Moku Phasemeter

User Manual

LIQUID INSTRUMENTS

Table of Contents

Introduction

The Phasemeter is an instrument that can measure phase of a periodic signal with a precision of 1 µrad, making it useful for analyzing frequency stability in oscillators and lasers. The instrument can display data as a Timeseries, Power spectral density, or Allan deviation.

The Phasemeter utilizes a phase-locked loop (PLL) in its operation, and this guide provides details on the underlying architecture of the instrument. We also include a general example in the quick start guide and a small number of in-depth examples to provide a foundation for new users.

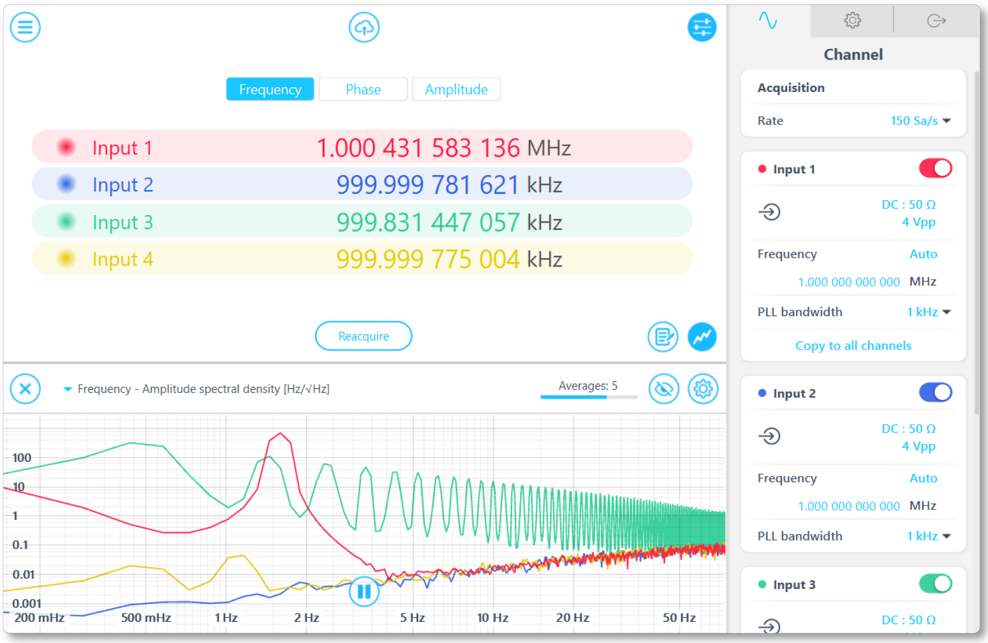

Figure 1. Phasemeter user interface with a live display of the instantaneous frequency (top) and Power spectral density (bottom) with the settings panels (right).

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 Phasemeter Datasheets.

Quick start guide

This guide walks through the typical workflow when using the Phasemeter.

In this example, we measure the phase fluctuations of a phase modulated signal with an external reference. The signal is a 1 Vpp, 1.234 MHz sine wave that is phase modulated +/-90 degrees at 10 Hz. The signal is generated from an external device that shares the same clock, and it is connected to Moku's Input 1.

Step 1: Configure the analog front end settings for the inputs

Toggle the input signals to be traced by the Phasemeter by pressing the . Set the analog front end settings for the input by clicking on the field. In this case, Input 1 is enabled in AC coupling mode with a 50 Ω input impedance to match its source. Select the smallest input range without clipping the signal. In this example, the voltage range is chosen to be 4 Vpp to accommodate the 1 Vpp signal. Note that input range is set as "attenuation" in Multi-Instrument Mode.

Step 2: Configure the initial frequency of the phase-locked loop (PLL)

To start tracking the input signal, an initial frequency estimate needs to be provided for the PLL. This can be set manually (recommended) or automatically acquired. To use automatic acquisition, press the Auto button, which will estimate the frequency. To manually set the frequency, toggle the Auto off, and set the value in the field. In this example, the frequency is manually set to 1.234 MHz for the PLL to track.

Step 3: Configure the PLL bandwidth

Depending on the rate of phase change in the signal, the appropriate tracking bandwidth needs to be selected. Here, the PLL bandwidth is selected to be 1 kHz as the default value. Press Re-acquire to allow the PLL to start tracking phase with the given settings.

Step 4: Observing the phase data on the Data visualization panel

Click on the icon to view the frequency/phase/amplitude information on the Data visualization panel. The desired parameter can be viewed by switching between the Frequency/Phase/Amplitude tabs. For this example, the phase information is displayed to observe the modulation signal. The panel is set to record a timeseries with Autoscaled enabled. By clicking the drop down on the top left of the panel, the Timeseries can be changed to a Power Spectral Density plot. A clear button can be used to ensure the data is free from any outliers.

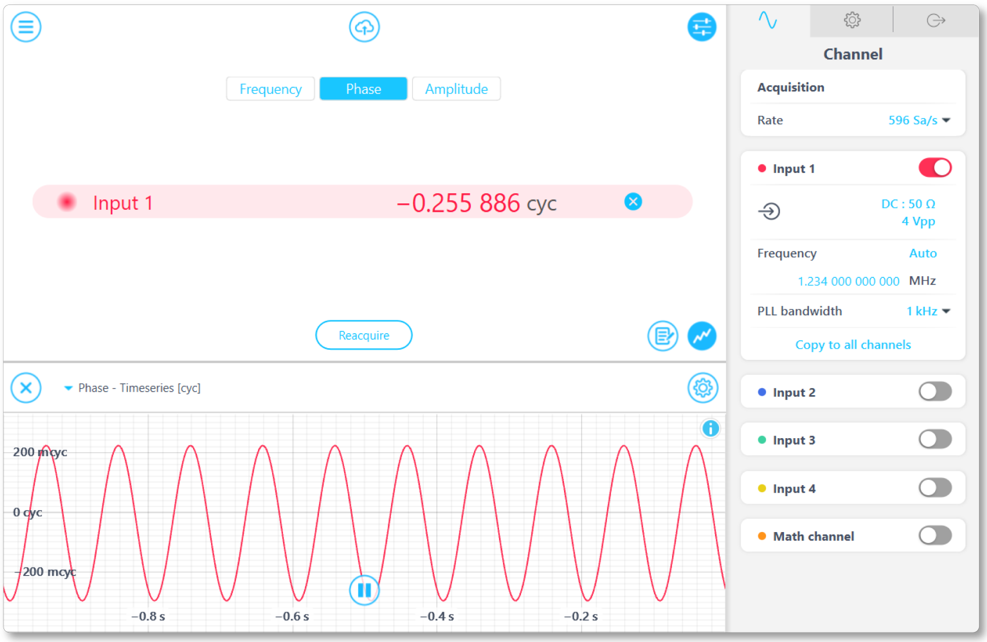

Figure 2. Phasemeter data visualization panel showing the time series of the phase fluctuations of the signal.

Step 5: Choose the acquisition rate for the data

In this example, the default rate of 150 Sa/s is used as this is sufficient to reconstruct the phase modulation rate of 10 Hz. The data visualization panel can only display data up to the first three sampling rates, while the Data Logger can run at all available sampling rates

Step 6: Logging using the embedded Data Logger

Clicking on the to bring up the Data Logger interface. The Data Logger records the frequency, phase, amplitude along with the I and Q components for each channel. To perform a one-hour logging session, we will set the log to immediate start for 01:00:00.000 (hh:mm:ss.ms) and press to start the logging. The logging session can be stopped at any moment by pressing .

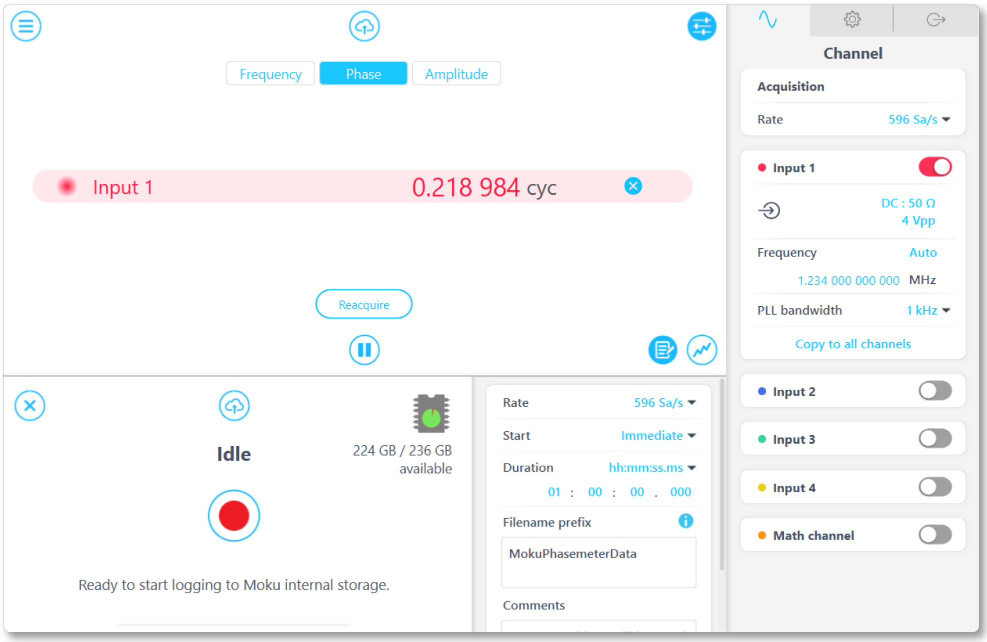

Figure 3. Phasemeter interface to access the embedded Data Logger.

Step 7 : Downloading the logged data

Once the Data Logger has completed logging the information, click on the to open the file directory and download the logged file. The file can be converted to either .csv, .npy, .hdf5 or .mat format to be used for further post-processing.

Principle of operation

Phasemeter operates using a digital phase-locked loop (PLL), which uses the demodulated feedback signal to drive a voltage-controlled oscillator. The reference oscillator follows the phase fluctuations of the input signal using the corrections from the feedback loop. The corrections can in turn be used to provide a readout of the phase information of the input signal. The block diagram of the Phasemeter can be shown with an input signal $A \cdot \sin(\omega_1 t + \phi_1)$ with a reference signal $B \cos(\omega_2 t + \phi_2)$.

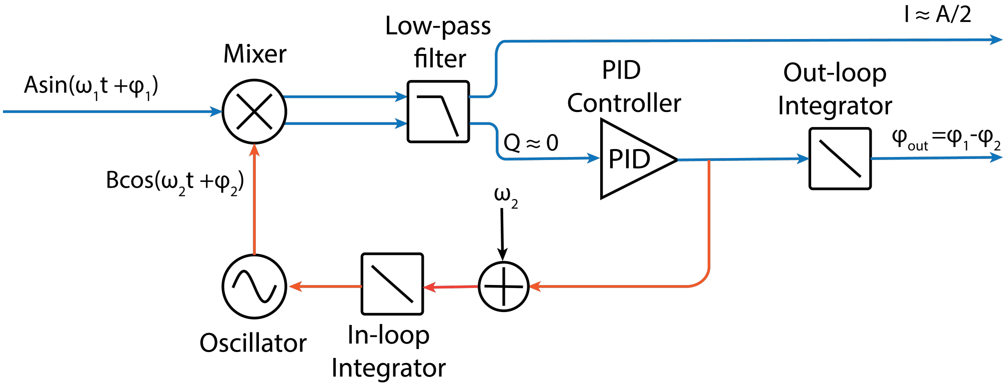

Figure 4. Block diagram of the Phasemeter. The Phasemeter detects the phase error between the input signal and the reference oscillator and is fed back to the reference oscillator to follow the input. The error signal provides the phase readout in the phasemeter.

Similar to a Lock-In Amplifier, the instantaneous phase error between the input signal and local oscillator is detected by demodulation using a digital mixer and low-pass filter. The demodulation process produces two spectral components: an up-shifted signal with a frequency equal to the sum of the input and reference signals, and a down-shifted signal with a frequency equal to the difference of the input and reference signals.

The low-pass filter is used to filter the up-shifted components and gives the phase error. The detected phase error is then passed through a PID controller to generate a feedback control signal to continuously update the phase of the reference oscillator. In the above figure, the feedback controller updates φ_2 with the difference between the reference oscillator and the input signal, leading to φ_2 following the frequency and phase difference between the two signals.

The phase of the input signal relative to the local oscillator is measured by keeping a record of every change made to the phase of the local oscillator by the controller. By utilizing a dual-phase demodulator, both the I and Q components can be computed. The Q component is used in the feedback loop, as it provides an approximately linear slope for error correction. When the feedback loop is engaged, the Q component is driven to zero, while the I component reaches a maximum and gives the amplitude of the signal.

Tuning the PLL bandwidth is paramount for optimizing the tracking capability of the PLL. A wider bandwidth can allow the Phasemeter to track larger phase deviations but increases susceptibility to additive noise. Conversely, a narrower bandwidth reduces additive noise but limits the maximum rate of phase change or frequency error that can be tracked reliably. The effective bandwidth of the PLL is determined by the amplitude of the input signal, the low-pass filter and the gains of the PID controller and integrators.

Using the instrument

Channel configuration

Signal inputs

The analog input to the Phasemeter allows you to change the analog frontend settings for each input.

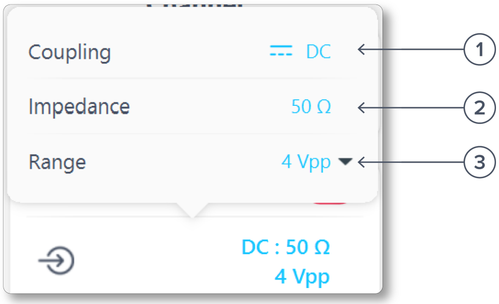

Figure 5. Input channel configuration on Phasemeter.

ID	Description
①	Select between AC and DC input coupling.
②	Select between 50 Ω and 1 MΩ input impedance. *Note that the input impedance is fixed on the Moku:Go at 1 MΩ.*
③	Select an input range.

PLL settings

The Moku Phasemeter features a phase-locked loop (PLL) with variable bandwidth and option to define the initial (acquisition) frequency.

① Toggle between the PLL auto-acquiring the initial frequency or PLL using a user-defined frequency.

② View or manually type in the initial frequency for your PLL down to µHz.

③ Select the PLL bandwidth to configure the tracking behavior of the PLL.

④ Copy the front-end settings and the PLL settings to all input channels (Option only on Input 1).

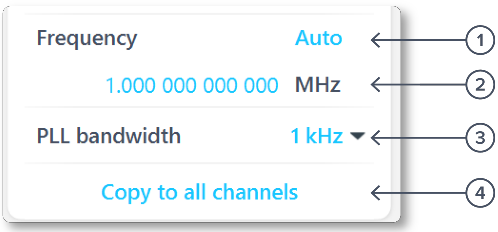

Figure 6. PLL settings on the Phasemeter.

The Phasemeter will attempt to track frequencies around the specified acquisition frequency. If the frequency of the tone is known, you can set it manually by tapping the ② below the "Frequency" label (**Recommended approach**). If the frequency of the tone is not known, you can enable auto-acquisition mode, by pressing the ① icon. This will automatically search for and track the highest-magnitude tone between 10 kHz and maximum supported frequency determined by hardware. Note: Auto-acquisition does not work reliably for tones below 10 kHz.

The Moku Phasemeter will measure the phase of an input signal whose frequency varies at a rate up to the specified bandwidth. Note that the selected bandwidth should not exceed one tenth of the acquisition frequency.

Outputs

The Phasemeter output can be used as a sine wave generator with the option to be phase locked to the input signal. Alternatively, the output can be used to generate the frequency/phase/amplitude fluctuations of the input signal. The output can be found by pressing the tab.

Sine output

The Phasemeter features sine generators with manual control over amplitude, frequency, and phase. The sine wave can be set to be phase-locked to their corresponding input channel with the option to multiply the frequency up to 250x or down to 1/8x.

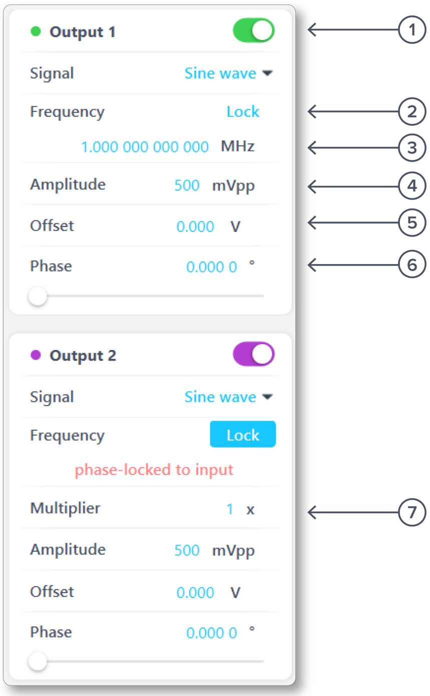

Figure 7. Phasemeter Sine wave output.

① Enable/Disables the output.

② Activate/Deactivate phase lock to the tracked input signal.

③ Configure the frequency.

④ Configure the amplitude of the sine wave.

⑤ Configure the amplitude DC offset of the sine wave.

⑥ Configure the phase offset to the sine wave. The phase offset is with respect to the input signal from 0 to 360 degrees.

⑦ Choose the multiplier factor for phase locked tones from 0.125x to 250x.

Frequency/Phase/Amplitude output

Alternatively, the output can generate a voltage signal that is proportional to the accumulated phase error, frequency offset, or amplitude for closed-loop control applications. This output can then be used for various applications such as feedback loops or for further filtering.

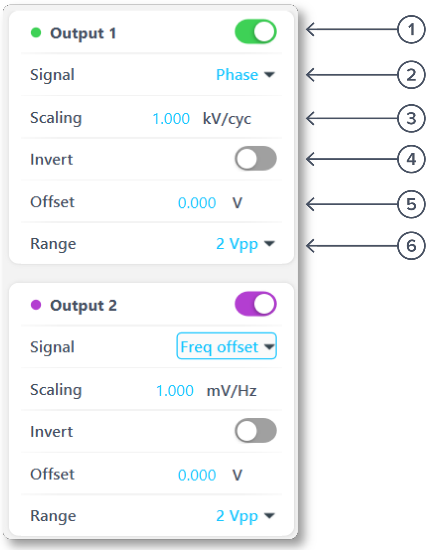

Figure 8. Phasemeter Phase/ Freq offset/Amplitude output.

① Enable/Disables the output.

② Choose between Phase/Freq offset/Amplitude or Sine wave.

③ Configure the gain scaling from cycle/Hz/Vpp to V.

④ Activate/Deactivate the signal to be inverted.

⑤ Configure the amplitude DC offset of the output.

⑥ Configure the voltage range available on the hardware platform.

Advanced settings

Advanced configuration settings on the Phasemeter enable users to modify the instrument's behavior as needed. Click on the ⚙ tab to open the Advanced options.

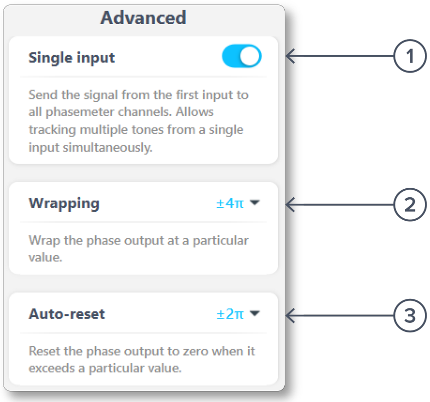

Figure 9. Advanced settings on Phasemeter.

① Enable/Disable Single input mode, which uses Input 1 for all channels and can be used to track multiple tones simultaneously.

② Enable phase Wrapping. Allows the phase information to be wrapped around +/-π, +/-2π or +/-4π.

③ Enable Auto-reset. Allows the phase to be reset every time it reaches the limits of +/-π, +/-2π or +/-4π.

Observing the data

The Phasemeter displays acquired data in multiple formats, including a time series plot and real-time measurements.

Measurements

Live measurements of input frequency, phase, and amplitude for all channels are displayed and updated in real time

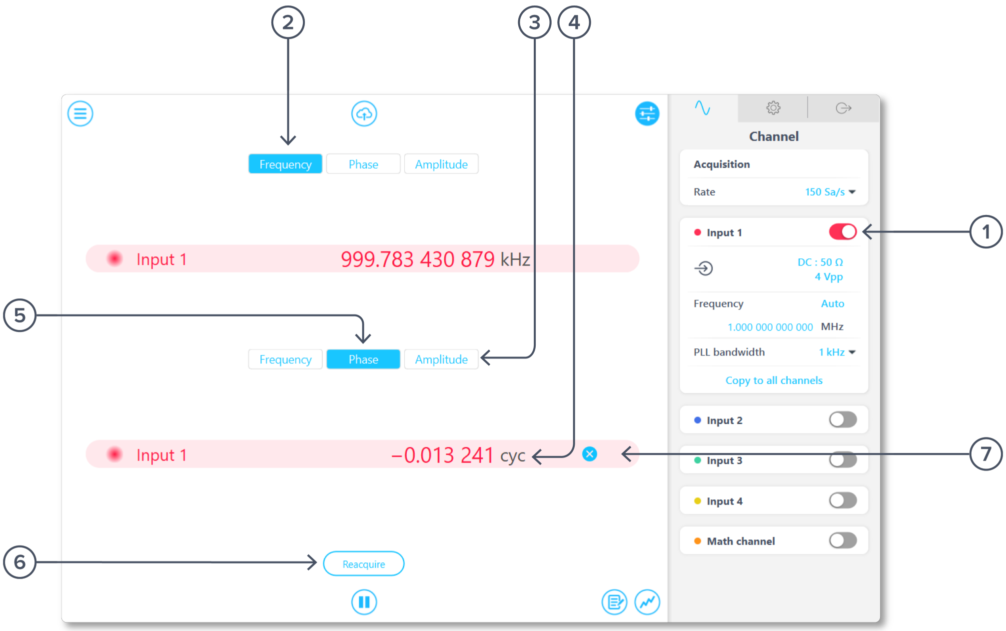

Figure 10. Phasemeter measurement tabs.

① Enable/Disables the output.

② The Frequency measurement tab displays the input signal’s frequency in hertz (Hz).

③ The Amplitude measurement tab displays the input signal's amplitude.

④ Tap the unit field to switch between cycles (cyc), radians (rad) or degrees (deg) for Phase, or between volts RMS (Vrms), volts peak-to-peak (Vpp), or decibels (dB) for Amplitude.

⑤ The Phase measurement tab displays the input signal's phase.

⑥ Tap the Reacquire button to reset all the phasemeter channels simultaneously. All channels are reset at the same time to maintain synchronization.

⑦ Zero the phase offset by tapping the icon on the right side of the display.

Math Channel

Phasemeter also includes an in-built Math channel that can measure the difference in signal characteristics between two channels. The measurement tab also displays live Math channel measurements in frequency, phase, and amplitude. The ✖ on the Math channel also zeroes the two channels involved in the Math channel subtraction.

Note: The Math channel can only be viewed as live data and is not stored using the Data Logger.

Data visualization

The Phasemeter includes a wide range of data visualization plots for observing tracked frequency, phase, and amplitude, including time series, spectra, and Allan deviation plots. The settings can be configured as:

Figure 11. Data visualization on Phasemeter.

① Select between the different plot data types.

② Access settings for the data visualization panel

- Enable/Disable Autoscale feature to dynamically update the plot scales with the signal.
- Enable to use the current plot as a reference for future measurements, Disabling will remove the reference measurements from the panel and memory.

- Clear the signals displayed on the data visualization panel.

③ Enable/Disable the different channels to be displayed on the panel. Disabling the channel here would only affect the display with the Phasemeter still tracking the channel.

④ Pause the measurements.

⑤ Open /Close the data visualization panel.

Data plots on the panel

Below are the different types of data plots available on the Phasemeter data visualization panel:

1. Time series: View parameters over time spans ranging from 0.5 seconds to 600 seconds. Adjust timescale and span using pinch gestures (iPad) or using your mouse/trackpad (Desktop) anywhere on the graph. Set the start and end times of the span manually using the slide rule located above the graph.

2. Power Spectral Density: Display a signal's distribution of power at different frequencies. The units of power spectral density are proportional to amplitude^2/Hz (e.g., cycles^2/Hz). The frequency range is dependent on the acquisition rate and is averaged over time.

3. Amplitude spectral density: Provide a measure of a signal's amplitude at different frequencies. The units of amplitude spectral density are proportional to amplitude/√Hz (e.g. cycles/√Hz). Amplitude spectral density is equal to the square root of the power spectral density.

4. Rayleigh Spectrum: Measure the coefficient of variation of the power spectral density. A value of 1 indicates Gaussian variation, less than one indicates coherent variation, more than one indicates incoherent variation.

5. Spectral coherence: Generate a unitless statistic to measure the similarity between two signals.

6. Allan deviation: Measure the stability of clocks and other oscillators. Allan dev is equal to the square root of the Allan variance. An Allan deviation of 2×10^{-6} at an averaging time of t = 1 seconds means there is an RMS error of 2×10^{-6} between two measurements taken one second apart. Refer to this app note for further information.

Note that the data visualization is limited to acquisition rates less than 596 Sa/s on the Moku:Delta and Moku:Pro and limited to less than 477 Sa/s on the Moku:Lab and Moku:Go.

Data Logging

Phasemeter comes with an embedded Data Logger to store the frequency, phase and amplitude information. The log also contains the I and Q data from the demodulation process.

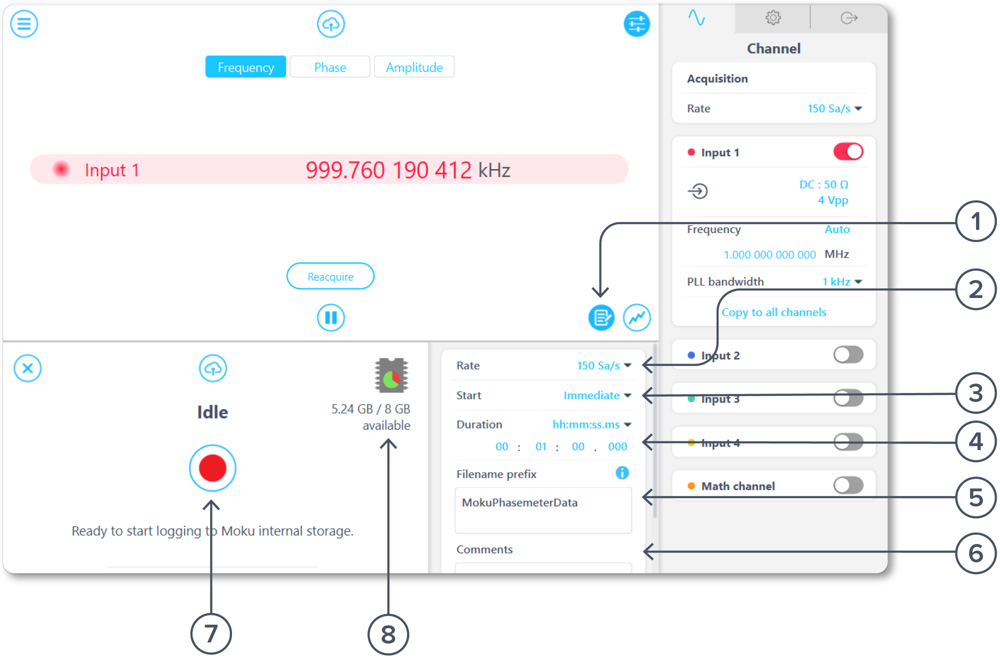

Figure 12. Embedded Data Logger in the Phasemeter.

① Open /Close the Data Logger interface.

② Choose the acquisition rate (varies between hardware models).

③ Select the acquisition start option between Immediate and Delayed.

④ Set the duration for the data log in time format {hh:mm:ss:ms} with minimum at 1 ms.

⑤ Set the filename prefix for the data log file.

⑥ Set the comments for the data log file.

⑦ Start/Stop the logging session. When stopping a session, the data till that moment would be stored with the filename specifications.

⑧ Indication of available memory for data log storage.

Exporting data

Measured timeseries data, spectra, and Allan deviation can be downloaded by clicking the as Live data. Note that there are two of these buttons, one in the live data window and one in the Data Logger window. Live data provides real-time signal visualization for immediate monitoring and quick adjustments. Logged data, on the other hand, is recorded over time for detailed analysis, documentation, and long-term system evaluation.

Live data

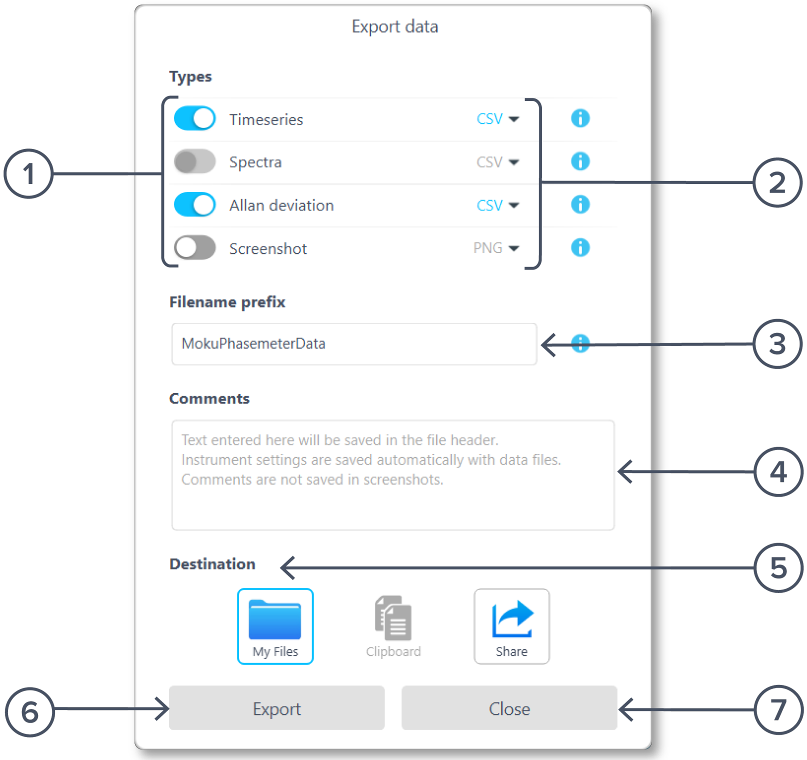

Figure 13. Phasemeter Interface for exporting Live data.

To save live data:

① Select the type of data to export:

- **Timeseries** Saves the frequency, phase and amplitude data of all the channels in either a CSV or MATLAB format, along with instrument settings
- **Spectra** Saves the PSD and Rayleigh Spectrum of the frequency, phase and amplitude of all the channels.
- **Allan Deviation** Saves the Allan deviation of the frequency, phase and amplitude on all the channels. The data also includes the errors of the Allan deviations.
- **Screenshots** Save the app window as an image, in either a PNG or JPG format.

② Select the **export format**.

③ Select the **Filename Prefix** for your export. This is defaulted to “MokuPhasemeterData” and can be changed to any filename of alphanumeric characters and underscores. A timestamp and the data format will be appended to the prefix to ensure the filename is unique.

For example: “MokuPhasemeterData_YYYYMMDD_HHMMSS_Traces.csv”

④ Enter additional **Comments** to be saved as a text-based file header.

⑤ Select the export **Destination** on your locals computer. If "My files" or "Share" is chosen, the exact location is selected when the Export button is clicked. Multiple export types can be exported simultaneously using My Files and Share, but only one export type can be exported to the clipboard at a time.

⑥ **Export** the data, or

⑦ **Close** the export data window, without exporting.

Logged data

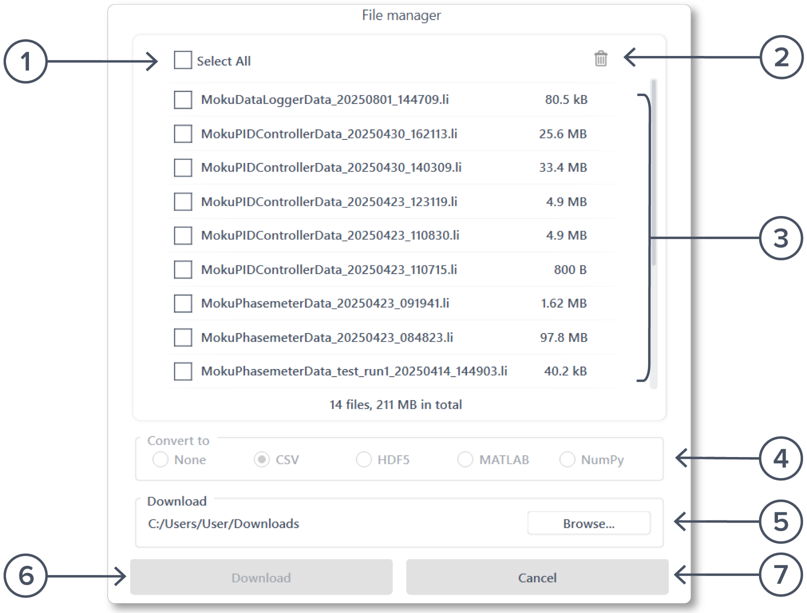

Figure 14. Phasemeter Interface to export Logged data.

To save logged data:

① **Select all** files logged to the device's memory, to download or convert.

② **Delete** 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.

Examples of using Phasemeter

Measuring an oscillator stability

Clocks are important for day-to-day timekeeping as well as in advanced applications. A common high-precision reference is provided by GPS satellites, which carry atomic clocks and are often used for synchronization in astronomy, satellite systems, and research labs. In digital platforms, clocks are simply streams of pulses repeated at a fixed frequency.

In this example, the Phasemeter can be used to measure the stability of an external device's clock. The device provides a standard 10 MHz clock reference for phase synchronization. This is given as input to the Moku, with the Moku measuring the relative clock deviations between the external device and the Moku itself.

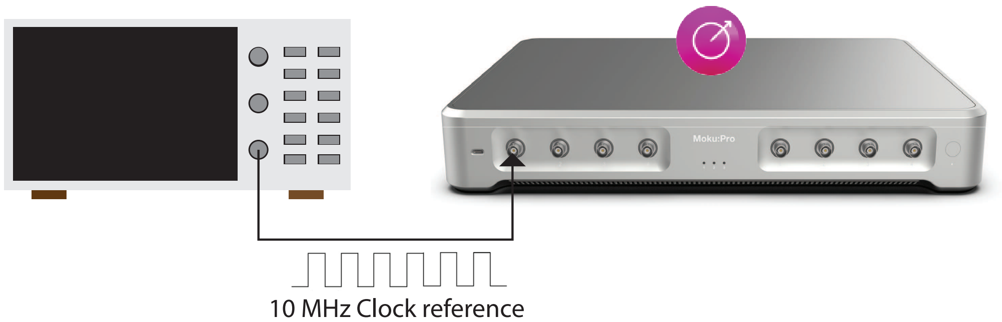

Figure 15. Setup to measure clock stability using Moku Phasemeter.

Step 1: Configure the analog front end settings for the signal input

- Set the analog front end settings for the input. In this case the input has a 50 Ω input impedance to match its source, 4 Vpp voltage range to accommodate the signal and use AC coupling.

Step 2: Select the PLL frequency configuration

- Set the PLL frequency that would be used by the Phasemeter for tracking. The PLL frequency can be manually set if the frequency is known. Otherwise, toggling "Auto" will enable the phasemeter to track the strongest frequency component in the signal. In this case, the frequency is manually set to 10 MHz.

Step 3: Select the PLL bandwidth

- Configure the PLL bandwidth depending on the rate of frequency changes in your clock noise profile. For this case, the default value of 1 kHz is sufficient to measure the clock stability.

Step 4: Observe the frequency/phase/amplitude data on the scope

- Select the for pulling up the scope to view the frequency/phase/amplitude information. The scope can be varied to observe the real-time Time series or the corresponding Amplitude Spectral Density plots.

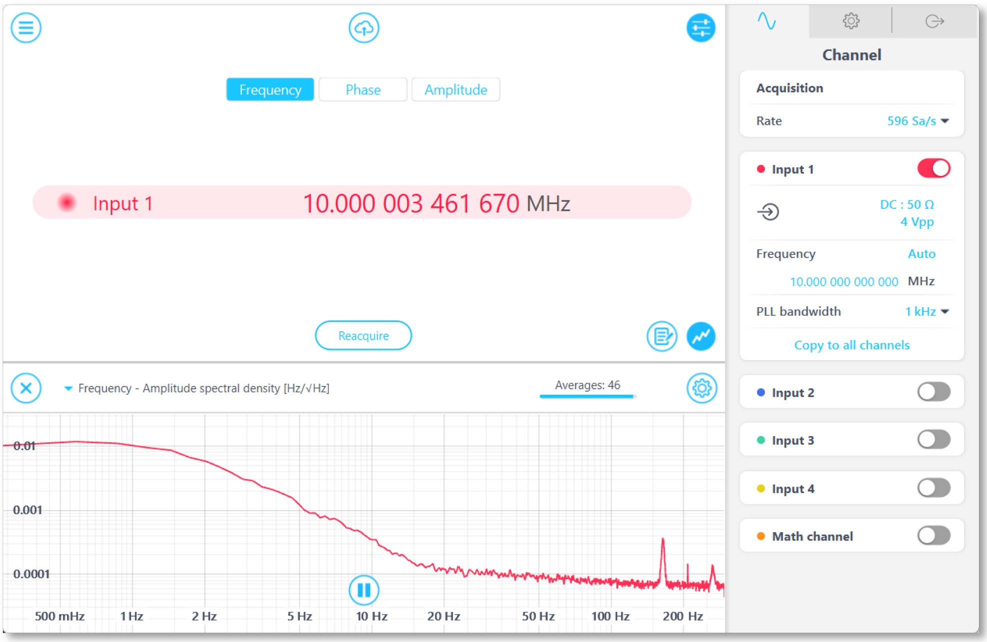

Figure 16. Phasemeter interface showing stability of the oscillator clock using an Amplitude Spectral Density.

Step 5: Log the information

- Select the [icon] to open the Data Logger interface. Set the required acquisition rate and duration and start recording the Phasemeter data. This data can later be downloaded to your device for observing long timescale behavior of the clock stability.

Laser offset phase-locking

One key application of phasemeters is in offset phase locking using heterodyne interferometry. Laser heterodyne interferometers are typically used to measure extremely small displacements, often on the order of nanometers or even fractions of a laser wavelength. In these systems, displacement information is encoded in the phase of a beat note generated by interfering two optical fields of slightly different frequencies at a photodetector.

Optical offset phase-locking is a common technique to transfer the phase stability of one laser beam to another. This is achieved by stabilizing the phase difference between the two lasers, that can be achieved by using the Phasemeter and the PID Controller in Multi-Instrument Mode.

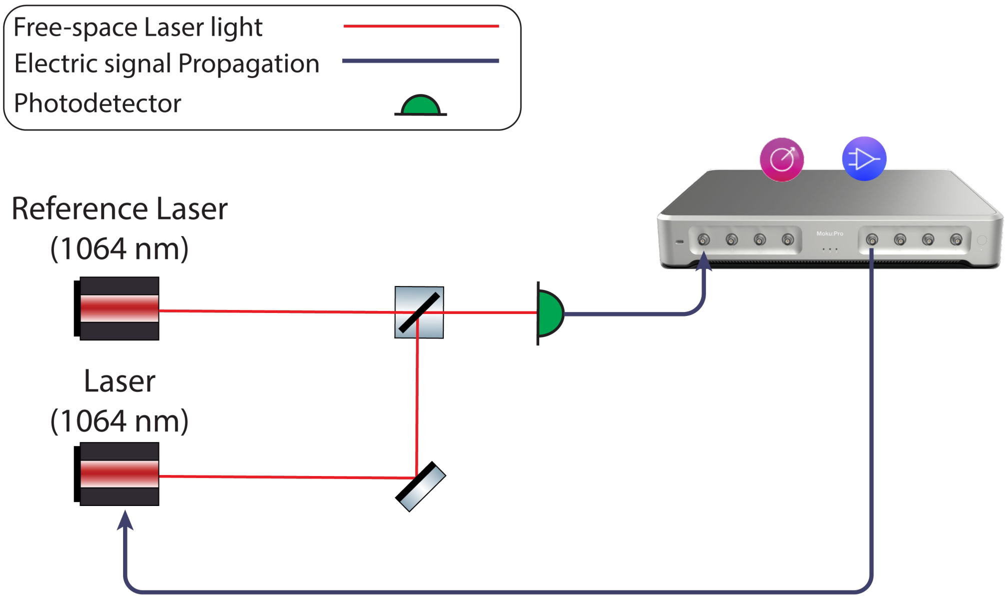

Figure 17. Experimental setup for offset locking two lasers using Moku.

The setup above considers two NPRO Nd:YAG lasers at 1064nm that are interfered with each other to generate a heterodyne beat frequency roughly around 21.456MHz. The interfered light is incident on the photodetector, which is then given as input to the Moku. Inside the Moku, the Phasemeter can be deployed and can be configured as follow:

Step 1: Configure the Multi-Instrument Mode

- Select the Multi-Instrument Mode on the Moku. Set the first slot as Phasemeter to measure the phase fluctuations and set the second slot with PID Controller to provide the controller for the stabilization.

Step 2: Configure the analog front end settings for the signal input and outputs

- The analog input is connected to the Phasemeter, while the PID Controller is given as output to control the laser frequency. In this case the input has a 50 Ω input impedance, 0 dB attenuation and use AC coupling, while the output has a 0 dB: 2 Vpp range.

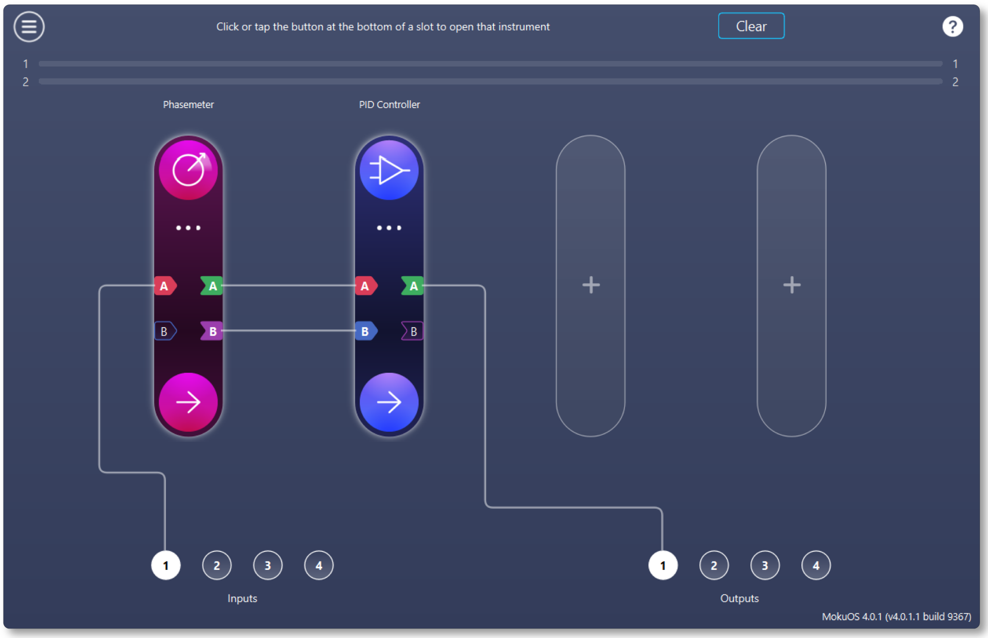

Figure 18. Multi-Instrument Mode setup using the Moku:Pro.

Step 3: Select the PLL frequency configuration

- On the Phasemeter, set the PLL frequency that would be used by the Phasemeter for tracking. The PLL frequency can be manually set if the frequency is known. Otherwise toggle on "Auto" will enable the phasemeter to track the strongest frequency component in the signal. Here the frequency is auto acquired as the laser noise changes the frequency over time.

Step 4: Select the PLL bandwidth

- Configure the PLL bandwidth depending on the rate of frequency changes in your laser noise profile. For 1064nm laser using NPRO, a bandwidth of larger than 1 kHz would be sufficient to track the free running laser frequency noise, while a stabilized laser can be tracked with a lower PLL bandwidth.

Step 5: Observe the frequency/phase/amplitude data on the scope

- Select the to view the frequency/phase/amplitude information. The scope can be varied to observe the real-time time series or the corresponding Power Spectral Density plots. Without any feedback, this would measure the free-running laser stability between the two lasers.

Step 6: Configure the phase output

- While the Phasemeter is tracking the laser stability, select the outputs Tab. Select the signal as Phase and choose a small scaling factor. Start at 10 nV/cycle scaling.

Step 7: Configure the PID Controller

- Before enabling the phase output, configure the PID Controller. The PID Controller details can be found in this manual. Set the channel matrix to [1,0;0,0] to utilize the first signal path for feedback control. Set the PID to only use the Proportional gain at 0 dB. Close all the switches and enable the output. On the Phasemeter, enable the Output1 to close the feedback loop.

Step 8: Tuning the controller

- Once the locks are engaged, slowly increase the gain parameters, either through the scaling in the Phasemeter output or through the PID Controller gains. When sufficient gain is applied, the phase deviations start to reduce. Optionally apply Integrator in the PID Controller to improve the performance. Looking at the Phasemeter live data, the performance of the lock can be monitored.

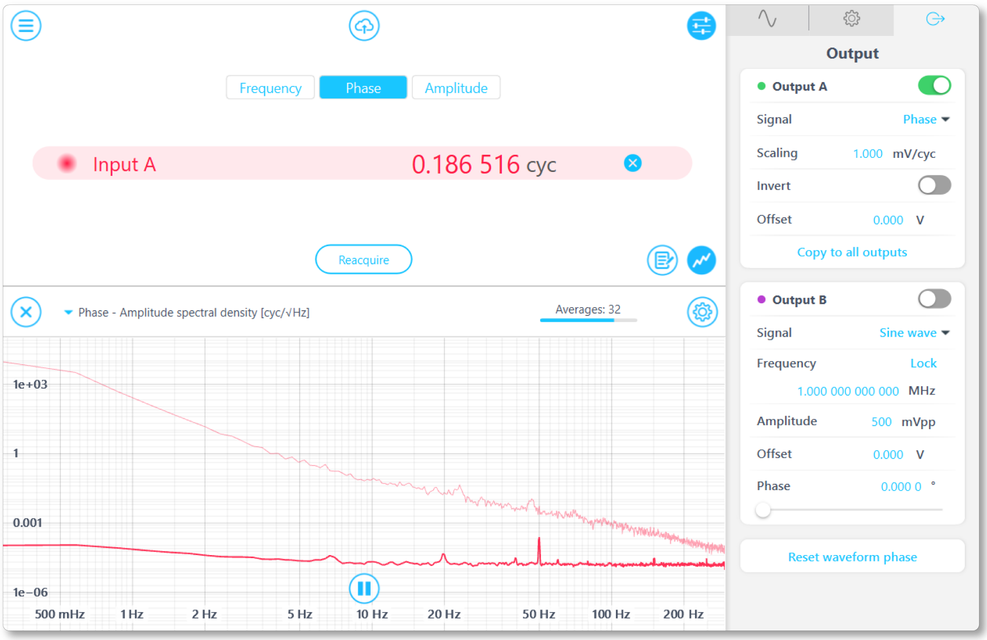

Figure 19. Phasemeter measuring the performance before and after the laser lock.

Step 9: Logging the information

- After optimizing the controller gains for the feedback system, select the to open the Data Logger interface. Set the required acquisition rate and duration and start record the Phasemeter data. This data can later be downloaded to your device for observing long timescale behavior of the offset phase lock. The Data Logger is also available on the PID Controller to observe the output to the laser actuators.

For more information on the laser offset-phase locking, please refer to this app note.

Additional tools

Main menu

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

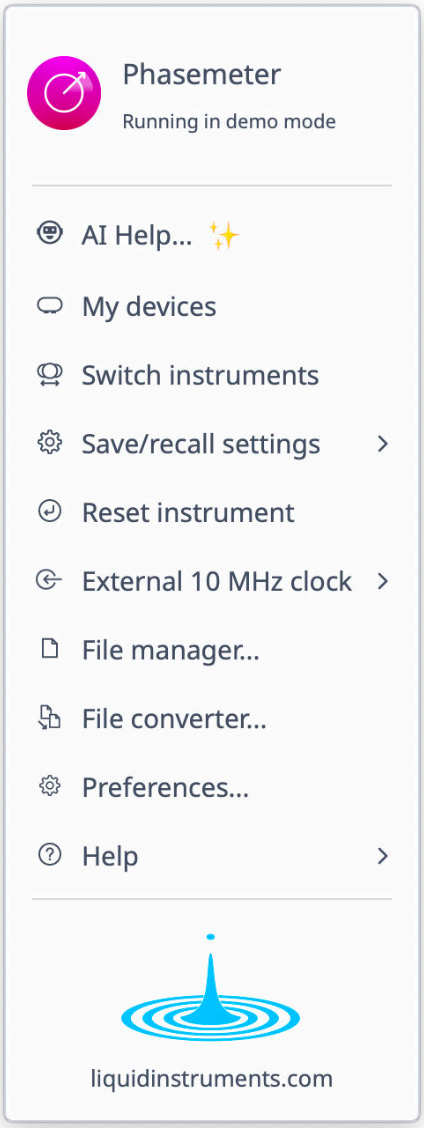

Figure 20. Main menu.

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.

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
- **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.

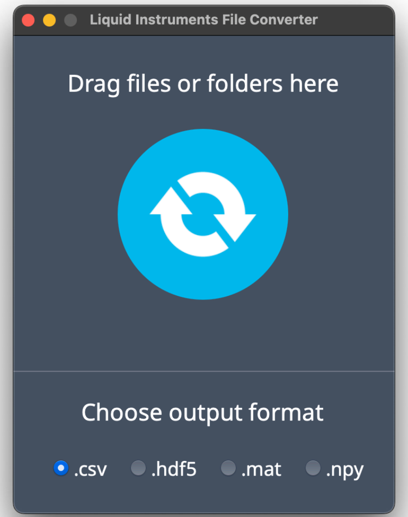

Figure 21. File converter user interface.

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.

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.

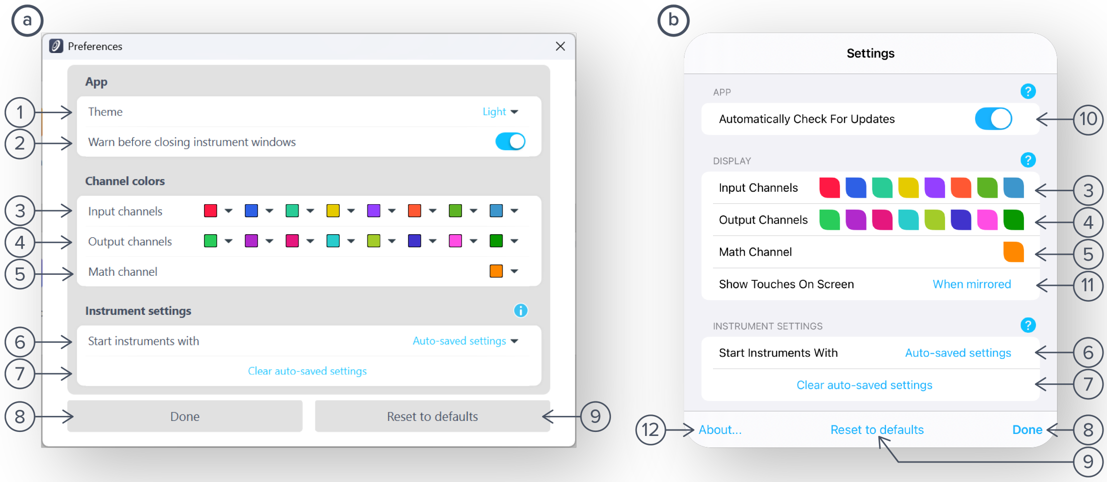

Figure 22. Preferences and settings for the Desktop (a) and for the iPad (b) App.

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.

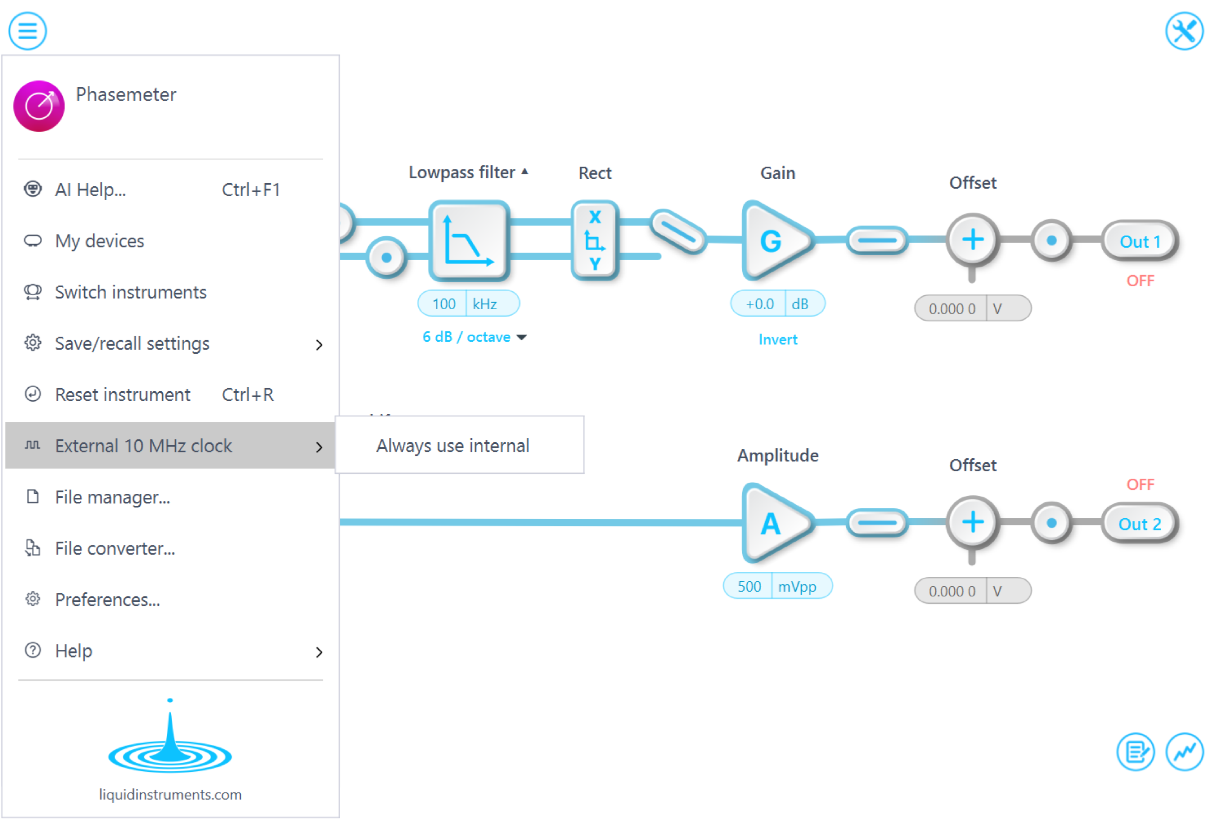

Figure 23. Moku main menu with "Always use internal" reference disabled and using an external reference.

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 24, 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 24. 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).