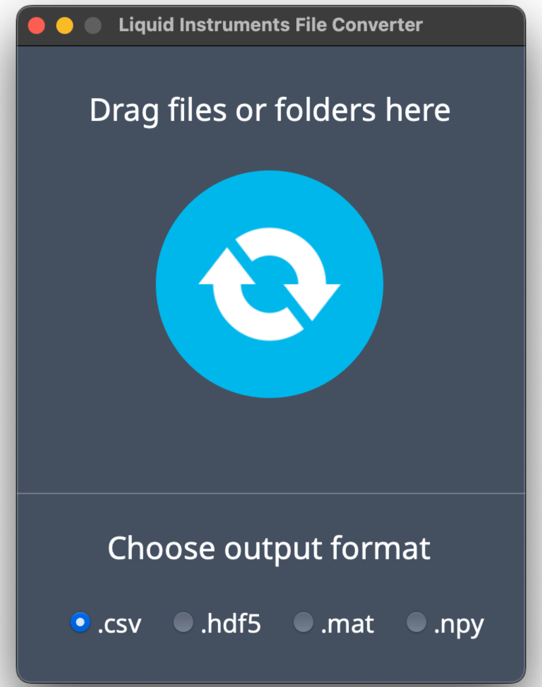


Figure 32: File Converter user interface.

To convert a file:

1. Select a file type.
2. Open a file (Ctrl/Cmd+O) or folder (Ctrl/Cmd+Shift+O) or drag and drop into the File converter to convert the file.

File manager

The File manager allows you to download the saved data from your Moku device to the local computer, with optional file format conversion. Once a file is transferred to the local computer, an icon appears next to the file.

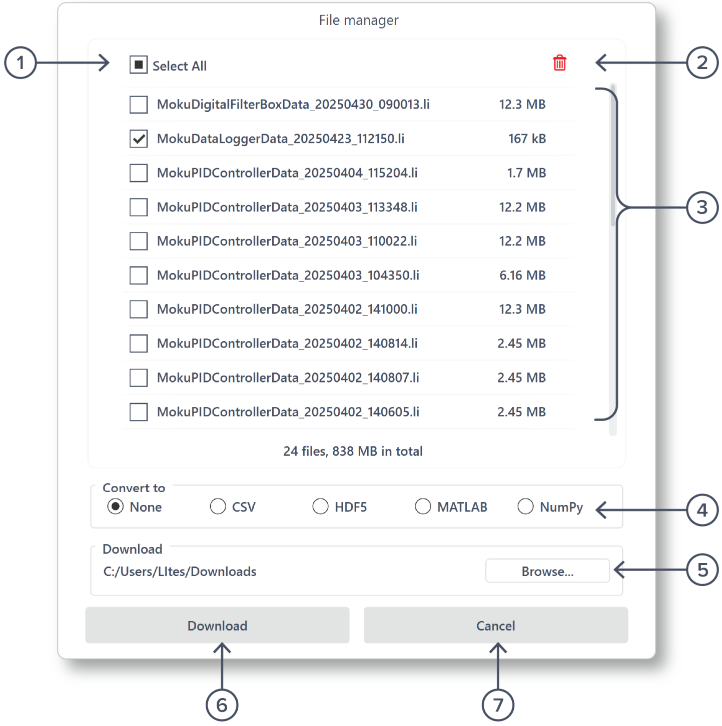


Figure 30. File exporting User Interface and settings.

To save logged data:

① **Select all** files logged to the device's memory, to download or convert.
② **Delete** all the selected file/s.
③ Browse and **select file/s** to download or convert.
④ Select an optional **file conversion format**.
⑤ Select a **location** to export your selected files to.
⑥ **Export** the data.
⑦ **Close** the export data window, without exporting.

Preferences and settings

The preferences panel can be accessed via the Main Menu . In here, you can reassign the color representations for each channel, switch between light and dark mode, etc. Throughout the manual, the default colors are used to present instrument features.

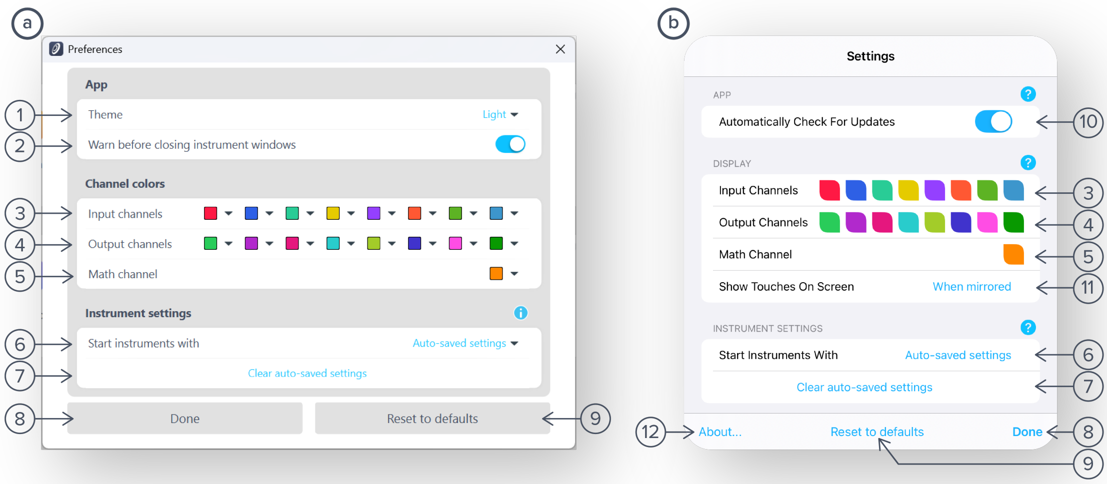


Figure 31. Preferences and settings for the Desktop (a) and for the iPad (b) App.

① Change the App theme, between dark and light mode.
② Choose if a warning opens before closing any instrument windows.
③ Tap to change the color associated with the input channels.
④ Tap to change the color associated with the output channels.
⑤ Tap to change the color associated with the math channel.
⑥ Select if instruments open with the last used settings, or default values each time.
⑦ Clear all auto-saved settings and reset them to their defaults.
⑧ Save and apply settings.
⑨ Reset all application preferences to their default state.
⑩ Notify when a new version of the app is available. Your device must be connected to the internet to check for updates.
⑪ Indicate touch points on the screen with circles. This can be useful for demonstrations.
⑫ Open information about the installed Moku application and license.

External reference clock

Your Moku may support the use of an external reference clock, which allows Moku to synchronize with multiple Moku devices, other lab equipment, lock to a more stable timing reference, or integrate with laboratory standards. The reference clock input and output are on the rear panel of the device. Each external reference option is hardware dependent, review the available external reference options for your Moku.

Reference Input: Accepts a clock signal from an external source, such as another Moku, a laboratory frequency standard, or an atomic reference (for example, a rubidium clock or a GPS-disciplined oscillator).

Reference Output: Supplies the Moku internal reference clock to other equipment that require synchronization.

If your signal is lost, or is out of frequency, your Moku will revert to using its own internal clock until the reference signal returns. If this occurs, check the source is enabled, and that the correct impedance, amplitude, tolerance, frequency, and modulation are attached to the reference. Check the required specifications in the device specsheets.

When the reference returns within range, status changes to "validating" and then "valid" once lock is re-established.

10 MHz external reference

To use the 10 MHz external reference function, ensure "always use internal" is disabled in the Moku application, found in the main menu under "External 10 MHz clock". Then, when an external signal is applied to your Moku reference input and your Moku has locked to it, a pop up will show in the app. On some devices, the external reference information will be shown in the LED status as well, more information can be found in your Moku Quick Start Guide.

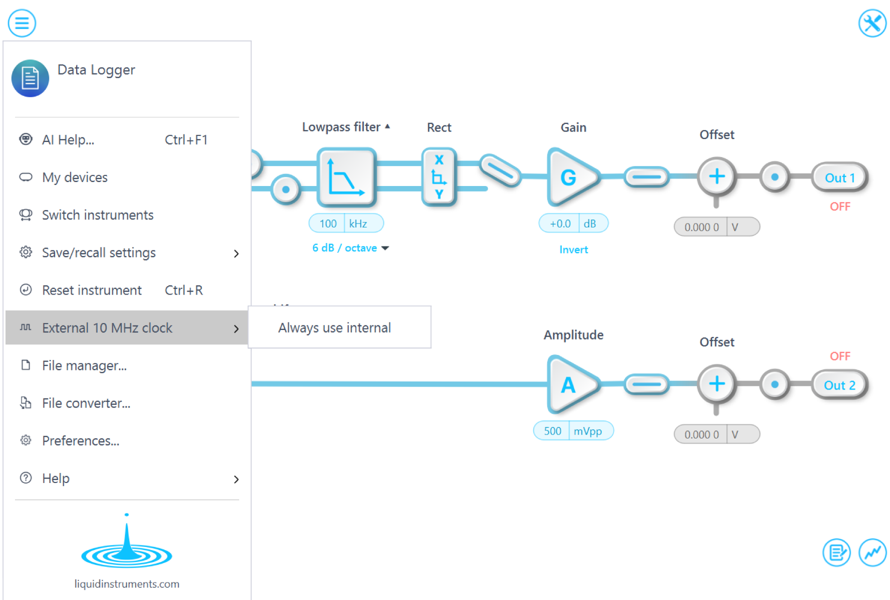


Figure 32. Moku main menu with "Always use internal" reference disabled and using an external reference.

Clock blending configuration

If available, Moku blends up to four clock sources simultaneously for more accurate phase, frequency, and interval measurements across all time scales. A low phase-noise Voltage-Controlled Crystal Oscillator (VCXO) is blended with a 1 ppb Oven-Controlled Crystal Oscillator (OCXO) for optimal wide-band phase noise and stability, which can be blended further with an external frequency reference and GPS disciplining to synchronize Moku with your lab and UTC.

The VCXO and OCXO will always be used for the clock generation signal. The external and 1 pps references are optional and can be enabled or disabled in the "Clock blending configuration..." settings from the main menu ☰. The loop bands are adjusted based on the different possible clock source configurations, shown in Figure 33, where the frequencies of the bands represent where each oscillator's phase noise dominates.

Read how the clock blending works on Moku:Delta for more details.

Figure 33. Moku clock blending configuration dialog with an external 10 MHz frequency reference and GNSS enabled.

① **VCXO jitter reference** is always used for clock generation, handling high frequency jitter with the lowest noise.

② **OCXO jitter reference** is always used for clock generation, ensuring moderate term stability.

③ **External 10/100 MHz frequency reference** uses a "10 MHz" or "100 MHz" external reference to correct drift in the local oscillator, noting your Moku will have to be restarted after each change between a 10 MHz and 100 MHz source.

④ **1 pps synchronization reference** uses an "External" or "GNSS" reference to sync with UTC and correct drift in the local oscillator. The estimated clock stability is a measure of how much the reference performance deviates relative to the local OCXO/VCXO timebase (as currently blended and, if enabled, steered by the external 10 / 100 MHz External reference).