

# eLockIn 205/2

models produced after 01/2014

dual input 4-phase DSP lock-in amplifier

– Manual –

- LOCK-IN AMPLIFIER 1 Hz TO 10 MHz
- SPECTRUM ANALYZER
- DATA ACQUISITION
- DIGITAL OSCILLOSCOPE



9<sup>th</sup> May 2022

Anfatec Instruments AG  
Melanchthonstr. 28  
08606 Oelsnitz /V.  
Tel.: +49 (0) 37421 24212  
Tel.: +49 (0) 37421 24221  
<https://www.anfatec.de>  
email: [mailbox@anfatec.de](mailto:mailbox@anfatec.de)

# TABLE OF CONTENTS

Introduction.....	.....
General Functions.....	.....
Specifications eLockIn205/2.....	.....
Lock-In Amplifier.....	.....
Oscilloscope: .....	.....
AUX In: .....	.....
AUX Out: .....	.....
Typical Data.....	.....
Input Impedance.....	.....
Typical Frequency Characteristics.....	.....
Lock-In Basics.....	.....
Lock-In Amplification.....	.....
Mathematical Description.....	.....
Characteristical Curves of a Lock-In Amplifier.....	.....
Noise Measurements.....	.....
Example: Electrical Force Microscope.....	.....
Users manual.....	.....
Technical introduction.....	.....
Front Panel.....	.....
Screen Organization.....	.....
The Menu Tree.....	.....
Rear Panel.....	.....
The LockIn Menu [ LockIn ] .....	.....
Input Settings [ LockIn / Input ].....	.....
Reference Output [ LockIn / Ref ].....	.....
Display Settings [ LockIn / Display ].....	.....
AUX Out [ LockIn / Out ].....	.....
The Spectra Menu [Spectra].....	.....
Run/Stop [ Spectra / Run ] or [ Spectra / Stop ].....	.....
Display [ Spectra / Display ].....	.....
Acquire [ Spectra / Acquire ].....	.....
Save [ Spectra / Save ].....	.....
Oscilloscope [ Scope ].....	.....
Run/Stop [ Scope / Run ] or [ Scope / Stop ].....	.....
Display [ Scope / Display ].....	.....
Acquire [ Scope / Acquire].....	.....
Save [ Scope / Save ] .....	.....
Measure [ Measure ].....	.....
Setup [ Setup ].....	.....
Handling of parameter settings.....	.....
FTP Data Access.....	.....
HTTP Web Access.....	.....
Change Login Password for Web-Interface.....	.....
How the web-interface works.....	.....
Known Problems.....	.....
Remote Control.....	.....
Access through WEB Interface.....	.....
General.....	.....
Technical Requirements.....	.....
Authentication.....	.....
Key Codes.....	.....
How to Set Any Value in Any Sub-Menu.....	.....
Examples For Keycode Operation.....	.....
Retrieving Screenshots.....	.....
Retrievig Channel Values.....	.....
Get Internal Settings and Status Information.....	.....

Example Program: LockInRemote.....	
Use CGI-Functions via Web Browser.....	
Exported functions of the eLockin.dll.....	
Remote by Matlab.....	
Remote control for the eLockIn in LabView .....	
TCP/IP access by LabView.....	
DLL based access by LabView.....	
Remote control for the eLockIn in Python .....	
Software-Update.....	
Revision History.....	
Manual Revisions.....	
Hardware Revisions.....	
Software revision.....	
Firmware revision.....	
Appendices.....	
Schematic diagram of the signal paths.....	
Diagram of the lockin amplification paths.....	
Frontpanel Dimensions.....	
Instrument Specific Test Sheets .....	
Preamplifier Connector Pinout.....	
CE Certificate.....	
Traceability Chart.....	

The manual uses the following style declarations to visualize certain meanings:

This style describes shortly, how to reach a parameter in the menus. The list of buttons to be pressed is written usually in these brackets: [ ... ].

***Parameter names changeable in the menus are written in this style.***

***The parameter values have this style.***

***In contrast to parameters which are hidden behind some menu entries, the main menus as LockIn, Measure, ... are directly accessible by buttons, which are written in this style. Also, file names used for the remote programming are written in this style.***

Examples are written like this.

**Notes to the user that might be helpful are shown in this style.**

Copyright 2002-2022 Anfattec Instruments AG. All rights reserved. Anfattec, AMU and eLockIn are trademarks of Anfattec. Other product and brand names may be trademarks or registered trademarks of their respective owners.

Anfattec Instruments AG assumes no responsibility for any damage or loss resulting from use of this handbook.

Anfattec Instruments AG assumes no responsibility for any damage or loss resulting from use of the software. Anfattec Instruments AG assumes no responsibility for any damage or loss by deletion of data as a result of malfunction, dead battery, or repairs. Be sure to make backup copies of all important data on other media to protect against data loss. Using any part of the software indicates that you accept the terms of Anfattec Software License Agreement.  
Manual date: 05/06/22

# 1 INTRODUCTION

## 1.1 GENERAL FUNCTIONS

The eLockIn205/2 is a versatile scientific instrument that supports the following functions:

*LockIn Amplifier*

*Digital Oscilloscope*

*Spectrum Analyzer*

*Data Acquisition*

The instrument works as a stand-alone device with a 5.6 inches TFT display. Its internal functionality is best described with the following overall schematics:

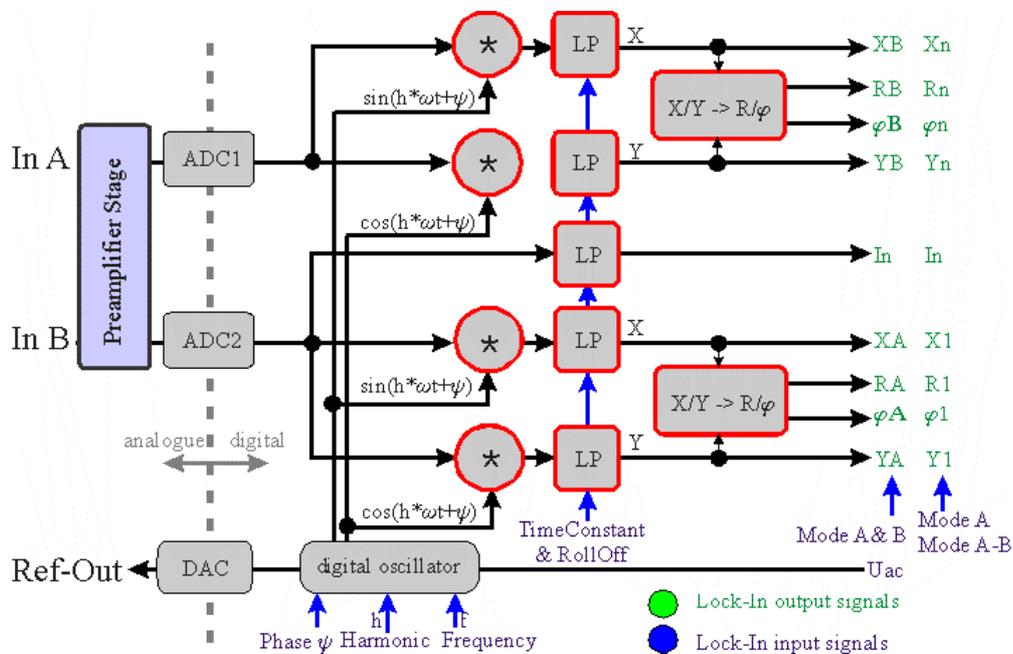


Figure 1: Schematic diagram with the basic components of the lock-in amplifier (see page 5).

The two input signals A and B can be used in three different input modes:

- Mode **A**: the input A is given to both AD-converters simultaneously.
- Mode **A-B**: after a first amplification, the difference between A and B is formed electronically. The resulting signal A-B is given to both AD-converters simultaneously.
- Mode **A & B**: the two inputs are amplified separately with the same amplification factor and feed independently into the two AD-converters.

**Note:** When the Harmonic is set to 1 and the input configuration A-B or A is chosen, the signals  $X_n, Y_n, R_n$  and  $\Phi_n$  are displayed as  $X, Y, R,$  and  $\Phi$  and equal the results provided as  $X1, Y1, R1$  and  $\Phi1$ .

The different stages of input amplification and switching between the three input modes

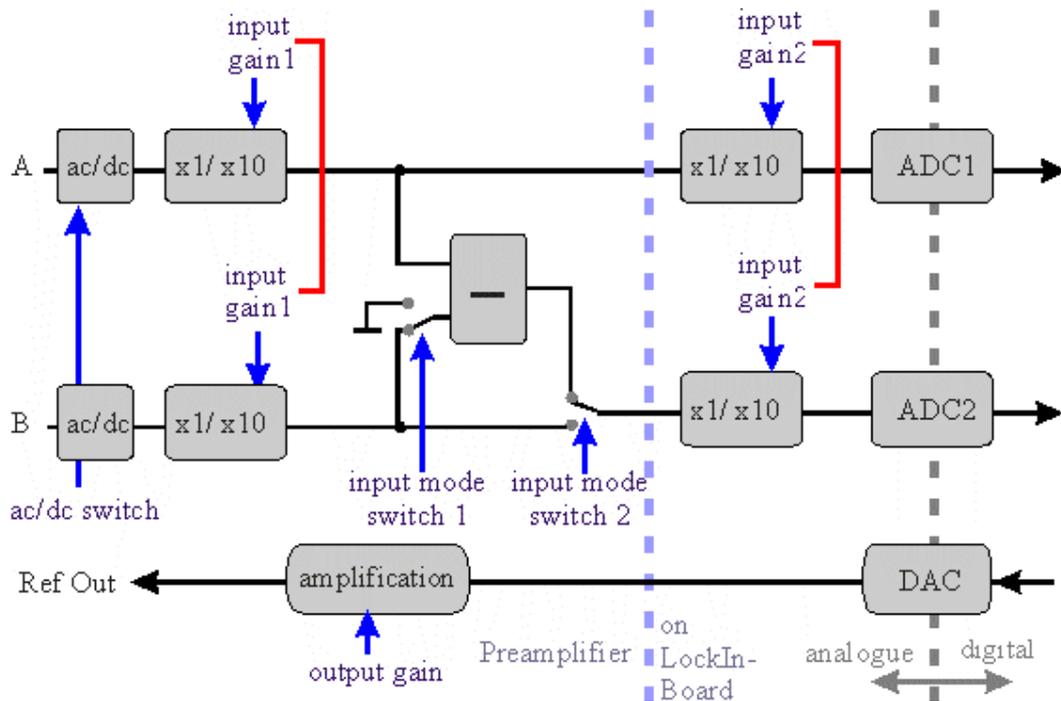


Figure 2: Schematic of the input stage describing the switching between the three input modes A, A-B and A & B as well as the position of the input amplification stages in order to realize the three different input gains.

The signals which have to be analyzed run through analogue amplification stages whose 3 dB bandwidth is 10 MHz. They are converted with a 40 MHz sampling rate. The whole lock-in analysis is realized in a fully digital design. The digital oscillator output signal is D/A-converted with a rate of 160 MHz and amplified to provide the reference output for the user.

All internal signals marked in green (output) or blue (input) color are handled with approx. 40 kHz rate, so that the shortest time constant available for these signals is 25 ms. Further 96 bit digital low-pass filters extend the time constant range up to 1 ks. The user has access to all lock-in inputs and lock-in outputs as well as to 8 additional analogue input channels AUX-In which are converted with 24 bit @ 40 kHz.

Out of this signal variety, the user selects 4 signals in order to display them in acquired frequency spectra or in the oscilloscope screen.

Besides these general functions, the eLockIn205/2 provides:

- ◆ 64 Megabytes of internal data storage
- ◆ USB connection for external flash memory (USB flash drive)
- ◆ Remote Control via Ethernet access
- ◆ HTTP-Server

## 1.2 SPECIFICATIONS ELOCKIN205/2

### 1.2.1 LOCK-IN AMPLIFIER

#### General

Digital Quad-Phase Lock-In Amplifier	
Dynamic Reserve	135 dB <sup>(1)</sup>
Noise (single ended A, dc 50 Ohm)	~ 5 nV <sub>rms</sub> /Hz <sup>0.5</sup> @ 100kHz
Remote Control	Ethernet
Time Constants	0.1 μs to 1 ks (Butterworth filter) 1 to 200 cycles in Sync mode
Full Scale Sensitivity	10 nV ... 10 V in 1-2-5 sequence
Phase Resolution	0.0001°

#### Signal Input

<b>Voltage Input</b> (A and B)	BNC, single-ended (A) or dual (A&B) or differential (A-B)
Input Coupling	DC or AC (f <sub>-3dB</sub> = 2 Hz)
Sampling Rate	> 500 MHz
Converter Resolution	16 Bit
Damage Threshold	+/- 12 V (in ON state) +/- 2 V (in OFF state)
Full Scale Input Ranges	±3.6 V <sub>rms</sub> , ±360 mV <sub>rms</sub> , ±35 mV <sub>rms</sub>
Roll-Off	6 dB/oct, 12 dB/oct, 24 dB/oct
Time Constants	100 ns to 1 ks in 1-2-5 steps or in Sync-Mode: 1/f to 200/f in 1-2-5 steps
Gain Deviations between Dynamic Ranges:	< 1 % (dc to 4 MHz) < 5 % (4 MHz to 10 MHz)
Gain accuracy @ 22°C	± (0.0004 % of range) + 0.8 % measurmt. (dc to 2 MHz) or up to 5 % of meas. (2 MHz to 10 MHz)
Phase error (calibrated with 15 cm long cables)	< 0.5° (dc to 1 MHz) < 5° (2 MHz to 10 MHz)
Input Impedance	~ 1 MΩ    15 pF (see 1.3.1)
Bandwidth	dc to 10 MHz
Input Noise @ U <sub>out</sub> = 0 V, t = 1 ms	with 50 Ω @ input
@ 100 kHz, high dynamic	< 5 μV <sub>rms</sub> /Hz <sup>0.5</sup>
@ 100 kHz, normal dynamic	< 250 nV <sub>rms</sub> /Hz <sup>0.5</sup>
@ 100 kHz, low dynamic	< 7 nV <sub>rms</sub> /Hz <sup>0.5</sup>
Harmonic distortion	- 80 dB

## Reference Output

Internal Oscillator	10 mHz .. 10 MHz
Internal Frequency Resolution	3 mHz <sup>(1)</sup>
Frequency Accuracy	+/- 50 ppm from 0 °C to 70 °C
Reference Output Voltage	< 0.1 mV <sub>rms</sub> ... 7 V <sub>rms</sub>
Maximum Output Current	typ. 90 mA
Amplitude accuracy @ 22°C	0.8 % (dc to 2 MHz) 5 % (2 MHz to 10 MHz)
Phase error (absolute)	< 0.5° (dc to 1 MHz) < 5° (1 MHz to 10 MHz)
Output Noise @ U <sub>out</sub> = 1 mV <sub>rms</sub> , τ = 1 ms @ 100 kHz	< 500 nV <sub>rms</sub> /Hz <sup>0.5</sup>

## Reference Input

Frequency Range	1 Hz .. 10 MHz (for TTL) and 1 Hz .. 2 MHz (for sine wave)
PLL Locking Time	< (100 ms+ 10 Cycle)
Phase Error	< 4 deg @ f = 1 kHz
Input Amplitude	TTL or sine signal > 100 mV <sub>rms</sub> For some frequencies, up to 1 V <sub>rms</sub> sine wave amplitude might be required
Input Impedance for TTL	1 MΩ
Input Impedance for sine	1 MΩ (small signals) 1 kΩ (large signals)

### 1.2.2 OSCILLOSCOPE:

#### Available Channels

Lock-in channels of 1<sup>st</sup> input  
external A/D-channels  
internal data channels  
internal D/A-channels  
Lock-in channels of 2<sup>nd</sup> input

RA,  $\varphi_A$ , XA, YA  
AD1 to AD8  
f,  $U_{ac}$ , In  
DA5 to DA8  
RB,  $\varphi_B$ , XB, YB in "A&B"  
Rn,  $\varphi_n$ , Xn, Yn in "A" & "A-B"  
Noise

#### Mathematical Channel

Maximum Number of Simultaneously  
Displayed Channels  
Sampling Rate

4  
39.062 kHz

### 1.2.3 AUX IN:

Number of Available Channels  
Input Range  
Sampling Rate  
Resolution

8  
 $\pm 10$  V  
39.062 kHz  
24 Bit

### 1.2.4 AUX OUT:

Number of Available Channels  
Output Range  
Group Delay  
Output noise

8  
 $\pm 10$  V  
1  $\mu$ s  
< 500 nV<sub>rms</sub>/Hz<sup>0.5</sup>

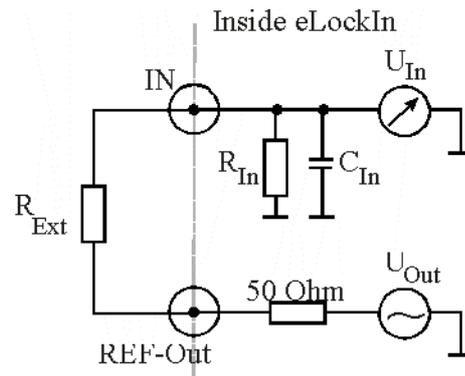
## 1.3 TYPICAL DATA

### 1.3.1 INPUT IMPEDANCE

External Impedance:  $R_{ext} = 1 \text{ MOhm} \parallel 15 \text{ pF}$   
between input and output

Potential divider:

$$\frac{X_{In}}{U_{In}} = \frac{X_{ext}}{(U_{out} - U_{In})} \quad (1)$$



Input and Output impedance are parallel circuits of a resistor and a capacitor, e.g.:

$$\frac{1}{X_{In}} = 1/X_{C-In} + 1/R_{In} = 2\pi f C_{In} + 1/R_{In} \quad (2)$$

At  $f = 1 \text{ kHz}$ , the ohmic part is much larger than the capacitive part:

$$R_{In} \ll \frac{1}{2\pi \cdot 1 \text{ kHz} \cdot C_{In}} \quad \text{Thus:} \quad R_{In} = \frac{R_{ext} U_{In}}{(U_{out} - U_{In})} \quad @ \text{ 1 kHz.}$$

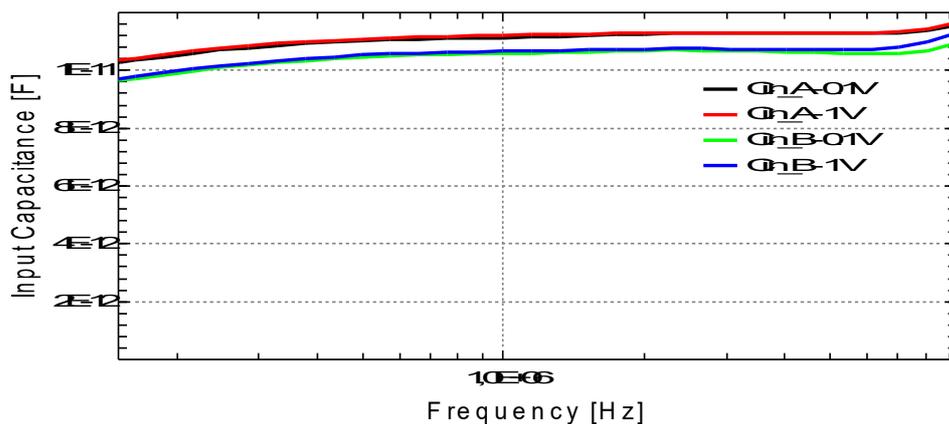
$C_{in}$  results from (1) and (2):

$$C_{In} = \frac{1}{2\pi f} * \left( \frac{(U_{out} - U_{In})}{(X_{ext} U_{In})} - \frac{1}{R_{In}} \right)$$

with

$$X_{ext} = \frac{1}{1/X_{C-ext} + 1/R_{ext}} = \frac{1}{2\pi f C_{ext} + 1/R_{ext}}$$

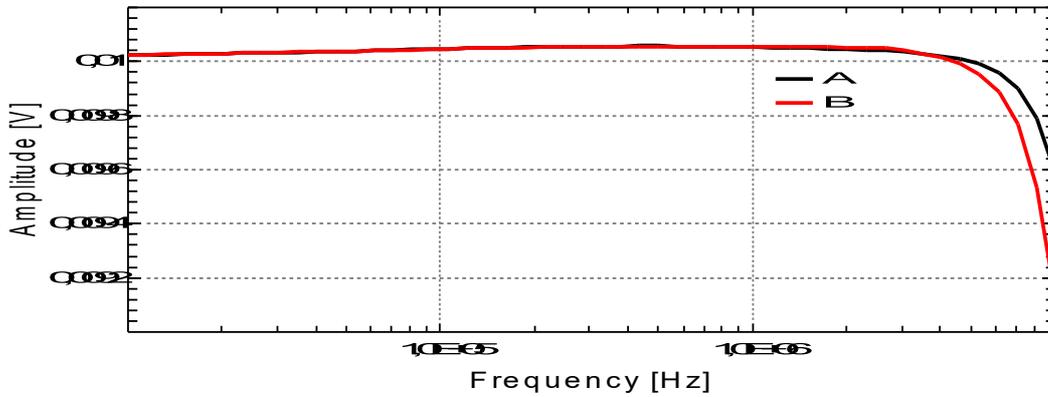
Results:



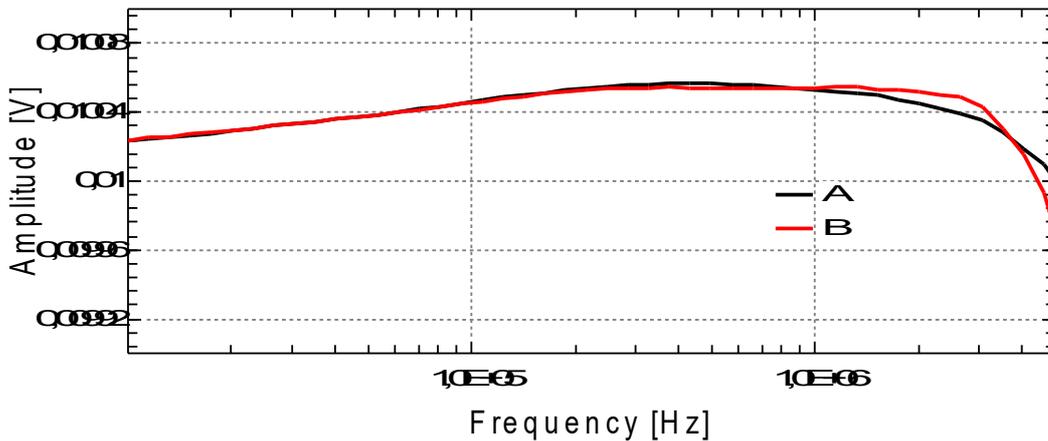
Bereich	0.1 V	1 V	10 V
Rin Channel A	1.12 MOhm	1.09 MOhm	350 KOhm
Rin Channel B	1.17 MOhm	1.15 MOhm	369 KOhm

### 1.3.2 TYPICAL FREQUENCY CHARACTERISTICS

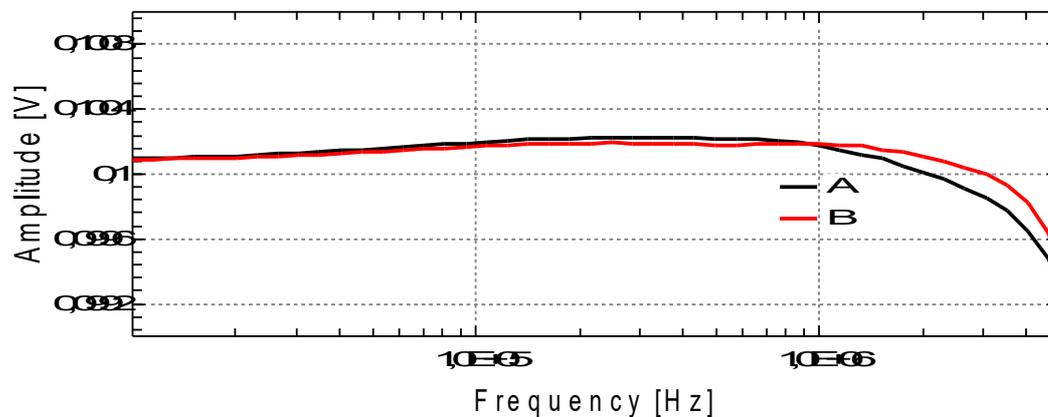
- Connect Input to Output
- Measure  $R(f)$  and  $\varphi(f)$



Input range: 0.1 V, A&B, Amplitude: 10 mV, error range: - 10 % ... +2 %



Input range: 0.1 V, A&B, Amplitude: 10 mV, error range: - 1%.. +1 %, 10 kHz to 5 MHz



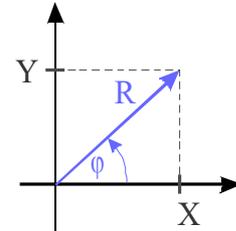
Input range: 1 V, A&B, Amplitude: 100 mV, error range: - 1 %.. +1 %, 10 kHz to 5 MHz  
For more data on this specific instrument, please check test sheet section.

## 2 LOCK-IN BASICS

### 2.1 LOCK-IN AMPLIFICATION

The lock-in amplifier is a measuring instrument for extracting and amplifying low signals from noise by means of a correlation analysis. For this method, it is necessary to know about the time dependence of the signal to be measured. Practically this signal is modulated at a certain reference frequency  $\omega_{ref}$  and the resulting frequency spectrum is simultaneously analyzed at  $\omega_{ref}$ . For analyzing the input signal the lock-in amplifier transforms the time-dependent signal at  $\omega_{ref}$  into the frequency domain and averages it during the time  $\tau$ . In other words, the lock-in amplifier can be described as a narrow-band amplifier with the center frequency  $\omega_{ref}$  and a bandwidth that is determined by the time constant  $\tau$ .

A schematic diagram with the basic components of the lock-in amplifier is shown in Figure 5 on page 5. First of all the received signal is amplified and digitized. Divided into two separate channels, the signal is multiplied by the mentioned reference signal (with frequency  $\omega_{ref}$ ) and the  $90^\circ$  phase-shifted reference signal respectively. The reference signal is generated with the lock-in's digital oscillator. An additional implemented phase displacement  $\Psi$  enables to compensate phase differences caused by the measuring equipment. After multiplication, the resulting signals are low-pass filtered and provide now information about the real part  $X$  and the imaginary part  $Y$  of the analyzed signal relating to the phase position of the reference signal. Out of them the amplitude  $R$  as well as the relative phase shift  $\varphi$  are calculated.



### 2.2 MATHEMATICAL DESCRIPTION

In this chapter, a short mathematical description of the multiplication and integration steps as the key procedures of a digital lock-in amplifier is given.

For instance, one tries to analyze the noisy signal  $s(t)$  that depends on a value  $v(t)$ .  $v(t)$  itself may consist of a constant part  $v_0$  and an additional periodic time-dependent part with the amplitude  $v_1$  and the frequency  $\omega_{ref}$ :  $v(t) = v_0 + v_1 \cos(\omega_{ref} t)$ . The latter term modulates the analyzed signal amplitude  $s(v(t))$ .

Then,  $s(v(t))$  can be developed into a Taylor series:

$$s(v_0 + v_1 \cos(\omega_{ref} t)) = \sum_{k=0}^{\infty} s^{(k)}(v_0) \cdot \frac{v_1^k}{k!} \cdot \cos^k(\omega_{ref} t) \quad (1a)$$

$$\text{with the derivatives } s^{(k)}(v_0) = \left. \frac{d^k s}{d v^k} \right|_{v=v_0} \quad k \in \mathbb{N} . \quad (1b)$$

Inside the lock-in amplifier  $s(v(t))$  is multiplied by the reference signal with frequency  $\omega_{ref}$  and integrated over the time  $\tau$ . That means, in order to determine the value of  $s(t)$  at the time  $t$ , all measured values in the past time interval  $\tau$  have to be taken into account. Therefore  $\tau$  should be a multiple of  $2\pi/\omega_{ref}$  or bigger because it should contain at least one period of oscillation to receive correct results.

Beside the modulation at  $\omega_{ref}$ , modulations of  $s(t)$  at the frequencies  $m \cdot \omega_{ref}$  with  $m = 1,$

2, ... (1<sup>st</sup>, 2<sup>nd</sup>, ... harmonic) are expected. For that reason the lock-in amplifier calculates the following Fourier components  $S(m \cdot \omega_{ref})$  in the frequency domain from the input signal  $s(t)$  in the time domain:

$$S(m \cdot \omega_{ref}) = \frac{1}{\tau} \int_{-\tau/2}^{\tau/2} s(v(t)) \cdot e^{-im\omega_{ref}t} dt \quad (2)$$

Together with equation (1) one receives the desired dependency of the measured signal on the value of  $v_0$  at the frequencies  $m \cdot \omega_{ref}$ :

$$S(m \cdot \omega_{ref}) = \sum_{k=0}^{\infty} s^{(k)}(v_0) \frac{v_1^k}{k!} K_m^k \quad (3a)$$

with

$$K_m^k = \frac{1}{\tau} \int_{-\tau/2}^{\tau/2} \cos^k(\omega_{ref}t) \cdot e^{-im\omega_{ref}t} dt \quad (3b)$$

Although the factors  $K_m^k$  converge to zero for  $k$  to infinity, some factors remain different from zero. Series elements of the  $n^{\text{th}}$  order are also negligible if the corresponding  $n^{\text{th}}$  derivative of  $s(v_0)$  is negligible, too.

The factors  $K_m^k$  for the first four harmonics are shown in the table below:

m	k =									
	1	2	3	4	5	6	7	8	9	10
1	$\frac{1}{2}$		$\frac{3}{8}$		$\frac{5}{16}$		$\frac{35}{128}$		$\frac{63}{256}$	
2		$\frac{1}{4}$		$\frac{1}{4}$		$\frac{15}{64}$		$\frac{7}{32}$		$\frac{105}{512}$
3			$\frac{1}{8}$		$\frac{5}{32}$		$\frac{21}{128}$		$\frac{21}{128}$	
4				$\frac{1}{16}$		$\frac{3}{32}$		$\frac{7}{64}$		

## 2.3 CHARACTERISTICAL CURVES OF A LOCK-IN AMPLIFIER

For filtering different information out of a received signal the corresponding frequency ranges ought not to overlap. The following experiment shows how small the difference between the frequencies should be that they can be analyzed independently:

A sinusoidal signal with a frequency of 100 kHz and an amplitude of 20 mV<sub>rms</sub> is fed in to the input of the lock-in amplifier. By sweeping the lock-in's reference frequency  $\omega_{ref}$  around 100 kHz the frequency spectrum of the input signal is analyzed. In Figure 3, the output of the lock-in amplifier is plotted in dependence on  $\omega_{ref}$  for different values of damping (Roll-Off = 6 dB/octave, 12 dB/octave and 24 dB/octave). The time constant  $\tau$  has been kept constant.

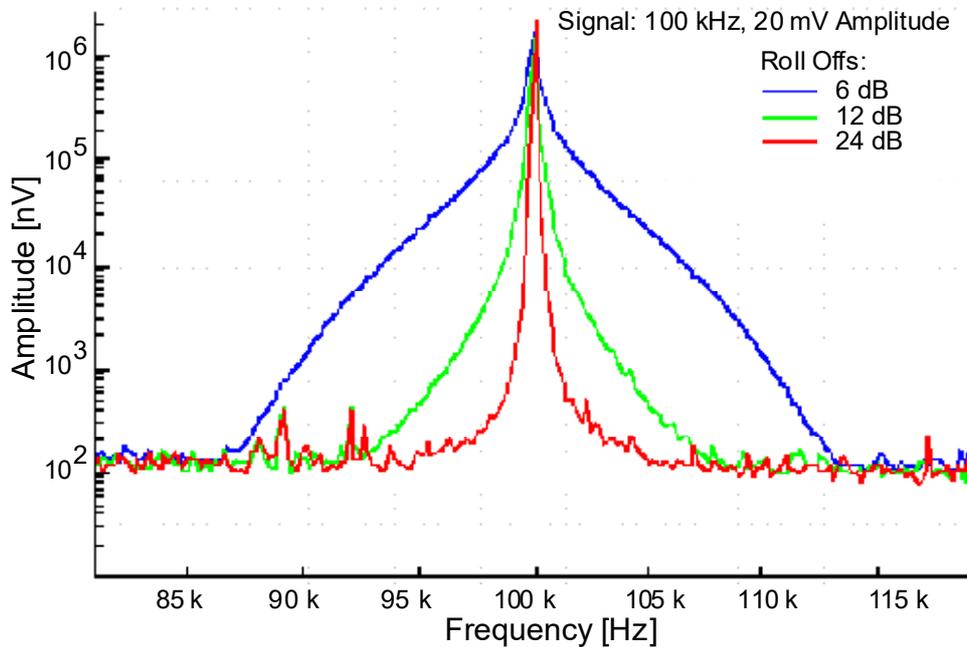


Figure 3 Output signal of a lock-in amplifier for a single frequency input signal of 100 kHz and 20 mV<sub>rms</sub> amplitude at the input. The time constants have been kept constant.

For a damping of 6 dB/oct, the measured amplitude at 95 kHz drops to approximately 1 % regarding to the maximal value at 100 kHz. For 24 dB/oct, the amplitude at 95 kHz is practically not measurable. So the curves show that the lock-in amplifier uses a very narrow frequency band for signal detection.

## 2.4 NOISE MEASUREMENTS

Lock-in amplifiers are capable to measure noise. They detect a signal at a certain center frequency  $\omega_{ref}$  with an equivalent noise bandwidth. For a Gaussian noise, the equivalent noise bandwidth of a real low pass filter is the bandwidth which passes the same amount of noise as a perfect rectangular filter with the equivalent noise band width.

The equivalent noise bandwidth of the **eLockIn** is determined by the time constant and the slope of the used Butterworth filter. It is calculated by

$$B_n = \int_0^{\infty} \frac{1}{1+\omega^{2n}} d\omega \quad . \quad (4)$$

The normalized Butterworth filter noise bandwidths are:

<b>Filter order</b>	<b>Bandwidth / <math>\tau</math></b>
1	1.570796
2	1.110721
4	1.026172

In order to measure noise spectra, the resulting data should be divided by the square-root of the used bandwidth.

The integrated spectrum is taken with a time constant of 10 ms and a slope of 24 dB (4<sup>th</sup> order) - bandwidth factor ~ 1,03. The related bandwidth is then  $1/(2*10 \text{ ms}) = 200 \text{ Hz}$ . In order to interpret the result as noise in units of  $\text{V}/\text{Hz}^{0.5}$ , the spectrum should be divided by  $14.14 \text{ Hz}^{0.5}$  (=  $\text{sqrt}(200 \text{ Hz})$ ).

## 2.5 EXAMPLE: ELECTRICAL FORCE MICROSCOPE

In this chapter electrical force microscopy (EFM) is briefly presented as an example for the application of lock-in amplifiers.

EFM is a related technique to the well-established atomic force microscopy (AFM). Its special aim is to detect electrical forces to learn something about the electrical properties of a surface, for example about the local distribution of surface potentials on electronic devices or different dopant concentrations in semi-conducting materials.

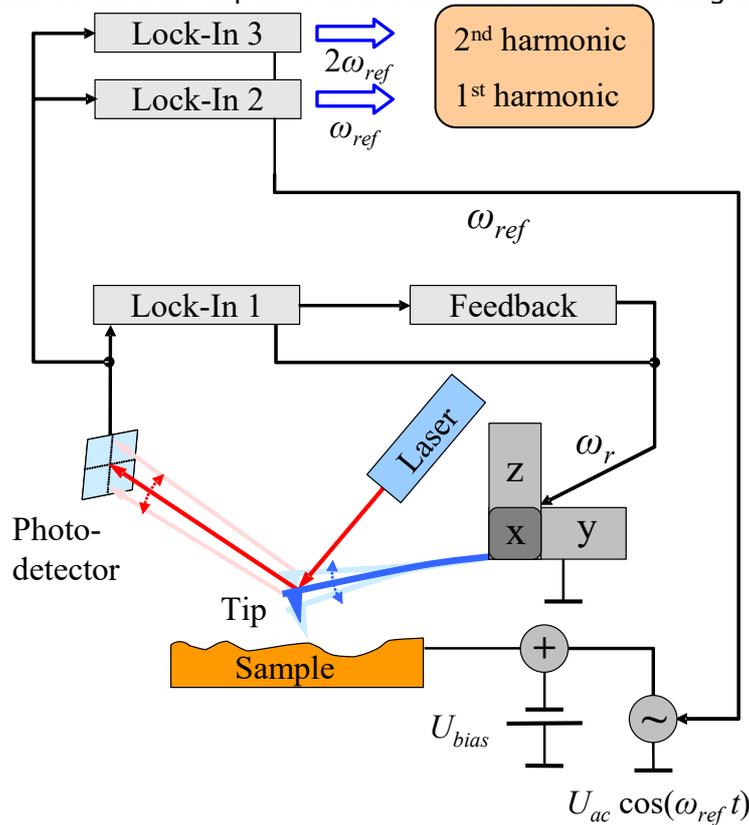


Figure 4: Schematic diagram of the EFM experimental setup.

The fundamental experimental setup shown in Figure 4 is based on a conventional atomic force microscope: In the non-destructive dynamic non-contact mode, an oscillating metallic tip fixed to a cantilever is scanning over the surface by means of a piezo scanning device. The distance between tip and sample can be controlled by monitoring the oscillation amplitude of the cantilever, because its value is influenced by short-range van der Waals forces. Therefore a reflected laser beam and a position-sensitive photo-detector are used. A lock-in amplifier analyzes the detector signal at the cantilever resonance frequency  $\omega_r$  and passes the determined amplitude value to a feedback control system that re-adjusts the tip-sample distance. The required displacement of the z-piezo can be recorded as topography signal.

In addition to the topography, EFM detects electrical forces, too. These forces are proportional to the derivative of the capacitance  $C$  of the tip-sample arrangement with respect to the tip-sample distance  $z$  and proportional to the potential difference  $U$  squared:

$$F_{el} = -\frac{1}{2} \frac{dC}{dz} U^2 . \quad (5)$$

A possible voltage dependency of  $C$  is neglected in this consideration. The voltage  $U$  contains a direct voltage part  $U_{DC}$  and an alternating voltage part  $U_{AC}$ :

$$U = U_{DC} + U_{AC} \cdot \cos(\omega_{ref} t) . \quad (6)$$

$U_{DC}$  consists of an additional applied bias voltage and, what is especially interesting from the physical point of view, potential differences caused by different electronic work functions and charges. For separating the impact of these electrical forces from other forces (e.g. van der Waals forces),  $U$  has to be modulated at the frequency  $\omega_{ref}$ . As a result of this, the measurable photo-detector signal is modulated, too.

Inserting equation (6) in (5) and using power-reduction formulas of trigonometric functions one can expect a force between tip and sample at the frequency  $\omega_{ref}$  as well as at  $2 \cdot \omega_{ref}$ . Figure 5 confirms this prediction: It shows the frequency spectrum of the cantilever oscillation with clear signals at the mechanical excitation frequency  $\omega_r$  and at the frequencies of the first and second harmonic of the electrical excitation.

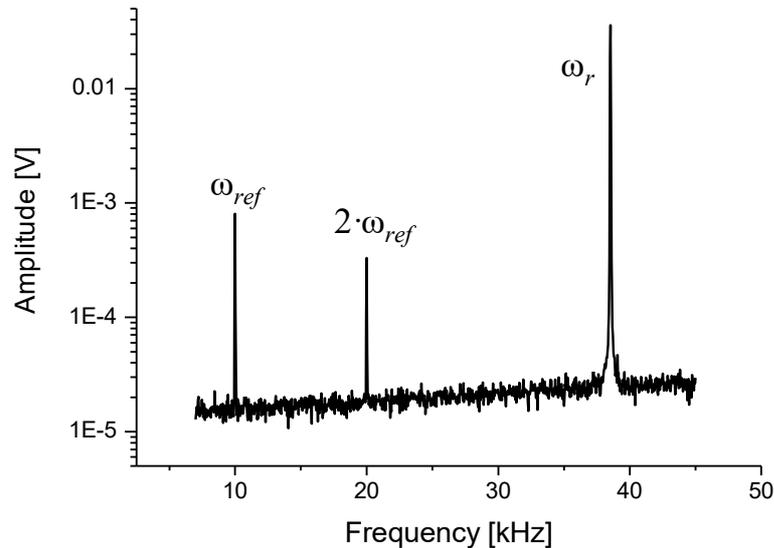


Figure 5: Frequency spectrum of the oscillating cantilever

Now two additional lock-in amplifiers (see Figure 4) can be used to analyze the photo-detector signal at the frequency  $\omega_{ref}$  and  $2 \cdot \omega_{ref}$  simultaneously. The measured amplitudes of the signals are proportional to the strength of the electrical forces.

# 3 USERS MANUAL

This manual part describes where to find and how to change a parameter. It also gives an overview of the front panel and rear panel structure. An abbreviation like [LockIn / Ref / f] means that the user has to press the buttons "LockIn" → "Ref" → "f" one after the other.

## 3.1 TECHNICAL INTRODUCTION

### 3.1.1 FRONT PANEL

#### ◆ Connectors / Switches:

- *In*: Signal input of the lock-in amplifier,
- *Power*: Main on/off button of the instrument,
- *Ref Out*: The lock-in amplifier reference output.

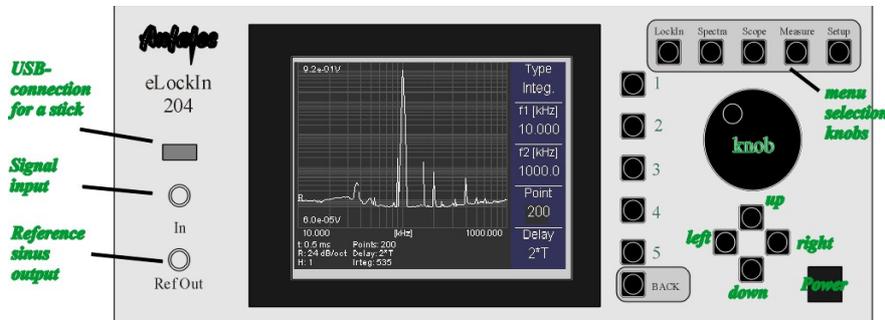


Figure 6: The front panel.

#### ◆ Fixed Navigation Buttons:

- **Knob**: Use this knob to change the selected parameter fast. When the knob is turned in a clockwise direction, the selected parameter will be increased. When it is turned counterclockwise, the selected parameter will be decreased. Use the **pushbutton** in the knob to switch between menu meanings.
- **Up / Down**: These two buttons increase / decrease the value of the selected parameter stepwise.
- **Left / Right**: These buttons move the cursor in the selected parameter input field one digit to the left or to the right. The digit that can be changed with the **Up** and **Down** buttons is always the marked digit.

#### ◆ Variable Navigation Buttons:

- **Button No. 1 to 5**: Selects the function left besides the button.
- **Back**: Goes one step upwards in the menu structure.

#### ◆ Menu Selection Buttons:

- **LockIn**: This is the control key to enter the **LockIn** mode. In the **LockIn** mode, parameters connected with the **LockIn** function can be changed. In addition, the channels to be displayed in the oscilloscope and spectra mode as well as the output

signal and the scaling of the DA-channels can be modified here.

- **Spectra**: This is the control key for the **Spectra** acquisition mode. The displayed channels are selected in the **LockIn** mode.
- **Scope**: Press this button to turn the instrument into Oscilloscope mode. The displayed channels are selected in the **LockIn** mode.
- **Measure**: This menu provides certain functions which make the usage of the instrument easier, such as the frequency selection in a recorded spectrum.
- **Setup**: In this menu the user can change some basic setup parameters, such as the IP address of the device or the warning signal.

**Note: When the parameters reach their maximum or minimum value, the increase or the decrease actions have no effect.**

### 3.1.2 SCREEN ORGANIZATION

The screen is divided into three parts:

- the main screen
- the menu display and
- the parameter display.

While the menu display helps to navigate through the parameter menus, the main screen contains information, which is specific for the selected mode of the device. The parameter display provides information, which is usually hidden behind several menu entries.

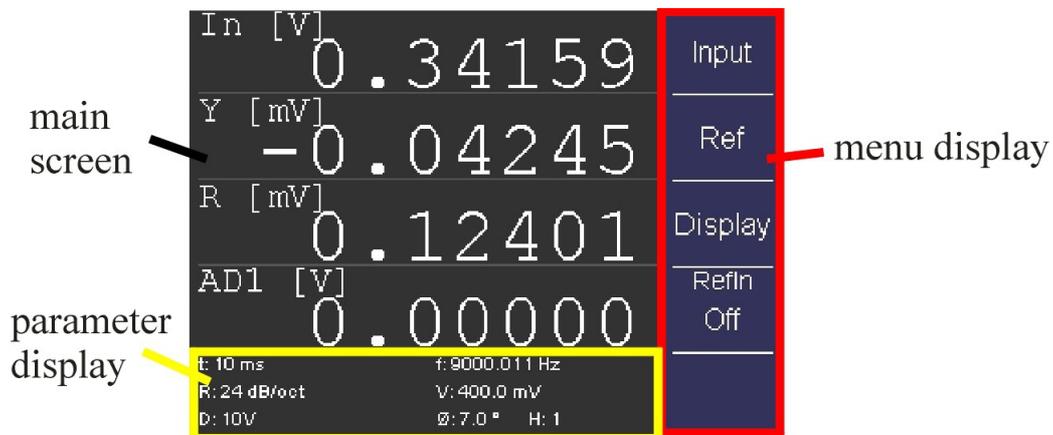


Figure 7: Screen organization at the example of the LockIn menu.

The main screen shows:

- ◆ current values of selected channels in **LockIn** mode,
- ◆ acquired data in **Spectra** mode and Oscilloscope mode and
- ◆ system parameters in the **Setup** mode.

The menu gives access to parameters. The menu trees are shown in the Figures 7 to 8. Note that each menu entry consists of a name and some have a value.



This menu entry consists of a **name** "f1 [kHz]" and a **value** "10.000". The value can be changed with the **Up** and **Down** functions.

This entry consists of a **name** only. It usually leads to a sub-menu.



### 3.1.3 THE MENU TREE

The five top base menus are selected with the Menu Selection buttons displayed in Figure 6 on page 18. The sub-menus are shown in the following pictures:

When [LockIn] is pressed, the first line of the menu structure mentioned in Figure 7 is shown in the menu screen, which means **Input** in the first line, **Ref** in the second line and **Display** in the third line.

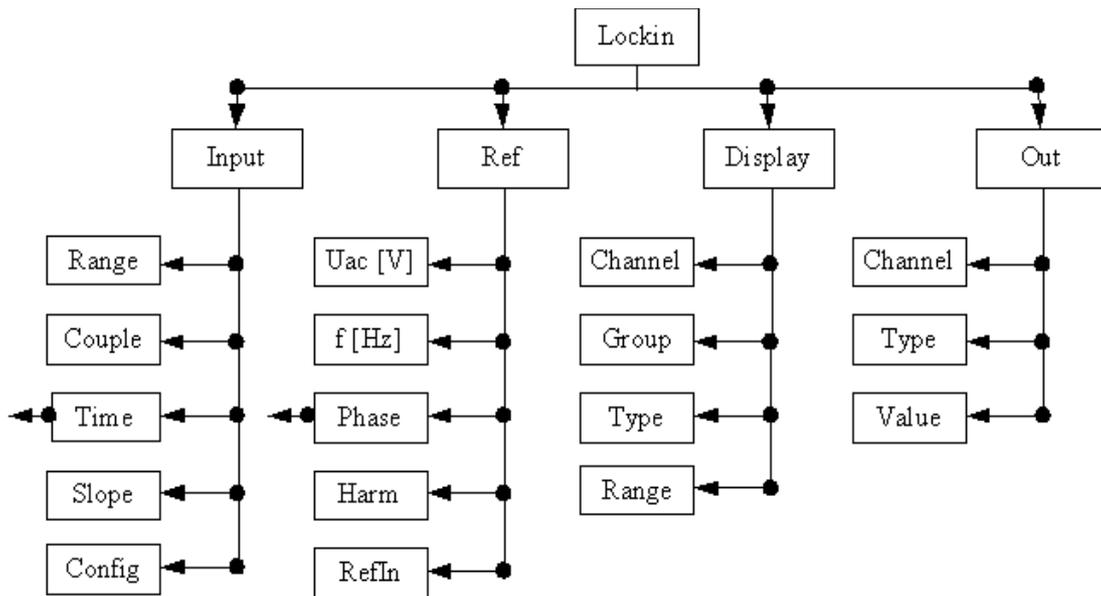


Figure 8 Menu structure in [LockIn] for standard operation mode (for dual reference mode please refer to page ).

When now **Input** is selected, the menu items will change to **Range** in the first, **Couple** in the second, **Time** in the third and **Slope** in the fourth line. Out of these four items, the selected item is perceptible from the darker background color. In order to select another item, press the button right of it once. In order to change a number displayed below the name, use the **Left / Right** functions to select the digit and the **Up / Down** function to change it.

The corresponding structures of the **Spectra** and the **Scope** menus are shown below:

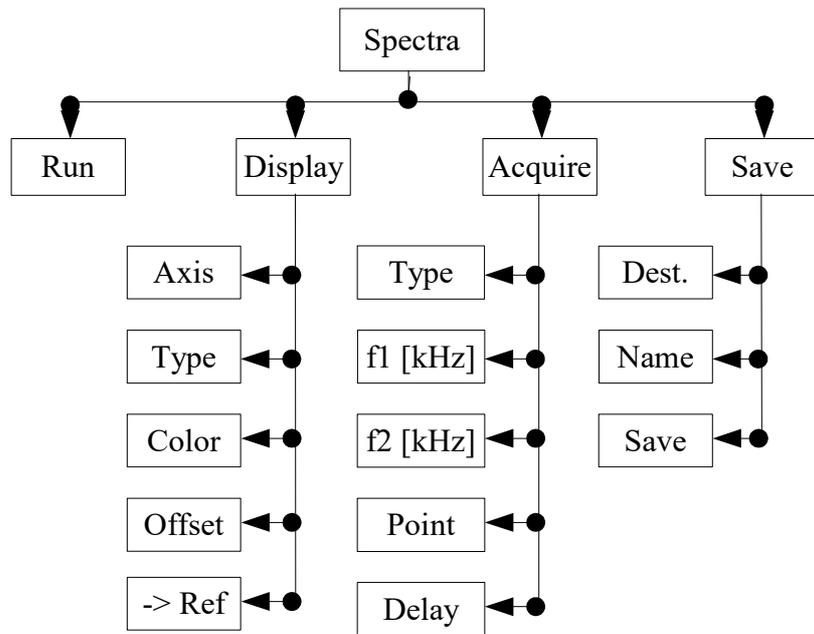


Figure 9: Menu structure of the Spectra mode.

The knobs **Measure** and **Setup** do not provide a deep menu structure. In order to understand the meaning of each entry in the menus, follow the manual structure starting with the **LockIn** menu in section 3.2 .

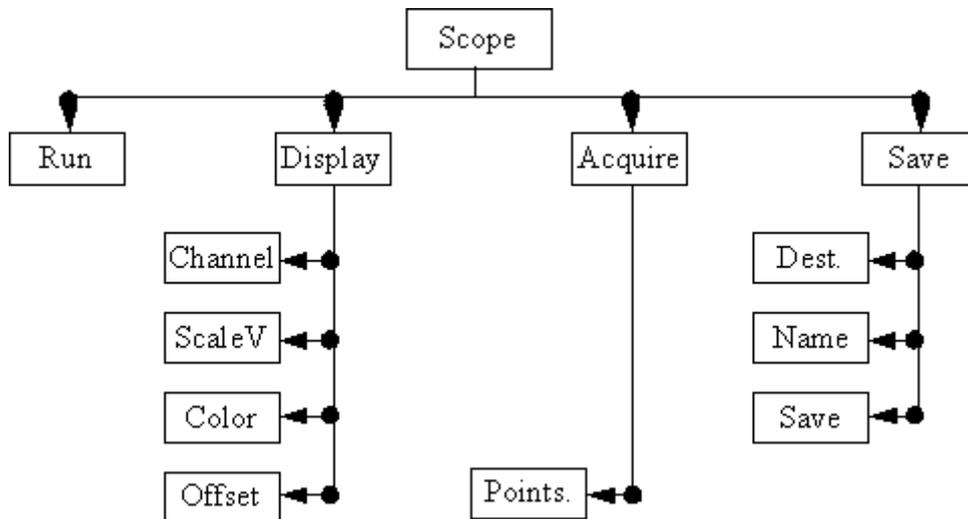


Figure 10: Oscilloscope menu called with the Scope button.

### 3.1.4 REAR PANEL

The back side panel provides

- 8 x BNC connectors for the internal A/D converters (-10 V .. 10 V) "Aux In"
- 4 x BNC for displaying the selected channels scaled as analogue outputs (DACs) "Display"
- 4 x BNC for Aux Out DACs "Out"
- 2 x BNC Ref-In connector for the reference input ("Ref In 1" for standard PLL, "Ref In 2" for LoL mode option)
- 2 x BNC for monitoring the amplified input signals A and B called "Monitor A" and "Monitor B"
- BNC Out for 10 MHz reference clock "10 MHz Out"
- BNC Out for TTL reference output "TTL Out"
- BNC Out for sine wave reference out on 2<sup>nd</sup> lockin frequency "Ref2"
- LAN (RJ45) network connector for remote control
- the 100 -240 V AC power supply (tested from minimum 100 V AC)
- HDMI – if HDMI is connected to a monitor, the front display of the eLockin will be off. Needs to be plugged in before eLockin is powered on.
- DVI – monitor connection for debugging purposes
- 2 x USB storage device, keyboard or mouse connection



Figure 11: Rear-Panel of the **eLockIn** (picture might vary from current revisions). The ground connector above the power supply allows to separate the protective ground from the housing and thus increase the resistance between them to 1 MOhm.

## 3.2 THE LOCKIN MENU [\[ LOCKIN \]](#)

### 3.2.1 INPUT SETTINGS [\[ LOCKIN / INPUT \]](#)

#### **Input Range** [\[ LockIn / Input / Range \]](#)

This setting changes the input amplification of the lock-in input stage by two hardware switches controllable from the software. These factors are according to the range value of "high dynamic reserve", "normal" and "low noise". Together with the input amplifications, dynamic range and sensitivity change. The lower the dynamic range, the higher the sensitivity is. The strategy for selecting a suitable **range** is: Choose the smallest value which is just bigger than the maximum input signal.

Example: When the maximum input signal is 300 mV<sub>pp</sub>, the "low noise" input range will yield an overflow. The "normal" and "high dynamic reserve" input ranges are both possible, but the "normal" range gives the better sensitivity.

**Note: If the Beep function [ Setup / Beep ] is enabled, the instrument beeps every time it detects an input overflow.**

### Input Coupling [ LockIn / Input / Couple ]

If the specification of the instrument allows it, this option switches between **DC** coupled input and **AC** coupled input.

**Note: The 3dB corner frequency of the input high pass is around 2Hz. Reference frequencies around 2 Hz and below may cause misleading results.**

### Time Constant [ LockIn / Input / Time ]

This parameter is related to the low pass filters in the lock-in amplifier. It describes the 3dB-cut-off frequency  $f_c$  of the filter as:

$$\tau = \frac{1}{f_c} .$$

$\tau$  might be defined as **time period** without synchronization to the period length related to the currently used as reference frequency. Then, the **eLockIn** integrates over the time  $\tau$ . Those filters refer to the entries in [Tau] and [Tau1].

Starting from software version 1.15, the entry in [Sync] acts in series with the Butterworth low pass filters. The Sync-Filter integrates over an integer number of periods synchronized to the current period length. This allows a time-optimized data acquisition at low frequencies.

The sync-filter produces an output, which contains steps (Example: at 1 kHz frequency with Sync=1, one gets a new output value exactly after 1 ms. Thus, the analogue output, which is sampled with 1 MHz, provides 1000 constant values before it jumps to the next value.). The Butterworth filter smooths the edges of this staircase-like output.

For high-frequencies, it might result in better noise performance, if the sync-filter is switched off.

Tau A  
10 ms

Sync  
1

Tau1  
= Tau

**Enable Sync-mode:** go to the **Sync** entry (2<sup>nd</sup> from the top) and turn the **knob** right or use **up**. The entry **Off** turns into a number (could be **1,2, ... ,200**), which is the amount of periods to be used for integration.

**Disable Sync-Mode:** go to the **Sync** entry and turn the **knob** left or use **down**, until the **number** turns into **Off**.

The time constants below the entries **Tau** and **Tau1** always stay active.

Please note the following:

- When the external reference is used, the Sync-filter might increase the measured signal noise.
- When the time constant is larger than Sync/f, then the Sync-filter has not much effect.

In time based mode, **Tau** equals the integration for channel A (**Input/Config = A&B**) or the n-th harmonic (**Input/Config = A** or **A-B**), while **Tau1** equals the integration for channel B (**Input/Config = A&B**) or the 1<sup>st</sup> harmonic (**Input/Config = A** or **A-B**). The two entries can be changed independently on each other.

In both entries **Tau** and **Tau1**, one can chose between a time (**0.1 ms, 0.2 ms, ... 500 s, 1 ks**) and a kind of frequency related integration (**10/f** and **100/f**). The later still uses the Butterworth filter, but chooses the closest time constant for the currently selected frequency during spectra acquisition. This feature, however, is useful for noise measurements only, because the switching between different filter settings causes peaks in the acquired spectra.

For the acquisition of spectra with varying integration adapted to the currently selected frequency, one has to enable the **Sync** filter. In this mode, the Butterworth filters are disabled and the **Slope** has no meaning anymore, which is displayed as **Sync** in the **Slope** menu entry.

For instruments with the LoL-Mode option, **Tau1** is used for the 2<sup>nd</sup> demodulation.

### Time Constant [ LockIn / Input / Time ] → on a time base

Integration just on a timebase means that the time constant is not related to frequency at all.  $\tau$ , the time constant, defines the length of the time interval taken into account for the low pass filtering. The **eLockn** employs a Butterworth-type low-pass filter. Together with the **Slope**, which is nothing else than the order of the low pass filter, the bandwidth and the raise time of the filters are given as follows:

<b>Roll-Off</b>	<b>Formula for bandwidth</b>	<b>Raise time / settled time to 90 %</b>	<b>Raise time / settled time to 99 %</b>	<b>Raise time / settled time to 99.9 %</b>	<b>Overshot</b>
6 dB/oct	$\frac{\pi}{2\tau}$	0.43 $\tau$	0.74 $\tau$	1.11 $\tau$	0,00 %
12 dB/oct	$\frac{\pi}{2\sqrt{2}\tau}$	0.43 $\tau$	0.52 $\tau$ / 1.05 $\tau$	0.53 $\tau$ / 1.64 $\tau$	4.32 %
24 dB/oct	$\frac{\pi}{8\sin(\frac{\pi}{8})\tau}$	0.63 $\tau$ / 0.96 $\tau$	0.70 $\tau$ / 1.66 $\tau$	0.70 $\tau$ / 2.72 $\tau$	10.8 %

The time constant can be set in logarithmic steps (1-2-5) between 0.1 ms and 1 ks.

### Syncron Filter [ LockIn / Input / Time ] → synchronized integration

In synchronized low-pass filtering, the integration time is strictly connected to the current center frequency. It is given in multiples of 1/f and can vary between 1/f and 200/f.

The parameters "bandwidth" and "raise time" are not applicable for the synchronized filtering, anymore. The result is provided the moment it is valid. Thus, the oscilloscope might show staircase-like outputs for lower frequencies with accordingly high multiples

of the period.

During spectra acquisition, the integration time is constantly adapted to the current frequency, so that the spectrum takes more time at low frequencies.

### **Roll-Off or Slope [ LockIn / Input / Slope ]**

This parameter is related to the order of the low pass filter. We can select among 6 dB/oct, 12 dB/oct and 24 dB/oct. The 6 dB/oct selection equals a low pass filter slope of 1<sup>st</sup> order, while the 12 dB/oct and 24 dB/oct equal the 2<sup>nd</sup> and 4<sup>th</sup> order, respectively.

### **Config [ LockIn / Input / Config ]**

This entry switches between the different input modes, if available from the model configuration.

<b>Mode</b>	<b>Description</b>
<b>A</b>	The input A is amplified in two different stages and given to the two ADC inputs of the digital lockin amplifier design, simultaneously. This reduces the noise. The 4-phase lockin evaluates two phases at the 1 <sup>st</sup> harmonic and two phases at any higher harmonic.
<b>A-B</b> <i>Models 203, 204/2, 205/2</i>	The two inputs A and B are amplified separately in a 1 <sup>st</sup> stage. Then, the difference between them is built electronically by a differential amplifier. The 2 <sup>nd</sup> amplification stage is applied to this difference and fed into the input ADC2. The lockin measures two phases at the 1 <sup>st</sup> harmonic and two phases at any higher harmonic.
<b>A&amp;B</b> <i>Models 203, 204/2, 205/2</i>	The two inputs A and B are amplified separately and fed into the two inputs ADC1 and ADC2. One measures with two phases at the input A and with two further phases at the input B. Both inputs operated at the same frequency (1 <sup>st</sup> <b>or</b> higher harmonic) with the same input amplification (input gain).
<b>I</b> <i>current input</i>	Switches to the current input, <b>if available</b> . In this input type, only two amplification stages are available. Thus, 0.1 V and 1 V input range equal each other.

No matter which input mode is chosen, the **Time** Constant in [ LockIn / Input / Time ] and the **Slope** in [ LockIn / Input / Slope ] apply to all measured signals (all four phases).

### **3.2.2 REFERENCE OUTPUT [ LockIn / REF ]**

The internal digital sine stream is built from a look-up table. The building process depends on three parameters: The main frequency, the phase and the harmonic. The generated reference output signal relates to this internal stream through these three parameters. In addition to that, the reference output has the amplitude as an independent parameter.

### **Reference Output Amplitude [ LockIn / Ref / Uac ]**

The reference output is a sine signal. The parameter **Vac** is its RMS amplitude.

## Reference Output Frequency [ LockIn / Ref / f [Hz] ]

The frequency selected here is the reference frequency  $f_{Ref}$  of the reference output and middle frequency of the lock-in amplifier. The analysis of the signal is done at the  $h^{th}$  of its harmonics  $f_m$ :

$$f_m = f_{Ref} h$$

## Reference Output Phase [ LockIn / Ref / Phase ] → phase

Phase stands for the phase shift between the periodic excitation provided from the reference output or, in PLL mode, from the external trigger source and the detected oscillation at the input.

In case the harmonic  $h$  is equal to one, the number defined here is the phase difference between the reference output signal and the signal that is multiplied with the lock-in input. For higher harmonics, the a phase offset  $\Phi$  is added to the detected signal.

**AutoPhase** – an automated phase detection – is implemented. Select the phase channel and use the **[Measure]** function to set the currently detected phase as phase offset.

## Harmonic [ LockIn / Ref / Harm ]

This parameter indicates at which harmonic  $h$  of the reference frequency we measure.

## PLL Input [ LockIn / Ref / RefIn ]

In order to use an external reference source, the **eLockIn** provides an internal PLL circuit. This circuit locks onto the external reference. One can switch the reference input between high signal amplitudes (LV-TTL) and low signal amplitudes (sine wave) with the switch **RefPLL** in **[ Setup / MISC / RefPLL ]**.

This software knob **[RefIn]** switches the PLL **On** or **Off**. In case the reference input is **On**, the frequency in the parameter display shows the locking frequency as "PLL: **frequency-value**". In case the reference input is **Off**, the frequency value shown in the parameter display is the **frequency** of the internal reference.

### 3.2.3 DISPLAY SETTINGS [ LOCKIN / DISPLAY ]

In this sub-menu the user can choose which input signal should be displayed on the front panel. Moreover, the display sensitivity can be adapted separately for each input. At maximum the device can display four inputs. We refer to them as channels, however, they can originate from at least 27 different sources:

- lock-in output channels – **XA, YA, RA,  $\phi A$**  (signals connected to the lock-in functionality detected at the  $n^{th}$  harmonic) → **Group LIA n**
- lock-in input – **In** (signal taken directly from the front panel input) → **Group Misc**
- lock-in settings – **f** (reference frequency of the lock-in amplifier), **Uac** (reference output amplitude) → **Group Misc**
- A/D converter inputs – **AD1, AD2, AD3, ... AD8** → **Group ADC**
- D/A converter inputs – **DA1, DA2, DA3, ... DA8** → **Group DAC**
- quad-channel-LockIn output channels – **XB, YB, RB,  $\phi B$**  (if the **eLockIn** has the internal functionality of two different inputs) or **X1, Y1, R1, j1** (if the **eLockIn** has one input only or its input setting is set to Input A, only) – in later case, the

input signal is simultaneously characterized at the first and a higher harmonics.)

→ *Group LIA 1*

In order to classify this large amount of selections, the channels are organized in the **Groups**: *LIA n, Misc, ADC, DAC, LIA 1*. The **Group "Off"** stands for "no group" = "no signal".

### **Channel [ LockIn / Display / Channel ]**

In order to change the settings of the four channels, this button realizes the channel selection. All selections displayed in **Group, Type** and **Range** are valid on the selected channel only.

### **Group of Displayed Signal [ LockIn / Display / Group ]**

The large amount of possible input signals is organized in **Groups**: *LIA n, Misc, ADC, DAC, LIA 1*. The **Group "Off"** stands for "no group" = "no signal".

### **Signal Source [ LockIn / Display / Type ]**

The **Type** enables the source selection. All types described in Select Display Channels in the Lockin-Menu are available on all four **Channels**. The **Sources** selected in the first four channels are sent to the first four D/A outputs at the rear panel. In order to disable one channel, **Off** has to be selected.

### **Output Scaling [ LockIn / Display / Range ]**

This parameter changes the display sensitivity, but not the measurement sensitivity. If the real value is bigger than the range, "**-ovld**" will be displayed. Simultaneously, it changes the scaling for the output amplitude sent to the D/A converters. The chosen sensitivity is related to the maximum output value of the D/A converter, which is 10 V. Thus the D/A output is scaled to this display range.

Example: If channel 1 displays the lock-in amplitude R and its range is set to 1 mV, a detected amplitude R of 1 mV gives an output value of 10 V at the DA1 (rear panel). If the signal shrinks to 1  $\mu$ V, the output at the DA1 is only 10 mV. Because DA1 adds its own noise, its better to reduce the display range as much as useful to improve the output signal taken from the DA's at the rear panel.

## 3.2.4 **AUX OUT [ LockIn / Out ]**

In principle, this option allows to set the values provided at the eight output channels with 24 bit converters. With this, the implementation of feedback circuits for Kelvin Probe applications is prepared.

In the current instrument philosophy, the first four channels are always scaled with the display settings four the four channels. The AUX outputs AUX5 to AUX8 can be set to a certain voltage.

### **AUX Out Cannel [ LockIn / Out / Channel ]**

This option selects the channel out of the eight AUX channels, for which the settings **Type** and **Value** are controlled here.

### AUX Out Type [LockIn / Out / Type]

Set to **Scaled**, the related channels provides the output with respect to the setting given in [ LockIn / Display / Range ]. Set to **Const.**, a **value** can be set for the channel.

In the current state, the channels DAC1 to DAC4 are automatically set to **Scaled**, while the channels DAC5 to DAC8 are set to **Const.**

### AUX Out Value [LockIn / Out / Value]

For the channels 5 to 8, a value given in V can be set here. In the remote control regime, this function can be used to control external parameters of a set-up and thus to program more enhanced measurement procedures.

## 3.3 THE SPECTRA MENU [SPECTRA]

While the **LockIn** mode displays values as numbers, the Spectra mode allows to collect such information in dependence on the frequency. The data are shown as graphs and can be saved as ASCII files for later analysis.

**Note: The channels in spectra mode are the same as selected in the LockIn mode. The channel selection is only possible in LockIn mode (see section 3.1.3 on page 18).**

In addition to the channels available in **LockIn** mode, the **Spectra** mode provides a reference channels **Ref**, which is used to store spectra temporarily.

Example: In order to measure a frequency spectrum of the amplitude R of an oscillating cantilever, take care that the following settings are correct:

- R is selected as source for channel 1 (3.2.3: Channel),
- channel 1 is really displayed in a visible color (see 3.3.2: Color), and
- the display of the other three channels is disabled (see 3.3.2: Color) to avoid confusion in the shown curves.

### 3.3.1 RUN/STOP [SPECTRA / RUN ] OR [ SPECTRA / STOP ]

The first menu item in the menu toggles the state of the instrument in **Spectra** mode. To **Start** the spectrum acquisition, press button **1** (see Figure 6) when the node displays **Run**. To **Stop** the acquisition process, press **Stop**.

In order to acquire spectra in logarithmic steps, the **Type** of the frequency axis **freq** has to be set to **Log**.

### 3.3.2 DISPLAY [SPECTRA / DISPLAY ]

In **Spectra** mode, the lines are superimposed on each other. The scaling on the screen is always automatically, which means they are scaled to fit the screen area independently on each other. The user can switch channels off (see 3.3.2: Color), control the color of each line and whether they are displayed linearly or logarithmic (see 3.3.2: Type). The line can be shifted by changing a parameter called offset (see 3.3.2: Offset).

### Axis [Spectra / Display / Axis ]

This menu item gives access to the display parameters of all the selected channels (see

3.2.3). Besides that, the parameters of one reference channel whose data are saved on the memory and the frequency axis **freq** are available. When an axis is selected, its display parameters are shown in the lower nodes (**Type**, **Color**, and **Offset**).

#### Type [ Spectra / Display / Type ]

This menu item shows either the text **Lin** (linearly scaled) or **Log** (logarithmic scaled). If the selected axis is **freq** (frequency), this parameter equals also the sweeping type of the frequency.

#### Color [ Spectra / Display / Color ]

This menu item shows the color of the selected axis. If the **Color** is **Off** then the channel is not displayed. This menu item does not exist if the selected axis is **freq**.

#### Offset [ Spectra / Display / Offset ]

This shifts the selected curve along its axis. This menu item does not exist if the selected axis is **freq**.

#### Save Reference Channel [ Spectra / Display / -> Ref ]

If any other than the reference channel is selected in **[Spectra/Display/Axis]** and the reference channel is empty, **None** is displayed under the menu item **-->Ref**. Press once to store the data of the selected axis into the reference channel. Then, the name of the stored channel (e.g. **R\_A**) is shown under **-->Ref**.

Reference data stored in the reference channel and can be overwritten any time by simply pressing **-->Ref** once more.

After reference data exist, **[Spectra/Display/Axis]** offers an entry **Ref**. If **Ref** is selected in **[Spectra/Display/Axis]**, the name of the menu entry "**-->Ref**" is changed into "**Scale**". Now, the setting for the vertical scaling of the reference channel can be either linked to any other available channel. Then, this menu entry shows the name of the linked data channel (**Ch 1**, **Ch 2** ... or **Ch 4**) and both linked channels (e.g. **Ch2** and **Ref**) define the minimum/maximum value for their vertical scaling together.

Otherwise, the scaling of the reference channel can be chosen to be independent on the other channels. Then, the menu entry **[Spectra/Display/Scale]** is set to **auto**.

### 3.3.3 ACQUIRE [ SPECTRA / ACQUIRE ]

For the spectrum acquisition, the following parameters can be set:

- start frequency **f1**
- stop frequency **f2**
- number of data **Points** in the spectrum (up to 100000 points)
- acquisition **Delay** (is connected to the time constant)

Furthermore, there are several possibilities to handle spectra taken in sequence. Therefore we introduced a **Type** of spectrum acquisition, which allows to repeat acquisition automatically (continuous) or to sum up spectra (integrated).

### Type [ Spectra / Acquire / Type ]

This parameter has three options:

- ◆ **Single**: The specified spectrum is acquired just once. When it is finished, the acquisition process stops.
- ◆ **Conti.**: which stands for continuously. The specified spectrum is acquired as long as **Run** is selected. When it is finished, the acquisition process restarts.
- ◆ **Integ.**: is an integrating acquisition of spectra. The spectra are taken continuously, however, the final spectrum is the average spectrum calculated from all taken spectra. The integration restarts with every **Run / Stop** selection.

### f1 and f2 [ Spectra / Acquire / f1 ] and [ Spectra / Acquire / f2 ]

**f1** = start frequency = first point in the spectrum.

**f2** = stop frequency = last point in the spectrum.

### Point [ Spectra / Acquire / Point ]

This is the number of points in the spectrum. The specific frequencies in the spectrum are calculated based on this number, **f1**, **f2** and the sweeping type (see 3.3.2: Type).

### Delay [ Spectra / Acquire / Delay ]

This parameter controls the delay between the time when the instrument sets a new frequency and the time it takes the output data. Its length is given in multiples of  $\tau$ , the time constant (see Time Constant). In order to check useful values with respect to the settle time, refer to the table on page 24.

## 3.3.4 SAVE [ SPECTRA / SAVE ]

After the acquisition process finished, data can be saved to the hard disk or the USB flash drive for later analysis. The maximum volume of the hard disk for storing data is 32 megabytes. Up to 100 data files (i.e. **Dat1**, **Dat2**, .. **Dat100**) can be kept in the hard disk. If the USB flash drive is used, it should be compatible with the USB v1.1 specification. The USB flash drive should be formatted as FAT32.

### Destination Storage Media [ Spectra / Save / Dest. ]

This parameter chooses the storage media to save the data to. The options are:

- **HD**: The data will be saved onto the internal hard disk.
- **USB**: The data will be saved onto the USB flash drive.

If the hard disk is selected, the last-saved file name is displayed in the **Name** item. If that file is still in the memory, **Exist** will be displayed in the **Save** item. In case the file does not exist, **Empty** will be displayed in the **Save** item. When USB is selected, the instrument will check whether a working USB flash drive is connected in one of the USB connectors or not. If it finds one, the **Name** and **Save** items will be displayed in the same way as for the **HD**. If no USB flash drive is available or detected, **Error** will be displayed in the **Save** item.

### Choose a File Name [ Spectra / Save / Name ]

The file name is not dependent on the storage medium. When a storage medium is selected, the directory **data** will be created in the root directory of the storage medium. The user can only choose from 100 predetermined names: **Dat1**, **Dat2**, ... ,

**Dat100.**

### **Save the File [ Spectra / Save / Save ]**

When a valid storage device and a name is selected, the file can be saved. If the user select a file which already exists, it will be overwritten without notice. Just press button **3** (see Figure 6) next to the **Save** menu item to activate the action. If the file is successfully saved, the **Save** item will display **Saved!** , otherwise **Error** will be displayed (e.g. if the storage media is full).

## **3.4 OSCILLOSCOPE [ SCOPE ]**

For the oscilloscope function, the input channels of the **eLockIn** are taken with 40 kHz data acquisition rate and displayed vs. time. Additionally, it is possible to Fourier transform these data, so that a life spectrum (as if operated like a spectrum analyzer) appears.

The switch between "display vs. time" and "display vs. frequency" is done when the time/freq axis is chosen as channel.

Please note that the data shown are always the same data that have been selected in the lockin menu.

### **3.4.1 RUN/STOP [ SCOPE / RUN ] OR [ SCOPE / STOP ]**

As in the **Spectra** mode, this menu item is just a switch between on and off state of the data acquisition in the oscilloscope. In the default state, the instrument waits for instructions and does not acquire data. Thus, **Run** is displayed and meant as instruction. To start the data acquisition press once. As long as oscilloscope data are acquired, this entry displays **Stop** as instruction.

### **3.4.2 DISPLAY [ SCOPE / DISPLAY ]**

The display screen of the oscillo**scope** works like the display of the **Spectra** mode. All measured data are superimposed on each other.

The scale can be fixed or **auto**. If **auto** is selected, the screen is divided by 4 parts and channel **1..4** will be scaled to its full area (from top to bottom). The user can control the **Color** of each line. The lines can be shifted by changing the parameter **Offset**. The information area shows the displayed channels, their mean values and standard deviations.

### **Channel [ Scope / Display / Channel ]**

The meaning of each channel, displayed as value under the channel number, is selected in **LockIn** mode. Here, the channels can only be selected. In addition, the time axis appears as channel.

If a channel is selected that is not the time or freq axis, then its display parameters are shown in the menu items **ScaleV** , **Color** , and **Offset**.

If **Channel** = **time** or **Channel** = **freq**, then the menu entries **x-Axis** and **ScaleH** appear.

*NOTE: the frequency axis is available for eLockIn software versions larger than 1.15f, only.*

## [x-Axis \[ Scope / Display / Channel = time/freq \]](#)

This menu entry allows to switch between "display vs time" and "display vs. frequency"

## [ScaleH \[ Scope / Display / Channel = time/freq & ScaleH\]](#)

If the x-Axis is set to **time**, the user can select the time per displayed division. The available numbers vary from 20  $\mu$ s/div to 5s/div. Horizontally, the oscilloscope screen is divided into 10 divisions.

When 100 ms/div is selected, the total time period shown on the whole screen equals 1 s (=10 \*100 ms).

If the x-Axis is set to **freq**, the user can select the frequency window displayed per each division. The left most frequency always equals zero.

If "10 Hz" (meaning 10 Hz/division) is selected, the total screen shows frequencies from 0 Hz to 100 Hz. The vertical line are found at 10 Hz, 20 Hz, ..., 90 Hz.

Values for available frequency/division setting range from 0.2 Hz to 2 kHz.

In contrast to the **Spectra** mode of the eLockIn, the data displayed vs. frequency in this **Scope** mode are calculate by a FFT transformation of the data acquired in a time period related time/division setting. Thus, when switching between **time** and **freq**, the related **ScaleH** value always is calculated by inverting the last selected **ScaleH** value.

When in ScaleH=time 1ms was chosen, ScaleH=freq will result in 1kHz.

## [ScaleV \[ Scope / Display / ScaleV \]](#)

Horizontally, all channels use the same time or frequency axis. The vertical scale **ScaleV** usually (and best) is set to **auto** for all channels on vertical axis (Ch1 to Ch4 from the lockin menu).

If the vertical scale is set to **ScaleV=auto**, then the channel is displayed with its min/max data to almost the 4<sup>th</sup> part of the vertical screen, which equals a bit more than two of the vertical divisions. With **[Scope/Display/Offset]** one realizes that the different channels get separated from each other.

Each channel can also be scaled separately as if it would be shown on a dc coupled oscilloscope. The available values for the vertical scaling range (for instance) from **1 $\mu$ V** to **50V**. In the list of available settings for **ScaleV**, **auto** is the last entry.

## [Color \[ Scope / Display / Color \]](#)

= **Color** of the selected channel. If **Off** is selected, the channel is not visible.

## [Offset \[ Scope / Display / Offset \]](#)

This **Offset** settings allows to vertically shift the display of the different channels against each other. If the horizontal axis is set to **x-Axis=time**, zero vertical offset means that the data are drawn on the middle of the vertical axis. If the horizontal axis is set to **x-Axis=freq**, zero offset data are drawn on the bottom most line of the screen.

### 3.4.3 ACQUIRE [SCOPE / ACQUIRE]

#### Points [Scope / Acquire / Points]

this entry allows to set the number of data *points* in the menu oscilloscope.

### 3.4.4 SAVE [SCOPE / SAVE]

One can save the oscilloscope data on the internal hard drive (HD) or a USB stick. As soon as they are saved on the HD, they also are saved into the WEB-space in a local directory /home/www/, which is accessible through the web-browser by typing:  
IP-address/data/osc.dat.

## 3.5 MEASURE [MEASURE]

*Measure* is intended to simplify procedures. Up to now, this function is used in spectrum display mode [Spectra] and when any phase is selected [LockIn/Ref/Phase].

In [Spectra],

*Measure* adds a cursor to the selected curve. You can shift the cursor along the curve with the *Knob*. Notice that the reference frequency shifts together with the cursor position. When you go back to [LockIn], the selected frequency is kept and you can change time constant or excitation amplitude.

In addition, the Q-factor of detected resonances is calculated and displayed.

You have measured a resonance curve in spectra mode. At the end, the reference frequency sets to the default value automatically. Now switch to Measure and shift the cursor to the position you like to use for the excitation. This changes the default reference frequency.

In [LockIn/Ref/Phase],

one click on the knob *Measure* starts an automated phase offset correction.

**Note: The cursor appears on the last selected data set out of the four display channels. No cursor will appear if another channel is selected in [ Spectra / Display ] or the signal is switched to invisible (*Color = Off*).**

## 3.6 SETUP [SETUP]

Run Time: 000 00:14:14	IP
Free RAM: 1962.7[MB]	192.16
Free Space: 205.6[MB] (99%)	DHCP
eLockin 203 HW:211 SW:1.15d	Off
MAC: BC:5F:F4:C7:C0:24	
static IP=192.168.2.7	Save
	RS232
	Off

The main screen in the *Setup* mode displays system parameters like

- run time of the instrument
- free RAM space
- free disk space
- software and hardware version
- MAC address
- IP address

The menu entries (Remote, Misc, Green, HD, Update) allow to set system parameters:

- **Remote:** The **IP** address, which is necessary to control the instrument over the LAN, can be changed here. **DHCP** can be enabled. In order to make any change of the IP address over or if the DHCP client is enabled or disabled, one has to confirm the changes with **Save**.  
In case the instrument is equipped with USB to RS232 adapter during start-up, **RS232** communication can be enabled here as well.
- **Misc** – opens a sub-menu with the following entries:
  - ◆ **Beep (On / Off):** This beep can help to identify a range overflow of the input channel.
  - ◆ **TimeOut:** The time given here is related to an internal timer and connected to several time-dependent functions:
    - ☑ This time multiplied by 1000 gives the time, after which the instrument stores the parameters automatically.
    - ☑ This multiplied by 10 gives the screen update rate of the **LockIn** and the oscillo**Scope** screen.
    - ☑ This time multiplied by 10 gives the screen update rate during spectrum acquisition.
  - ◆ **RefPLL:** selects the reference level for the PLL input. If **High**, the input level is expected to be a LV-TTL signal, because the switching level for a LV-TTL signal is 1.5 V. CMOS-TTL works as well. If **Low**, the PLL input can be sinusoidal wave with 100 mVpp minimum amplitude, which oscillates around zero.
- **Green** – opens a sub-menu. The entries in this menu are not stored. After restart, the brightness will be set to its full intensity and the auxiliary channels will be on ON state. Entries:
  - ◆ **Aux (On / Off):** switches the power supply of the analogue input and output channels on or off.
  - ◆ **Display:** is the brightness of the screen. 15 equals maximum and 0 switches it off.
- **HD:** Enters a sub-menu in which the data files stored on the internal hard disk can be
  - copied to the USB flash drive by > **USB (Do)** or
  - deleted all by **Clear. (Do)**
- **Update (?):** The **eLockIn** software can be updated with this function. Store the update files on a USB flash drive in a directory called "update" (case sensitive) and plug it into the front panel USB connector. Click on the **Update** button to start the update. The main screen will display messages about the updated files and the update success. To activate the new files switch the device off, wait a second and switch on again.

### 3.7 HANDLING OF PARAMETER SETTINGS

The settings done on the device are stored in a file. However, this storage is only done, if the **eLockIn** hasn't got new input for more than 10 seconds (**TimeOut** = 10 ms, see 3.6). If you like to switch off the device and find it back in the same state after switching on again, just do not press any button for 10 seconds before you switch off. In case a parameter setting is not found back or the parameter file is damaged, **eLockIn** uses default settings.

## 3.8 FTP DATA ACCESS

**Note: The FTP server is not active in the current version of the eLockIn software.**

For instruments with an older software version:

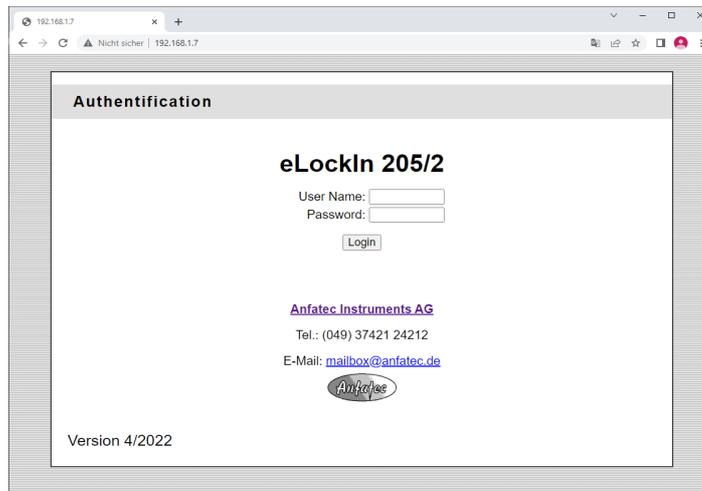
The easiest way to use ftp-access is by a Web browser. Type in the path: "ftp://root@[IP address]" (e.g. if the IP address of the device is set to "192.168.1.7" then type "ftp://root@192.168.1.7"). Stored data files are located on the hard disk at "ftp://root@[IP address]/d/data".

User name: root  
Default-Password: eLockIn (!! case sensitive !!)

**How to change the root password?** Create a file called "root.pwd" in a directory "update" on a USB flash drive. Inside this file, write the password as clear ASCII data. Plug the USB flash drive in the front slot and "update". The password is coded internally and stored. After the next restart, the **eLockIn** can be accessed with the new root-password.

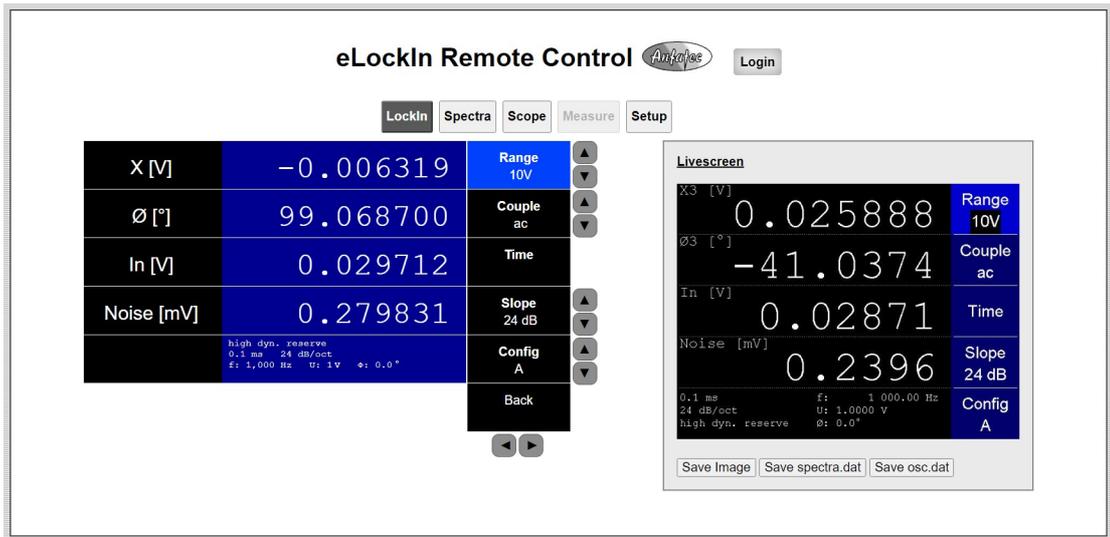
## 3.9 HTTP WEB ACCESS

Standard access via web browser uses "http://[IP address]". The "Log in" window appears:



User name: long  
Password: nga

After authenticate it is possible to control the eLockIn via the web-interface.



### 3.10 CHANGE LOGIN PASSWORD FOR WEB-INTERFACE

To change or add login data for the eLockIn web-interface the **user.lst** file can be modified. It is located on the eLockin update usb stick. Alternatively a user.lst can be newly created on an usb stick in a directory "update".

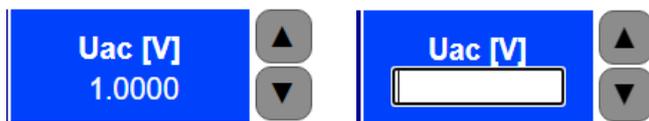
usb-drive:/update/user.lst

Each line contains user login data, consisting of name and password, which are separated by space:

```
user password
long nga
anotheruser anotherpassword
```

### 3.11 HOW THE WEB-INTERFACE WORKS

After successful login to the web-interface, the eLockIn can be controlled via the browser. By clicking on the menus or gray arrows, it is possible to navigate through the eLockIn menus. If a menu have up/down arrows it can be double-clicked to edit its value:



On input the edit field changes to yellow. The value will be set by pressing Enter. To cancel the input and remove the edit field ESC key on keyboard can be pressed.

If spectra is acquired the data file can be downloaded by clicking on "Save spectra.dat". To save the Oscilloscope data, first the file must be saved to web: Scope > Save > Dest. WEB. Only then is the data file available in the web space and can be saved by clicking on "Save osc.dat".

### 3.12 KNOWN PROBLEMS

If the browser does not update the data or live screen anymore, you may have to update the page with the key combination Cntl + F5 (Chrome/Firefox) or delete the cache of the browser.

If the channel names and the current settings are not displayed it is necessary to authenticate again.

→ "Login"

Login

1	-0.032051	Input
2	-137.304800	Ref
3	0.033237	Display
4	0.304398	Out
!		
		Back



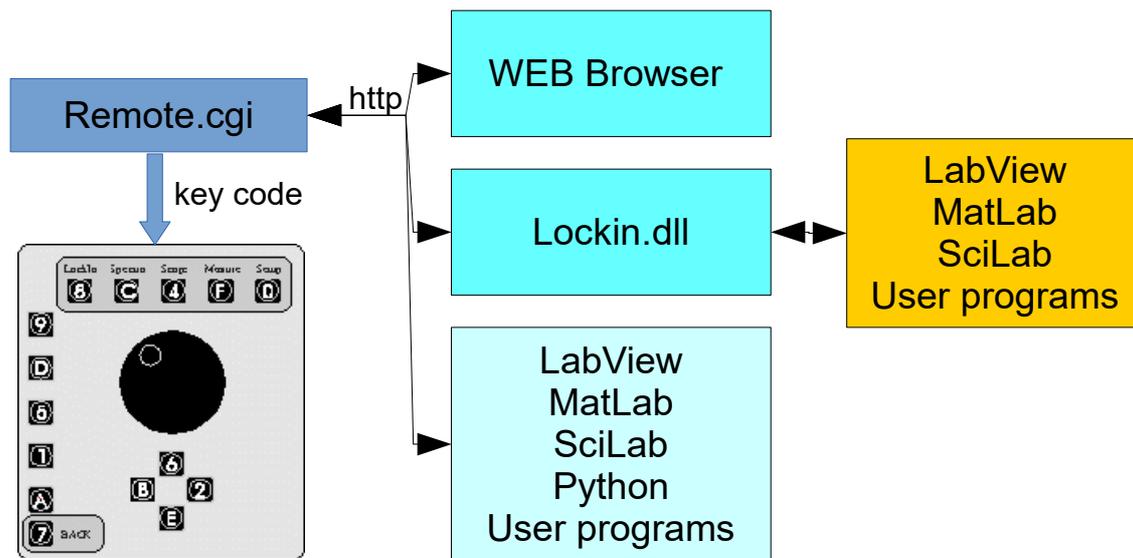
Some newer hardware versions react over-critical on the overload pin of the A/D-converters. Thus, it might be that the overload (red alarm sign) is shown even though the measurement is still ok.

# 4 REMOTE CONTROL

## 4.1 ACCESS THROUGH WEB INTERFACE

### 4.1.1 GENERAL

The communication between the remote computer and the eLockIn is based on TCP (Transmission Control Protocol) and http (Hypertext Transfer Protocol) for transferring HTML files and pictures. Commands are received with the remote.cgi program through an http connection:



Those http commands can be sent through

- a web-browser or
- by any http capable program (LabView, MatLab, Python, ...).

Alternatively, one can use the commands provided by the Lockin.dll library (see chapter 4.3) and the lockin-dll translates them into http commands.

The web server on the eLockIn provides CGI-programs and data files.

Available cgi files are:

- ***/cgi-bin/remote.cgi***, which accepts key codes and passes them to the ***eLockIn*** software by pretending that front panel key are pressed,
- ***/cgi-bin/refresh.cgi***, which sends an HTML page containing current channel values

Available data files are:

- ***/screen.jpg***, which is an image of the ***eLockIn***'s display (see )
- a textfile ***/data/lia.dat***, which contains the current values of the channels 1...4.
- a textfile ***/setup/lockin.ini***, which contains all settings of the instrument (see chapter 4.1.7)

### 4.1.1 TECHNICAL REQUIREMENTS

Both the **eLockIn** and the remote computer must be connected to a network. This could be a simple crosslink cable between the two devices, a LAN (Local Area Network) or even the internet.

On start-up, the **eLockIn** will set its IP address to a default value, which can be changed in the **Setup** menu. You have to know this IP address to connect to the **eLockIn**.

### 4.1.2 AUTHENTICATION

Authentication is required only if the user wants to set parameters on the eLockin.

Before the **eLockIn** can be remote-controlled, the user has to provide a user name and a password. This can either be done by entering the **eLockIn**'s IP address in a web browser and filling in the authorization form, or by calling **/cgi-bin/login.cgi** directly. login.cgi takes a string as argument: ?username=<user>&password=<pass> <user> denotes the user name and <pass> the corresponding password.

Thus, the following http request would authenticate user 'long' by sending his password 'nga':

```
http://192.168.1.7/cgi-bin/login.cgi?username=long
&password=nga
```

## 4.2 KEY CODES

A program named remote.cgi will be called by the **eLockIn**'s web server when it receives a request for **/cgi-bin/remote.cgi**.

Let's assume a TCP connection has already been established between the remote and the **eLockIn** on port 80 (default for http connections) and the user has identified himself as described in the 'Authentication' section 4.1.2.

If that is the case, commands can be sent to the **eLockIn** device by requesting **/cgi-bin/remote.cgi** via http and passing keys as arguments to it. Those keys are treated the same way as if they were pressed on the **eLockIn** itself. A typical http request would look like the following:

```
http://192.168.1.7/cgi-bin/remote.cgi?S_6_8
```

The request shown above would cause the **eLockIn** to

- dump a screenshot of the display to /screen.jpg ('S'),
- enter the number '6' at the position that is currently focused on the display ('\_6\_', underscores enclose numbers),
- switch to the **eLockIn** menu ('8').

For a complete list of the commands that can be sent to remote.cgi, see Table 1 or

Figure 12. Number to be set in certain sub-menus of the eLockIn have to be placed with two underscores, e.g. "\_1063.7\_"

All of these key codes (except 'S') correspond to buttons on the front panel of the **eLockIn** (see Figure 12). Therefore the **eLockIn** can be remote-controlled in the same way as if it is operated manually.

<i>Key (char) with remote.cgi</i>	<i>Key (char) with remote2.cgi</i>	<i>ASCII Code</i>	<i>Action</i>
8	A	56	Enter LockIn menu
C	B	67	Enter Spectra menu
4	C	52	Enter Scope menu
F	D	70	Enter Measure menu
0	E	48	Enter Setup menu
9	1	57	These buttons invoke the action displayed next to them on the screen, depending on the current position in the menu tree.
D	2	68	
5	3	53	
1	4	49	
A	5	65	
7	6	55	Go back in menu
+	+	43	Turn Knob Clockwise
-	-	45	Turn Knob Counter Clockwise
S	S	83	Save screenshot

Table 1: Command keys

Specific values can be adjusted either by using buttons (up, down) or by sending numbers to the **eLockIn**. The device accepts numbers wherever the software expects user input. Just like the other commands, numbers need to be passed to '/cgi-bin/remote.cgi', but have to be enclosed by two '\_' (underscore).

[http://192.168.1.7/cgi-bin/remote2.cgi?A22\\_100000\\_8](http://192.168.1.7/cgi-bin/remote2.cgi?A22_100000_8)

The command sequence shown above tells the **eLockIn** to

- A ... enter 'LockIn' menu
- 2 ... enter 'Ref' submenu
- 2 ... select the line in which the frequency can be changed
- \_100000\_ set the frequency to 100000 Hz (which is 100 kHz)  
Note: the second "\_" after the number is required even if there is no further menu entry.
- 8 ... return to top of the 'LockIn' menu

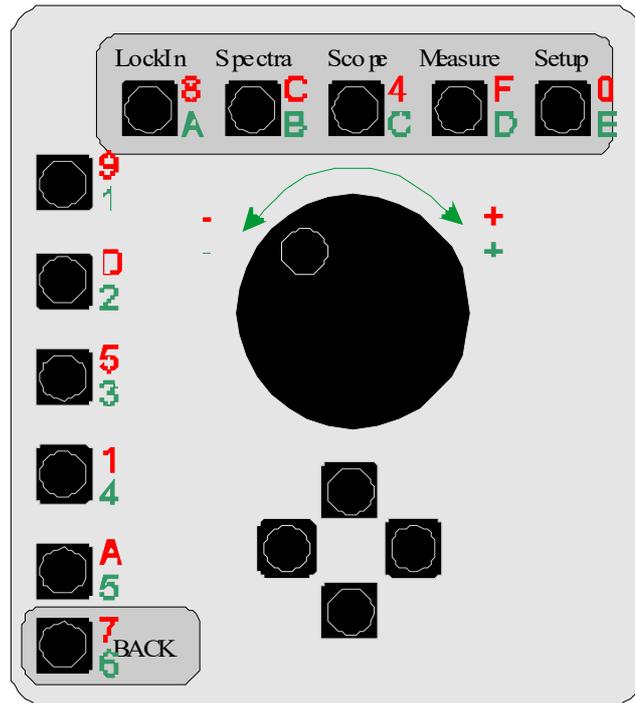


Figure 12: Front Panel knobs and characters/numbers used to address the front panel knobs by Remote Control access with *remote.cgi* and *remote2.cgi*

### 4.1.3 HOW TO SET ANY VALUE IN ANY SUB-MENU

In some sub-menus of the eLockIn series, values are set as numbers (e.g.: frequency is set as number in Hz). In this case, the user needs to set the number itself. In other sub-menus, a limited amount of dedicated values is available, only. The Roll-off, for instance, can be 6 dB/oct, 12 dB/oct or 24 dB/oct -- but never 1 dB/oct. In this case, the user sets the count of the value, which is "0" for 6 dB, "1" for 12 dB and "2" for 24 dB.

The following tables show the counts of the available values (which equals the "number" that one has to send to the eLockIn) and the related entries ("values") shown in the eLockIn's menu tree :

## Set Input range

<b>Number XX</b>	<b>Value</b>
1	"low noise"
10	"normal"
100	"high dyn. reserve"

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?899\\_XX\\_](http://192.168.1.7/cgi-bin/remote.cgi?899_XX_)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A11\\_XX\\_](http://192.168.1.7/cgi-bin/remote2.cgi?A11_XX_)

Menu-Entry: **LockIn-Input-Range**

lockin.ini entry: [InputRange](#)

## Set Input Coupling Type

<b>Number XX</b>	<b>Value</b>
0	dc coupled
1	ac coupled
2	dc 50 Ohm
3	ac 50 Ohm

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?89D\\_XX\\_](http://192.168.1.7/cgi-bin/remote.cgi?89D_XX_)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A12\\_XX\\_](http://192.168.1.7/cgi-bin/remote2.cgi?A12_XX_)

Menu-Entry: **LockIn-Input-Couple**

lockin.ini entry: [InputCouple](#)

## Set Input Time Constant

<b>Number</b>	<b>Value</b>
-9	0.1 $\mu$ s
...	...
-1	50 $\mu$ s
0	0.1 ms
1	0.2 ms
2	0.5 ms
3	1 ms
4	2 ms
5	5 ms
6	10 ms
7	20 ms
8	50 ms
9	100 ms
10	200 ms
11	500 ms
12	1 s
13	2 s
14	5 s
15	10s
16	20s
17	50s
18	100s
19	200s
20	500s
21	1ks
100	1/f
101	2/f
102	5/f
103	10/f
104	20/f
105	50/f
106	100/f

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?8959\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?8959_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A131\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?A131_XX)

for 1<sup>st</sup> Lockin Amplifier.

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?8955\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?8955_XX)

for 2<sup>nd</sup> Lockin Amplifier.

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?8955\\_-10](http://192.168.1.7/cgi-bin/remote.cgi?8955_-10)  
 Sets the time constant of 2<sup>nd</sup> LIA equal to 1<sup>st</sup> LIA.

Menu-Entry: **LockIn-Input-Time-Tau** and **LockIn-Input-Time-Tau1**  
 lockin.ini entry: `Timeconstant`

### Set Sync Filter Settings

<b>Number XX</b>	<b>Value</b>
0	"Off"
1	1/f
2	2/f
5	5/f
⋮	⋮
200	200/f

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?895D\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?895D_XX)  
 HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A132\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?A132_XX)  
 Menu-Entry: **LockIn-Input-Time-Sync**  
 lockin.ini entry: `Sync0` and `SyncLoL`  
 These two entries in the lockin.ini file equal each other.

### Set Roll-Off

<b>Number XX</b>	<b>Value</b>
6	6dB/oct
12	12dB/oct
24	24dB/oct

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?891\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?891_XX)  
 HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A14\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?A14_XX)  
 Menu-Entry: **LockIn-Input-Slope**  
 lockin.ini entries: `Rolloff:` and `RolloffLoL:`  
 These two entries in the lockin.ini file equal each other.

## Set Input Configuration

<b>Number XX</b>	<b>Value</b>
0	"A"
1	"A-B"
2	"A&B"

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?89A\\_XX\\_](http://192.168.1.7/cgi-bin/remote.cgi?89A_XX_)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A15\\_XX\\_](http://192.168.1.7/cgi-bin/remote2.cgi?A15_XX_)

Menu-Entry: **LockIn-Input-Config**

lockin.ini entry: `InputMode:`

## Set Reference Output Amplitude

<b>Number XX</b>	<b>Value</b>
Any floating point number with a "." as decimal delimiter	Automatically limited to 0 V ... 8 V

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?8D9\\_XX\\_](http://192.168.1.7/cgi-bin/remote.cgi?8D9_XX_)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A21\\_XX\\_](http://192.168.1.7/cgi-bin/remote2.cgi?A21_XX_)

This call sets the amplitude to 130 mV:

[http://192.168.1.7/cgi-bin/remote.cgi?8D9\\_0.13\\_](http://192.168.1.7/cgi-bin/remote.cgi?8D9_0.13_)

Menu-Entry: **LockIn-Ref-Uac[V]**

lockin.ini entry: `Amplitude:` in Volts

## Set Reference Output Frequency

<b>Number XX</b>	<b>Value</b>
Any floating point number with a "." as decimal delimiter	Automatically limited

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?8DD\\_XX\\_](http://192.168.1.7/cgi-bin/remote.cgi?8DD_XX_)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A22\\_XX\\_](http://192.168.1.7/cgi-bin/remote2.cgi?A22_XX_)

This call sets the frequency to 100230.5 Hz:

[http://192.168.1.7/cgi-bin/remote.cgi?8DD\\_100230.5\\_](http://192.168.1.7/cgi-bin/remote.cgi?8DD_100230.5_)

[http://192.168.1.7/cgi-bin/remote2.cgi?A22\\_100230.5\\_](http://192.168.1.7/cgi-bin/remote2.cgi?A22_100230.5_)

Menu-Entry: **LockIn-Ref-f[Hz]**

lockin.ini entry: `Frequency:` in Hz

## Set Offset Phase for Lia1 and Lia2

<b>Number XX</b>	<b>Value</b>
Any floating point number with a "." as decimal delimiter	Automatically limited to values between -180 deg and 180 deg.

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?8D59\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?8D59_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?A231\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?A231_XX)

for the phase offset of LIA1

Menu-Entry: **LockIn-Ref-Phase-Phase**

lockin.ini entry: Phase:

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?8D5D\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?8D5D_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?A232\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?A232_XX)

for the phase offset of LIA2

Menu-Entry: **LockIn-Ref-Phase-Phase1**

lockin.ini entry: Phase0:

## Set Harmonic

<b>Number XX</b>	<b>Value</b>
Any integer number	Resulting values range between 1 and 15

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?8D1\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?8D1_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A24\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?A24_XX)

Menu-Entry: **LockIn-Ref-Harm**

lockin.ini entry: Harmonic:

## Set Reference Input Coupling

<b>Number XX</b>	<b>Value</b>
0	RefIn = OFF
1	RefIn = ON
2	LoL active (available only, if LoL mode is implemented)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?8DA\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?8DA_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?A25\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?A25_XX)

Menu-Entry: **LockIn-Ref-RefIn**

lockin.ini entry: InputCouple:

## Select Display Channels in the Lockin-Menu

The idea of this function is to select which signal is shown in which of the four rows of the lockin-display screen. At the front panel of the eLockIn, the user would first select a channel 1..4 (Lockin/Display/Channel) and afterwards select the group or the signal that should be shown in the related row. The idea of the entry "Group" is that one gets faster to the desired signal, because switching between groups causes a display of the 1<sup>st</sup> signal within this group.

In remote control, (LockIn/Display/Group) is not used.

<b>Number XX</b>	<b>Value for A or A-B harmonic = n</b>	<b>Value for A&amp;B</b>	<b>Group</b>
1	X	X_A	LIA n
2	Y	Y_A	LIA n
3	R	R_A	LIA n
4	Phi	Phi_A	LIA n
5	Uac	Uac	Misc
6	f	f	Misc
7	In	In	Misc
8	AD1	AD1	ADC
...	...	...	ADC
15	AD8	AD8	ADC
16	DA1	DA1	DAC
...	...	...	...
23	DA8	DA8	DAC
24	Xn	X_B	LIA1
25	Yn	Y_B	LIA1
26	Rn	R_B	LIA1
27	Phin	Phi_B	LIA1
28	Noise	Noise	Math

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?859\\_0\\_5\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?859_0_5_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A31\\_0\\_3\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?A31_0_3_XX)

for the 1<sup>st</sup> row in the lockin display

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?859\\_1\\_5\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?859_1_5_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A31\\_1\\_3\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?A31_1_3_XX)

for the 2<sup>nd</sup> row in the lockin display

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?859\\_2\\_5\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?859_2_5_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A31\\_2\\_3\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?A31_2_3_XX)

for the 3<sup>rd</sup> row in the lockin display

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?859\\_3\\_5\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?859_3_5_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A31\\_3\\_5\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?A31_3_5_XX)

for the 4<sup>th</sup> row in the lockin display

Menu-Entry: **LockIn-Display-Channel & LockIn-Display-Type**

lockin.ini entry: **ChannelXType**: signal for the Xth channel

## Select Display Range in the Lockin-Menu

When a channel gets a new signal, the range of this channel automatically is set to a default value (e.g. [V] for R, X and Y). In order to visualize the value better or if the analogue outputs at the backside of the eLockIn will be used, one might like to show the signals in a different scaling (e.g. 1 mV).

<b>Number XX</b>	<b>Range</b>	<b>Available for channels of type</b>
< = 79	1 fV	
80	2 fV	
82 ..	10 fV	X_A
85 ..	100 fV ..	Y_A
88 ..	1 pV ..	R_A
91 ..	10 pV ..	X_B
94 ..	100 pV ..	Y_B
97 ..	1 nV ..	R_B
100 ..	10 nV ..	or
103 ..	100 nV ..	X
106 ..	1 µV ..	Y
109 ..	10 µV ..	R
112 ..	100 µV ..	Xn
115 ..	1 mV ..	Yn
118 ..	10 mV ..	Rn
121 ..	100 mV ..	AND
124 ..	1 V	Noise
> =125	10 V	
<= 3	10 mV	AD1 to AD8
4	100 mV	In
5	1 V	
>= 6	10 V	
any	10 V	DA1 to DA8
any	20 MHz	f
Any	180°	Phi_A, Phi_B or Phi, Phin

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?859\\_0\\_1\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?859_0_1_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A31\\_0\\_4\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?A31_0_4_XX)

for the 1<sup>st</sup> row in the lockin display

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?859\\_1\\_1\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?859_1_1_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A31\\_1\\_4\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?A31_1_4_XX)

for the 2<sup>nd</sup> row in the lockin display

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?859\\_2\\_1\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?859_2_1_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A31\\_2\\_4\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?A31_2_4_XX)

for the 3<sup>rd</sup> row in the lockin display

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?859\\_3\\_1\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?859_3_1_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A31\\_3\\_4\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?A31_3_4_XX)

for the 4<sup>th</sup> row in the lockin display

Menu-Entry: **LockIn-Display-Channel & LockIn-Display-Range**

lockin.ini entry: **ChannelXRange:** scaling of the Xth channel

### Set output voltages at analogue outputs DA5 to DA8

<b>Number XX</b>	<b>Value</b>
Any floating point number with a "." as decimal delimiter	Automatically limited to -10 V ... 10 V

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?819\\_4\\_5\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?819_4_5_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A41\\_4\\_3\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?A41_4_3_XX)

for DAC No 5 available at analogue output "Display 1"

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?819\\_5\\_5\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?819_5_5_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A41\\_5\\_3\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?A41_5_3_XX)

for DAC No 6 available at analogue output "Display 2"

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?819\\_6\\_5\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?819_6_5_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A41\\_6\\_3\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?A41_6_3_XX)

for DAC No 7 available at analogue output "Display 3"

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?819\\_7\\_5\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?819_7_5_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?A41\\_7\\_3\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?A41_7_3_XX)

for DAC No 8 available at analogue output "Display 4"

This call sets the value at Display output 2 to 35 mV:

[http://192.168.1.7/cgi-bin/remote.cgi?819\\_5\\_5\\_0.035](http://192.168.1.7/cgi-bin/remote.cgi?819_5_5_0.035)

[http://192.168.1.7/cgi-bin/remote2.cgi?A41\\_5\\_3\\_0.035](http://192.168.1.7/cgi-bin/remote2.cgi?A41_5_3_0.035)

Menu-Entry: **LockIn-Out-Channel & LockIn-Out-Value**

lockin.ini entry: none

### Start and Stop Spectrum Acquisition

This call toggles between Run and Stop of Spectrum acquisition:

HTTP-Call: <http://192.168.1.7/cgi-bin/remote.cgi?C9>

HTTP-Call: <http://192.168.1.7/cgi-bin/remote.cgi?B1>

Menu-Entry: **Spectra-Run**

## Get Acquired Spectrum

While a spectrum is acquired, the currently available data are continuously written into an ASCII-file. The user can get this file by calling:

HTTP-Call: <http://192.168.1.7/data/spectra.dat>

In case the spectrum is acquired logarithmically, the *spectra.dat* has the following structure:

```
10000.000000 313952.109451 1.005000 2
9999.998146682532 0.000121845705 44.127700000000 0.000091704069 9999.998146682532
10076.366002144974 0.000088171444 -1.451100000000 0.000082697557 10076.366002144974
10153.329899406146 0.000271010270 -127.491200000000 0.000118058745 10153.329899406146
10230.871212159840 0.000159494045 -155.797400000000 0.000048751126 10230.871212159840
10309.008566712266 0.000123891301 117.353000000000 0.000078561520 10309.008566712266
10387.751276216526 0.000079301988 159.333000000000 0.000050003534 10387.751276216526
10467.080714366415 0.000094011543 -170.926400000000 0.000060973697 10467.080714366415
10547.024820621244 0.000037265766 130.829100000000 0.000140710315 10547.024820621244
```

The first line "10000.000000 313952.109451 1.005000 2" tells that the available data are taken between 10000 Hz and 313.952 kHz. From frequency value to frequency value, the next value is calculated by multiplying with a factor of 1.005000. The "2" at the end of the line stands for logarithmic spectrum acquisition.

In case the spectrum is acquired linearly, the *spectra.dat* has the following structure:

```
10000.000000 10935465.456431 560.000000 1
9999.998146682532 0.000058221416 33.852000000000 0.000129871577 9999.998146682532
30010.006079635659 0.000275890032 -1.688100000000 0.000332576069 30010.006079635659
50020.014012588785 0.000079356642 147.287100000000 0.000084112314 50020.014012588785
70030.021945541914 0.000071338217 -120.204500000000 0.000120957837 70030.021945541914
90040.029878495043 0.000039686128 7.903700000000 0.000013312629 90040.029878495043
110050.047124601275 0.000016372577 -39.679600000000 0.000055554328 110050.047124601275
130060.055057554404 0.000010797937 -104.568600000000 0.000069756012 130060.055057554404
150070.062990507518 0.000051631786 159.696400000000 0.000034116511 150070.062990507518
170080.070923460647 0.000031628665 36.428600000000 0.000027792626 170080.070923460647
190090.088169566880 0.000017286069 113.276700000000 0.000018074870 190090.088169566880
210100.096102520009 0.000039342593 104.797300000000 0.000030390212 210100.096102520009
230110.104035473138 0.000032370389 -31.236300000000 0.000006540351 230110.104035473138
250120.111968426267 0.000002053404 -16.512000000000 0.000015392244 250120.111968426267
270130.119901379396 0.000008369768 -178.343500000000 0.000035894621 270130.119901379396
290140.137147485628 0.000001366333 67.956500000000 0.000023223657 290140.137147485628
```

The first line "10000.000000 10935465.456431 560.000000 1" tells that the available data are taken between 10 kHz and 1.0935 MHz. From frequency value to frequency value, the next value is calculated by adding 560.0 Hz. The "1" at the end of the line stands for linear spectrum acquisition.

Each of the following lines contains the Frequency in Hz and the chosen channels (in the LockIn menu) in their order and physical base unit. Thus, any voltages are saved in [V], phase signals are saved in ° and frequencies in Hz. The Noise channel is saved in [V/sqrt(Hz)].

## Select Axis for Spectrum Display

<b>Number XX</b>	<b>Value</b>
0	"Freq"
1..4	Ch1 .. Ch4
5	Ref

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?CD9\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?CD9_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?B21\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?B21_XX)

Note: The vertical scaling on the eLockIn screen is displayed for the last selected axis, only.

## Switch between linear and logarithmic Spectrum Acquisition

This feature is provided as Type selection for the frequency axis in ***Spectra/Display***.

This call sets the frequency axis to linear spectrum acquisition:

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?CD9\\_0\\_D\\_0](http://192.168.1.7/cgi-bin/remote.cgi?CD9_0_D_0)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?B21\\_E\\_2\\_0](http://192.168.1.7/cgi-bin/remote2.cgi?B21_E_2_0)

This call sets the frequency axis to logarithmic spectrum acquisition:

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?CD9\\_0\\_D\\_1](http://192.168.1.7/cgi-bin/remote.cgi?CD9_0_D_1)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?B21\\_E\\_2\\_1](http://192.168.1.7/cgi-bin/remote2.cgi?B21_E_2_1)

Menu-Entry: ***Spectra-Display-Axis & Spectra-Display-Type***

## Change the Color of displayed Spectra Data

<b>Number XX</b>	<b>Value</b>
1	White
2	Yellow
3	Blue
4	Green
5	Red
6	Cyan
7	Magenta
8	Gray
9	Brown
10-15	White

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?CD9\\_1\\_5\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?CD9_1_5_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?B21\\_1\\_3\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?B21_1_3_XX)

Changes color for the 1<sup>st</sup> channel

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?CD9\\_4\\_5\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?CD9_4_5_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?B21\\_4\\_3\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?B21_4_3_XX)

Changes color for the 4<sup>th</sup> channel

## Display Spectra with a vertical Offset to each other

<b>Number XX</b>	<b>Value</b>
Any integer number	Automatically limited to -240 ..240

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?CD9\\_1\\_1\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?CD9_1_1_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?B21\\_1\\_4\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?B21_1_4_XX)

Changes the Offset of the 1<sup>st</sup> channel

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?CD9\\_3\\_1\\_XX](http://192.168.1.7/cgi-bin/remote.cgi?CD9_3_1_XX)

HTTP-Call: [http://192.168.1.7/cgi-bin/remote2.cgi?B21\\_3\\_4\\_XX](http://192.168.1.7/cgi-bin/remote2.cgi?B21_3_4_XX)

Changes the Offset of the 4<sup>th</sup> channel

## Set Parameters for Spectrum Acquisition

The exact way, how a spectrum is acquired, is defined with parameters in **Spectra-Acquire**

All these parameters can be set with a single HTTP-call:

<http://192.168.1.7/cgi-bin/remote.cgi?>

[B31\\_TT\\_F\\_F1\\_5\\_F2\\_1\\_PP\\_A\\_DD\\_](http://192.168.1.7/cgi-bin/remote.cgi?B31_TT_F_F1_5_F2_1_PP_A_DD_)

<http://192.168.1.7/cgi-bin/remote2.cgi?>

[C59\\_TT\\_D\\_F1\\_3\\_F2\\_4\\_PP\\_5\\_DD\\_](http://192.168.1.7/cgi-bin/remote2.cgi?C59_TT_D_F1_3_F2_4_PP_5_DD_)

The values **TT**, **F1**, **F2**, **PP**, and **DD** define the spectrum are are set according to the following tables:

Table for the type of integration/acquisition:

<b>Number TT</b>	<b>Value</b>
0	Single
1	Conti.
2	Integ.

Table for the amount of data points:

<b>Number PP</b>	<b>Value</b>
10	10
20	20
50	50
... continues logarithmically	...
100000	100000

Table for the delay time per data point:

<b>Number DD</b>	<b>Value</b>
0	0.01*T
1	0.1*T
2	0.2*T
3	0.5*T
4	1*T
5	2*T
6	5*T

The start and the stop frequency F1 and F2 are set like any other real number:

<b>Number F1 or F2</b>	<b>Value</b>
Any floating point number with a "." as decimal delimiter	Automatically limited to the available frequency range of the lockin amplifier

## Set Display Settings for Oscilloscope

The exact way, how a spectrum is acquired, is defined with parameters in

### **Scope-Display**

All these parameters can be set with a single HTTP-call:

<http://192.168.1.7/cgi-bin/remote.cgi?>

**4D9\_CH\_D\_Scale\_5\_Col\_1\_Offset**

The value **CH** defines the channel to be changed with "0" for the time channels and "1" to "4" for the four available display channels. The value **Col** defines the display color for the oscilloscope screen.

**Scale** defines the scaling of the selected axis. It is defined differently for the time axis (**ScaleH**) and the four vertically displayed channels. For all data given in Volts, the scaling is according to the following table:

<b>Number ScaleV for voltage type channels</b>	<b>Value</b>
0	1 $\mu$ V
1	2 $\mu$ V
2	5 $\mu$ V
3	10 $\mu$ V
4	20 $\mu$ V
5	50 $\mu$ V
6	100 $\mu$ V
7	200 $\mu$ V
8	500 $\mu$ V
9	1mV
10	2mV
11	5mV
12	10mV
13	20mV
14	50mV
15	100mV
16	200mV
17	500mV
18	1V
19	2V
20	5V
21	10V
22	20V
23	50V
24	auto

If the signal available in any channel is a phase or the frequency, the scaling of the oscilloscope display cannot be changed for this channel.

If the Channel **Time** is chosen, only the first two parameters should be set:

[http://192.168.1.7/cgi-bin/remote.cgi?4D9\\_0\\_D\\_0\\_5\\_ScaleH](http://192.168.1.7/cgi-bin/remote.cgi?4D9_0_D_0_5_ScaleH)

On the oscilloscope screen, **ScaleV** (vertical scaling) is renamed into **ScaleH** (horizontal scaling) and is the given by:

<b><i>Number ScaleH for the Time channel</i></b>	<b><i>Value</i></b>
0	1µs
1	2µs
2	5µs
3	10µs
4	20µs
5	50µs
6	100µs
7	200µs
8	500µs
9	1ms
10	2ms
11	5ms
12	10ms
13	20ms
14	50ms
15	100ms
16	200ms
17	500ms
18	1s
19	2s
20	5s

The color value **Col** is set by:

<b>Number Col</b>	<b>Value</b>
1	White
2	Yellow
3	Blue
4	Green
5	Red
6	Cyan
7	Magenta
8	Gray
9	Brown
10-15	White

If the Channel **freq** is chosen, only the first two parameters should be set:

[http://192.168.1.7/cgi-bin/remote.cgi?4D9\\_0\\_D\\_1\\_5\\_ScaleH](http://192.168.1.7/cgi-bin/remote.cgi?4D9_0_D_1_5_ScaleH)

The value of **ScaleH** (horizontal scaling) and is the given by:

<b>Number ScaleH for the Time channel</b>	<b>Value</b>
>= 4	50kHz
5	20kHz
6	10kHz
7	5kHz
8	2kHz
9	1kHz
10	500Hz
11	200Hz
12	100Hz
13	50Hz
14	20Hz
15	10Hz
16	5Hz
17	2Hz
18	1Hz
19	0.5Hz
>= 20	0.2Hz

The example:

HTTP-Call: [http://192.168.1.7/cgi-bin/remote.cgi?4D9\\_Ch\\_5\\_Col](http://192.168.1.7/cgi-bin/remote.cgi?4D9_Ch_5_Col)

changes color for the Ch<sup>th</sup> data channel.

The Offset makes it possible not to show all data on top of each other, but to move the vertically. The number **Offset** is simply an integer between -4 and 4.

#### 4.1.4 EXAMPLES FOR KEYCODE OPERATION

The following examples are not complete at all, however, they explain the basic operation of the device through the KeyCode. Thus, even never described or new entries in the menu can be accessed by simply following a very simple schemata:

The lowest entry usually is decoded as "\_0\_" and following entries simply count up from this lowest entry.

KeyCodes cannot only be used from the remote.cgi (as shown here), but also by the DLL.

##### Set Frequency

Code:

[http://192.168.1.7/cgi-bin/remote.cgi?8DD\\_1040\\_](http://192.168.1.7/cgi-bin/remote.cgi?8DD_1040_)

Description:

8 ... enter Lockin menu

D ... enter Ref entry

D ... select Freq entry inside the Ref menu

\_ ... tells that a parameter is following

1040 ... is the new frequency value in Hz = 1.04 kHz

\_ finalizes the data transfer (never forget this final underscore)

##### Set Reference Output Amplitude

Code:

[http://192.168.1.7/cgi-bin/remote.cgi?8D9\\_0.15\\_](http://192.168.1.7/cgi-bin/remote.cgi?8D9_0.15_)

Description:

8 ... enter Lockin menu

D ... enter Ref entry

9 ... Select Uac entry inside the Ref menu

\_ ... tells that a parameter is following

0.15 ... is the new reference output amplitude value in V (= 150 mV)

\_ finalizes the data transfer (never forget this final underscore)

##### Set PLL Input to Low

Code:

[http://192.168.1.7/cgi-bin/remote.cgi?0D5\\_0\\_](http://192.168.1.7/cgi-bin/remote.cgi?0D5_0_)

Function: toggles the type of reference input for the PLL mode

Description:

0 ... enter Setup menu

D ... enter Misc entry

D ... select PLL entry inside the Misc menu  
\_ ... tells that a parameter is following  
0 ... is the lowest entry in this menu and equals "PLL/Low"  
\_ finalizes the data transfer (never forget this final underscore)

## Set Oscilloscope to Start

Code:

```
http://192.168.1.7/cgi-bin/remote.cgi?49_0_
```

Function: stops the oscilloscope screen from data acquisition.

Description:

4 ... enter Osci menu  
9 ... select Run entry inside the Misc menu  
\_ ... tells that a parameter is following  
1 ... equals the 2<sup>nd</sup> entry in this menu which is "STOP"  
\_ finalizes the data transfer (never forget this final underscore)

## How to set multiple parameters at once

After finishes one entry with "\_", one can simply add further entries from the current menu situation, which means that it is not required always to return to the top-most menu "LockIn" or "Spectra".

For instance, when keycode "859\_1\_0\_" was used to enable the entry Channel inside [Lockin/Display], one can simply continue to enter values in the Scale entry (key code "1") by adding "1\_124\_". The total key code:

```
http://192.168.1.7/cgi-bin/remote.cgi?859_0_1_124_
```

simultaneously makes sure that channel 1 is selected and its value is set to 1 V display range. The only limit is the length of the complete string which is about 128 characters.

Example Code

```
http://192.168.1.7/cgi-bin/remote.cgi?  
859_0_1_124_9_1_1_124_9_2_1_124_9_3_1_124_
```

Function: sets the display ranges of all 4 displayed channels to 1 V. The code shown with dark yellow background (9\_0\_1\_124\_) simply repeats by changing the channel number, only. 9\_2\_1\_124\_, for example, sets the display range for channel 3.

### 4.1.5 RETRIEVING SCREENSHOTS

The screen shot of the eLockIn display screen is saved with the command

```
http://192.168.1.7/cgi-bin/remote.cgi?S
```

After receiving the key code 'S', the **eLockIn** will save its current display content as a JPEG image in '/screen.jpg'. Then it can be retrieved by a web browser calling:

```
http://192.168.1.7/screen.jpg
```

or any program such as the Delphi example shipped with the **eLockIn**

### 4.1.6 RETRIEVING CHANNEL VALUES

If you are interested in obtaining actual channel values as ASCII file, you can either use the HTML file sent by /cgi-bin/refresh.cgi or the text file /data/lia.dat. To extract pure data from refresh.cgi's answer, surrounding HTML tags have to be cut away. On the other hand, lia.dat provides pure numbers and is faster.

### 4.1.7 GET INTERNAL SETTINGS AND STATUS INFORMATION

The current settings of the **eLockIn** are stored in a file ./setup/lockin.ini, which is available through the web-interface:

```
http://192.168.1.7/setup/lockin.ini
```

This file contains:

```
Amplitude: 0.0200  
Frequency: 1001.00  
Timeconstant: 9  
Rolloff: 24  
InputRange: 1  
Phase: 10.0  
Harmonic: 1  
AmplCaretPos: 3  
FreqCaretPos: 4  
FreqCaretPosTail: 2  
PhaseCaretPos: 5  
DisplayChannel: 2  
Channel1Type: 3  
Channel1Range: 120  
Channel2Type: 4  
Channel2Range: 7  
Channel3Type: 26  
Channel3Range: 120  
Channel4Type: 27  
Channel4Range: 7  
InputCouple: 0  
InputMode: 2  
RefInFlag: 0  
TimeConstLoL: 6  
RollOffLoL: 24  
SyncLoL: 0
```

```
Sync0: 0
Phase0: 0.0
EndOK
```

Frequency and amplitude are given as values in Hz and in V, respectively, while the phase is shown in degree. All other data are stored in the numeric code provided in the tables in section 4.1.3 .

Additionally, the state of the internal overload is provided by sending:  
<http://192.168.1.7/cgi-bin/remote.cgi>

As result, the eLockin sends

"idle 0"	if no overload is active
"idle 1"	if the frequency range is exceeded (blue exclamation mark)
"idle 2"	if the AD-converter input is in overload (red exclamation mark)
"idle 3"	if both overload states are active.

#### 4.1.8 EXAMPLE PROGRAM: LOCKINREMOTE

LockInRemote is an example program written in Borland Delphi 6 and designed to show the possibilities of remote-controlling an *eLockIn*. Its graphical user interface (GUI) looks similar to the front panel and each button works the same way the real one does. LockInRemote uses a TClientSocket object to set up a TCP connection to the *eLockIn*, working in non-blocking mode.

After starting the program, the screen will still be empty, because authentication is required first. Click **LockIn**→**Authenticate** to open the authentication menu. Enter your user name, password and the *eLockIn*'s IP and click '**Connect**'. If the

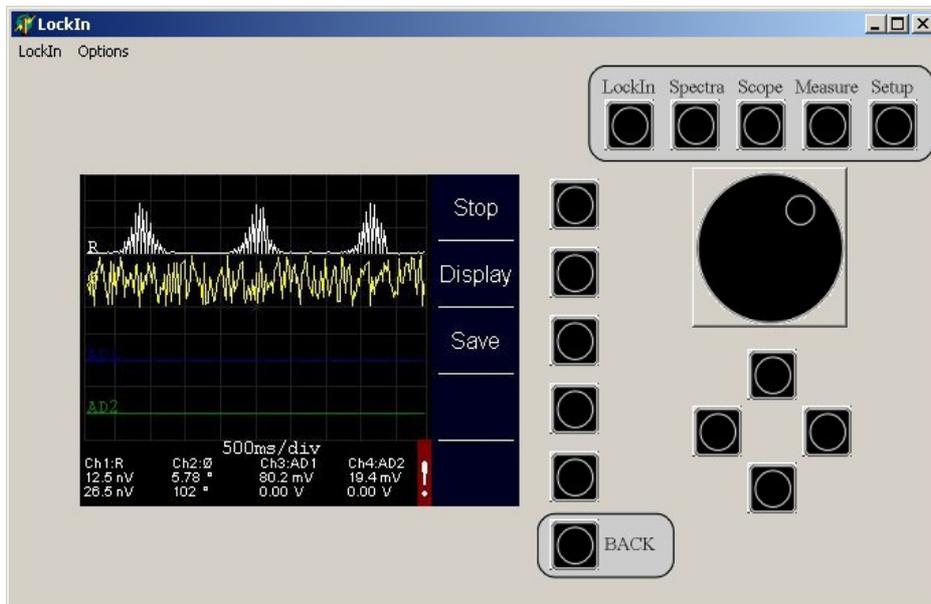


Figure 13: LockInRemote program surface.

*eLockIn* is found and your authentication data are valid, you will be logged in.

Now you may want to select '**Auto Refresh...**' from the '**Options**' menu in order to choose how the *eLockIn*'s data should be presented:

- Auto-Refresh Screen causes the program to retrieve a new screenshot continuously. This is useful if you want to navigate through the menu tree.
- Auto-Refresh Channel displays current channel data
- Auto-Refresh Off disables Auto-Refresh

It is possible to get new screen/channel data by refreshing them manually as well.

The '**Options**' menu also offers a simple speed test. Click '**Test Speed**' to check the abilities of your network connection. Choose a number and click '**Test**'. The same picture will be retrieved *n* times from the *eLockIn*, while time is measured. Later, speed and time per picture will be displayed.

All the buttons on the right side of the panel have the same functionality as their counterparts on the *eLockIn*. The row of buttons next to the screen represents different meanings, depending on which option is currently assigned to them on the display.

If you need to pass parameters to the *eLockIn*, choose '**Set Value**' from the '**LockIn**' menu. Enter parameters/values as described above (no underscores needed here!).

## 4.2 USE CGI-FUNCTIONS VIA WEB BROWSER

Web browsers are also able to address the CGI-interface of the *eLockIn* directly. The commands are based on the structure described in the previous chapter. The only difference is that the web browser needs the IP-address of the *eLockIn* for every command.

Example: Set the reference frequency to 100 kHz using the *remote.cgi* with the following parameter set:

```
http://192.168.1.7/cgi-bin/remote.cgi?8DD_100000_8
```

After '?', '8DD' indicates that the 'LockIn', the 'Ref' and the 'f [kHz]' menus are entered one after the other. The underscore is the spacer for the frequency value '100000'. With the '8' command, the *eLockIn* is turned back to the 'LockIn' main menu.

### 4.3 EXPORTED FUNCTIONS OF THE eLOCKIN.DLL

The functions exported by the dll are described in the following table:

<b>Name</b>	<b>Input</b>	<b>Output</b>	<b>Description</b>
<b>_Init</b>	IP-address type: pChar	No Return value.	Initializes the <i>eLockIn</i> by calling the current IP address.
IT IS ABSOLUTELY MANDATORY TO CALL THIS _Init-FUNCTION AS VERY FIRST FUNCTION BEFORE ALL OTHER FUNCTIONS CAN BE USED!!			
<b>_GetLockInData</b>	Channel [1 to 4] type: Integer	Value [real number] type: double	Returns the measured <i>Value</i> of the <i>Channel</i> as real number. ***
<b>_SetLockInInpRange</b>	Range type: double	No Return value.	Switches the input range between low noise (1), normal (10) and high dynamic reserve (100) -- see table on page 42 --
<b>_SetLockInCouple</b>	Couple type: double	No Return value.	Switches the input coupling between dc (0) and ac (1). -- see table on page 42 --
<b>_SetLockInFilterTime</b>	Time constant type: double	No Return value.	Switches the value of the filter. -- see tables on page 43 --
<b>_SetLockInSyncFilter (Vers. &gt; 1.0.0.2)</b>	Number of cycles: double	No Return value.	Switches the number of cycles for the sync filter. Values from 0 (Off) to 255 -- see tables on page 44 --
<b>_SetLockInFilterSlope</b>	Filter Slope type: double	No Return value.	Switches the roll-off of the filter. -- see table on page 44 --
<b>_SetLockInUac</b>	Uac type: double	No Return value.	Sets the amplitude of the reference output to a values given as double.
<b>_SetLockInFreq</b>	Frequency type: double	No Return value.	Sets the frequency of the reference output to a values given as double.
<b>_SetLockInPhase</b>	Phase type: double	No Return value.	Sets the phase offset of the internal oscillator with respect to the analyzed signal. The values are given as double.
<b>_SetLockInHarm</b>	Harmonic type: double	No Return value.	Sets the harmonic at which the signal is analyzed. The values are given as double.

_SetLockInAuxOut	Channel: [4 to 7] Data: type: double	No Return value	Sets the auxiliary output No 5 to 8. Channel = 4 equals „AuxOut 5“ at the backpanel of the eLockIn.
_SendKeyCode (DLL Vers. > 1.0.0.5)	Text: type: char		Allows to send any Key Code (see paragraph 4.2 and 4.1.3 ) to the device, so that any function can be called with the DLL.

\*\*\* In order to obtain values from channel 2 to 4, one has to call channel 1 first, because the channels 2 to 4 are not taken directly from the *eLockIn* via TCP/IP. When calling „\_\_GetLockInData“ for channel 1, the data of the other three channels simultaneously are transferred and stored locally in the memory. A subsequent call of „\_\_GetLockInData“ for the channels 2 to 4 takes the values from the memory, only.

## 4.4 REMOTE BY MATLAB

The remote control of the eLockIn by Matlab employs the DLL and requires a C-compatible header file "eLockIn.h". This header file shows all functions implemented in the DLL:

```
void _SetLockInInpRange (double Range);
void _SetLockInCouple (double Couple);
void _SetLockInFilterTime (double FilterTime);
void _SetLockInSyncFilter (double SyncFilter);
void _SetLockInFilterSlope (double FilterSlope);
void _SetLockInUac (double Uac);
void _SetLockInFreq (double Freq);
void _SetLockInPhase (double Phase);
void _SetLockInHarm (double Harm);
void _SetLockInAuxOut (long Channel, double AuxOut);

void _Init (char* IPasText);

double _GetLockInData (long Channel);
void _SendKeyCode (char* Text);
```

For the DLL Version 1.0.0.5, these functions have been tested to operate well with MatLab. Note that parameters given as "double" are requested to be provided as floating point number in Matlab, even if the number is an integer.

**IMPORTANT NOTE:** the function "\_Init" is the required to be called always (once per session) before any other function can be used.

Inside the directory "Remote\_by\_Matlab" is an example in Matlab that calls all these functions. The description of their parameters is given in the table on page 62.

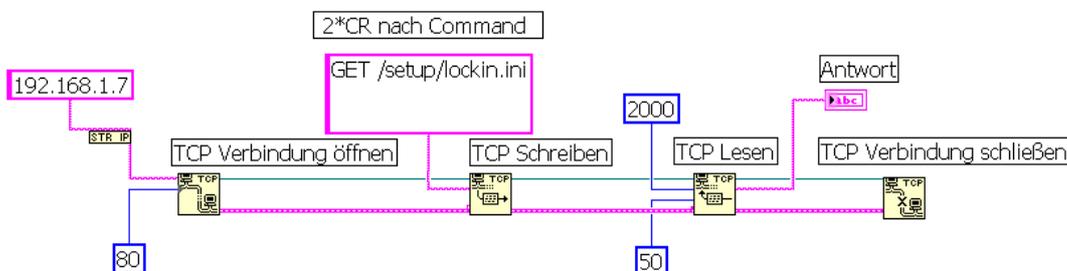
## 4.5 REMOTE CONTROL FOR THE eLOCKIN IN LABVIEW

For the remote control of the **eLockIn** under LabView, one can use the DLL-functions or one can address the **eLockIn** via TCP/IP or VISA communication. For the DLL based access, the following programs are provided:

- eLockin.dll provides the interface function
- eLockin.vi main program, which starts the communication and uses the Get\_eLockin.vi and the Ref\_eLockin.vi
- Get\_eLockin.vi data interface, which takes the data from the eLockIn
- Ref\_eLockin.vi settings of the reference output
- TestAuxOut.vi example for the AuxOut usage

All programs should be in the same program directory.  
For the TCP/IP access, the program tcpip.vi is provided.

### 4.5.1 TCP/IP ACCESS BY LABVIEW



This access type is demonstrated in the examples:

- tcpip.vi reads the lia.dat via a TCP-IP connection
- Keycontrol.vi demonstrates the key code use at the example of time constant
- Autophase.vi demonstrates the key code use for adjusting the phase offset
- tcpip\_settings.vi reads the lockin.ini via a TCP-IP connection

#### „TCP Open Connection“ == „TCP Verbindung öffnen“

##### inputs:

IP address of the eLockIn device --> purple --> 192.168.1.7  
Communication port --> blue --> 80

##### outputs:

Connection ID output --> thin green  
Error Output (Fehlerausgang) --> fat purple

#### „TCP Write“ == „TCP Schreiben“

##### inputs:

Connection ID --> thin green  
Error input --> fat purple  
Data input --> thin purple  
--> „GET /setup/lockin.ini **CR CR**“

**outputs:**

Connection ID output --> thin green  
Error Output (Fehlerausgang) --> fat purple

**„TCP Read“ == „TCP Lesen“**

**inputs:**

Connection ID --> thin green  
Error input --> fat purple  
Bytes to read --> blue --> 2000  
Timeout in ms --> blue --> 50

**outputs:**

Connection ID output --> thin green  
Data Output --> thin purple (String) --> displayed in screen  
Error Output (Fehlerausgang) --> fat purple

**„TCP Connection Close“ == „TCP Verbindung schließen“**

**inputs:**

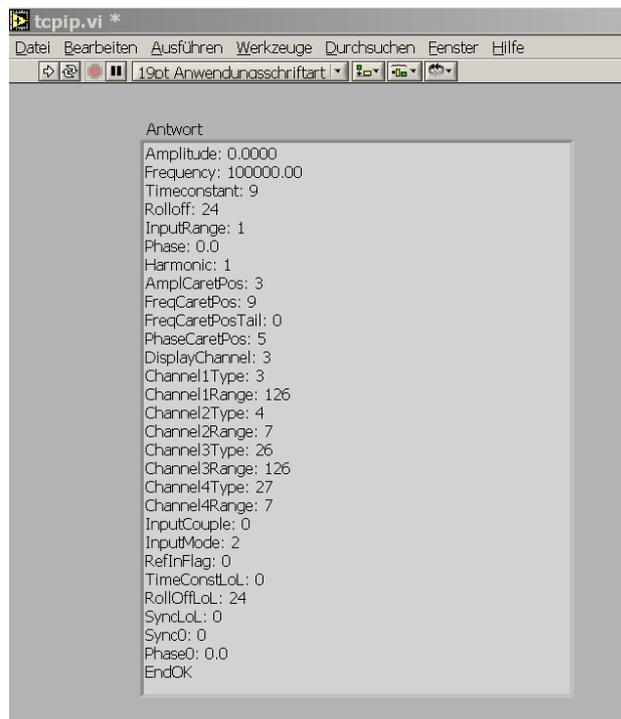
Connection ID --> thin green  
Error input --> fat purple

**Answers from the eLockIn:**

Result for „GET /setup/lockin.ini CR CR“



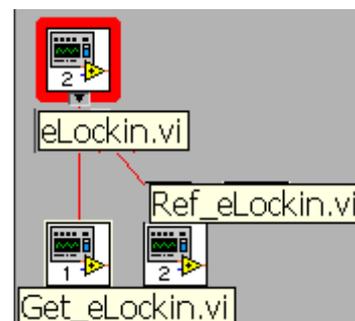
Result for „GET /setup/lockin.ini CR CR“



**4.5.2 DLL BASED ACCESS BY LABVIEW**

**eLockin.vi hierarchie**

The main program eLockin.vi contains a sequence in three steps and calls two sub-VI's: Ref\_eLockin.vi and Get\_eLockin.vi.



During run, it calls a three-step sequence:

- **Step 0:** eLockin.vi starts the communication with the lockin. The IP address is a value entry in the diagram used as input for the library function call „\_\_Init“. Then, the system waits for 500 ms to let the *eLockIn* initialize.
- **Step 1:** The Sub-VI „Ref\_eLockin.vi“ is called to set reference voltage and reference frequency. The other two functions of this Sub-VI, phase and harmonic, are not used here.
- **Step 2:** A curve diagram is drawn with 100 values, which are taken from the Sub-VI „Get\_eLockin.vi“, which uses the function call of „\_\_GetLockInData2 for the channel number 1. Between the single data, 100 ms delay time is given.

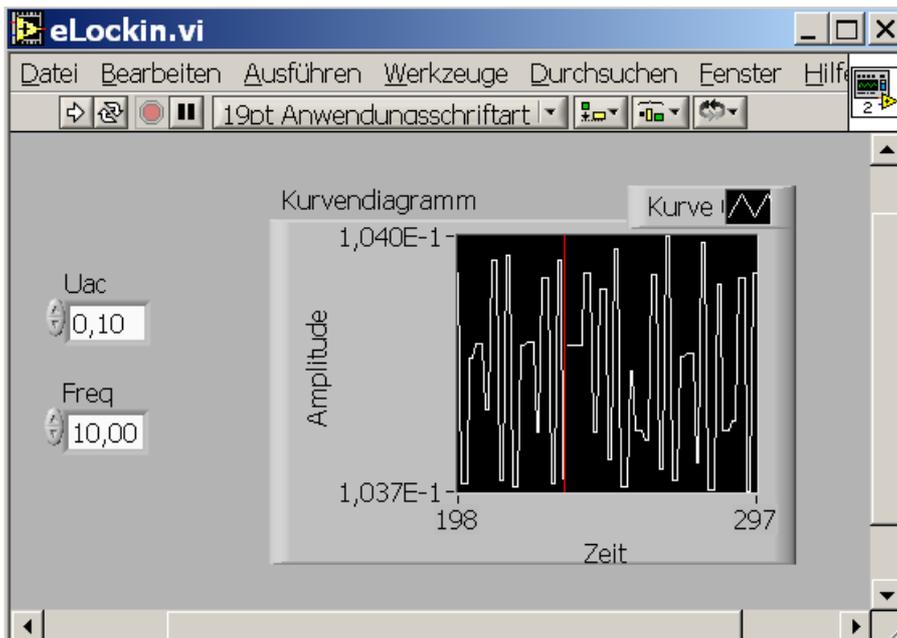
The example does not use all provided DLL-functions, but it shows how to implement them in own applications.

### Usage of the LabView control

To start LabView, double-click on the icon or program eLockin.vi.

The main program shows a curve diagram and two input windows for the reference amplitude and reference frequency:

The entered values are overtaken, when the start knob  is applied.



The same knob starts the acquisition of data in the curve diagram for one time cycle.

### TestAuxOut.vi

This is an additional example program, which shows how to use the function „\_SetLockInAuxOut“. The number taken as „Channel“ are counted from zero, while the labeling at the backside of the instrument count the AuxOut-Pins from One. Therefore, the 5<sup>th</sup> output named „DA5“ in the software is addressed from Channel = 4.

Remark: In order to check your settings, use [LockIn/Display/..] to set one out of the four lockin displays to the addressed channel (e.g. „DA5“).

## 4.6 REMOTE CONTROL FOR THE eLOCKIN IN PYTHON

In Python, the eLockIn can be remote controlled by simply addressing its web-interface with the key-codes translated by the remote.cgi (or remote2.cgi). The provided python examples are tested in python 3.5.2 with matplotlib installed.

Depending on which version of the Remote-CGI is used, you have to import eLockinRemote for remote.cgi or eLockinRemote2 for remote2.cgi in Python. The function calls defined in the main library are similar for both versions of the library.

Conveniently, the following py-files are provided:

- ***eLockinRemote.py/eLockinRemote2.py:***  
two versions of the main library, definition of function calls
- ***run\_spectra.py:***  
example run/stop spectra, get spectra data and plot
- ***function\_calls.py:***  
example for the usage of functions calls from the eLockinRemote library
- ***create\_gif.py:***  
example how to create a gif from a serie of current screen jpg images saved from web-interface; images will be saved to 'pics' directory

All files should be in the same program directory.

In any python script, first the eLockinRemote Library (or eLockinRemote2) need to be imported with

```
import eLockinRemote
```

and then connect to the eLockIn with the following call:

```
eLockinRemote.login()
```

before using any function calls. Any missing function can be easily added by following the scheme provided in the library. In order to address not yet described buttons and entries, simply transfer the key code and append a "\_0\_" for the first entry ("\_1\_" for 2nd entry and so on) or "\_xxx\_" for entering a number xxx.

If you're using a custom IP-address for the eLockIn, you have to change the IP in eLockinRemote.py (or eLockinRemote2.py).

## 5 SOFTWARE-UPDATE

**CAUTION:** The software update of the *eLockIn* can be compared with a **BIOS update** of a PC. One should spare the same caution. A power failure or power blackout during this procedure might cause a system crash, after which the *eLockIn* software needs to be restored in our facilities. **When this procedure fails, the instrument has to be shipped to us!**

If all precautions against power failure are done, follow these steps:

- ◆ Unpack the software update.
- ◆ Create a directory "update" (case-sensitive!).in the root on a USB stick.
- ◆ Copy all data into this directory.
- ◆ Plug the USB-Stick in the front panel of the eLockIn20-dual and wait a few seconds.
- ◆ Chose "SETUP". Press the knob besides "UPDATE". The update starts. The eLockIn204/2 displays all files which are uploaded from the stick in a list.
- ◆ The update is completed, if it displays "DONE".
- ◆ Switch the *eLockIn* OFF and wait for at least 5 to 10 s.
- ◆ Switch the instrument ON again. The new software should run.

## 6 REVISION HISTORY

### 6.1 MANUAL REVISIONS

This Manual is dated 05/09/22.

### 6.2 HARDWARE REVISIONS

The current hardware revision for the eLockin203 and eLockin204 is 211; for the eLockin205 it is 29.

The hardware revision is displayed at Menu > Setup.

### 6.3 SOFTWARE REVISION

Older software revisions do

- not allow to display the oscilloscope data vs. frequency

The current software revision is: 1.16d

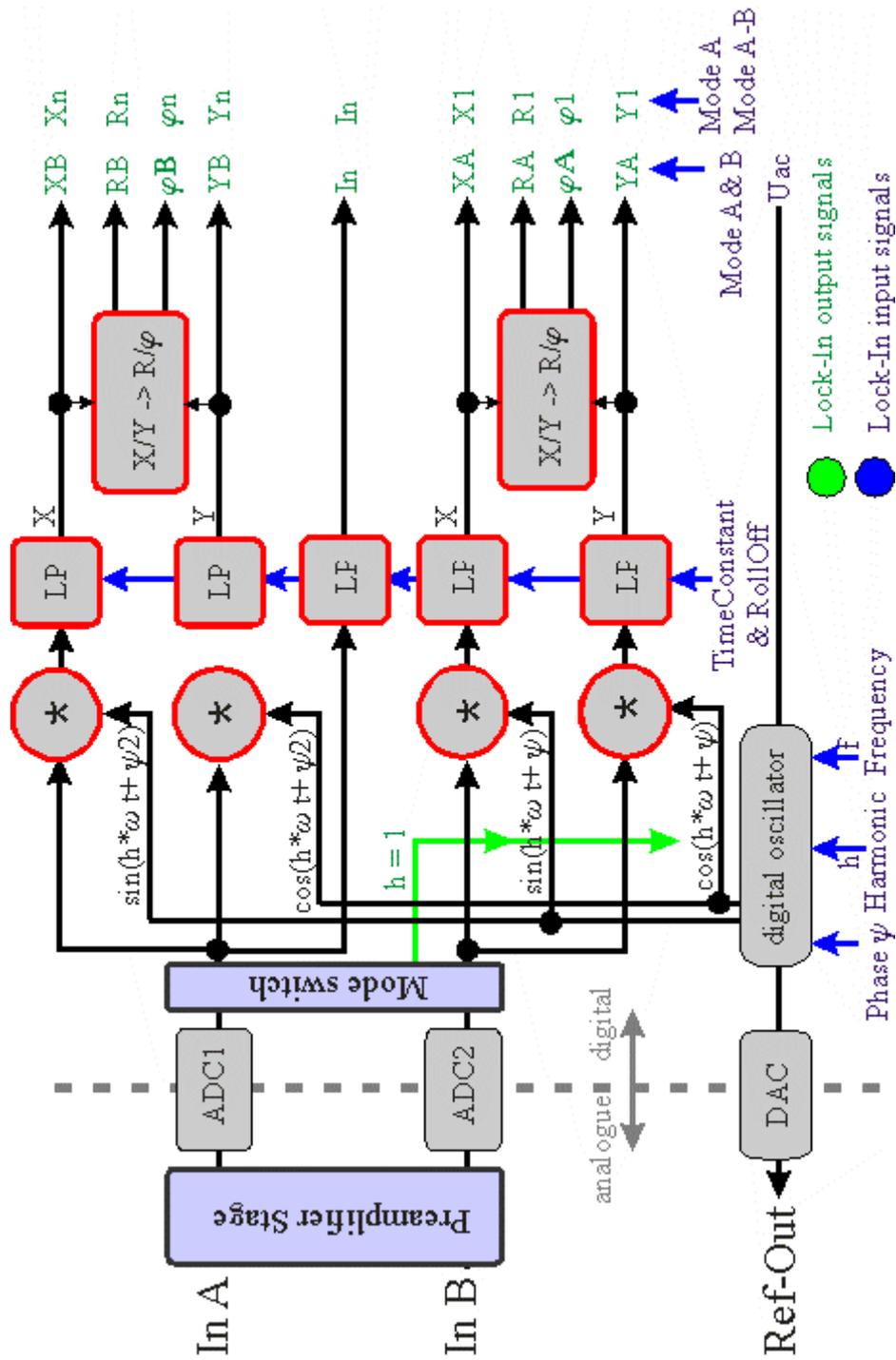
### 6.4 FIRMWARE REVISION

In this revision

- the sign of the phase of the lockin display is inverted, so that the phase lock caused by an RC-lowpass is displayed as negative value
- the algorithm for sync filtering has been improved

# 7 APPENDICES

## 7.1 SCHEMATIC DIAGRAM OF THE SIGNAL PATHS



## 7.2 DIAGRAM OF THE LOCKIN AMPLIFICATION PATHS.

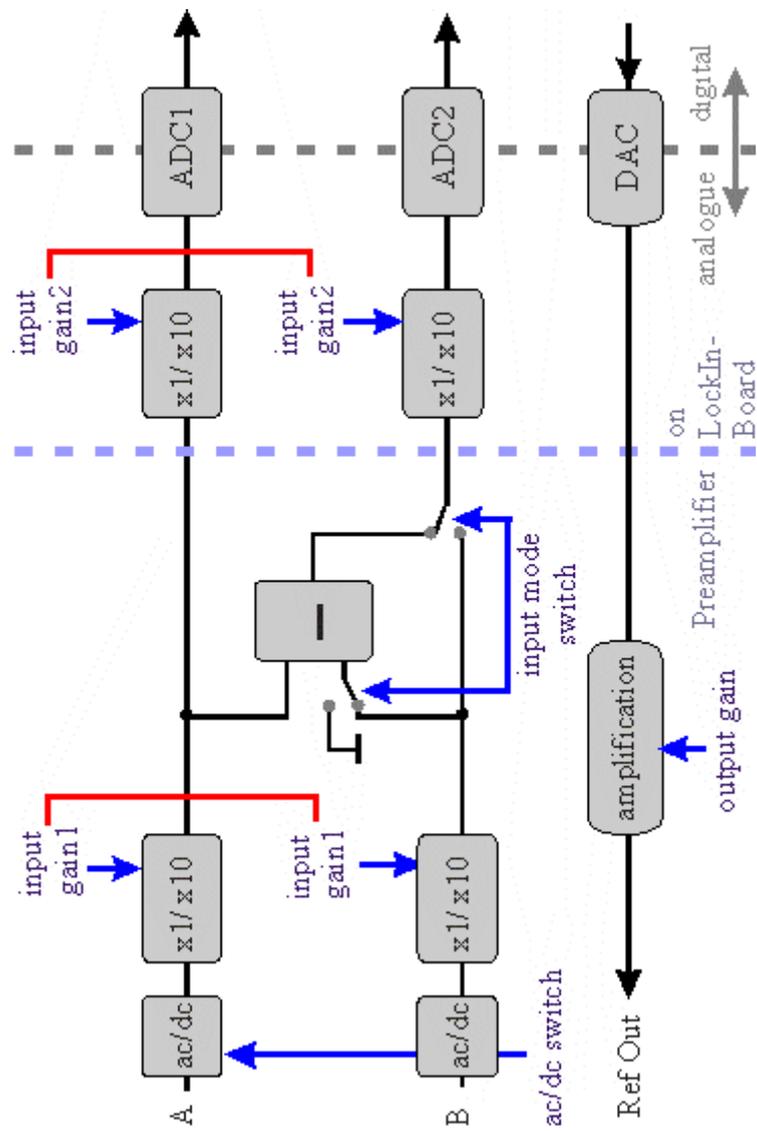
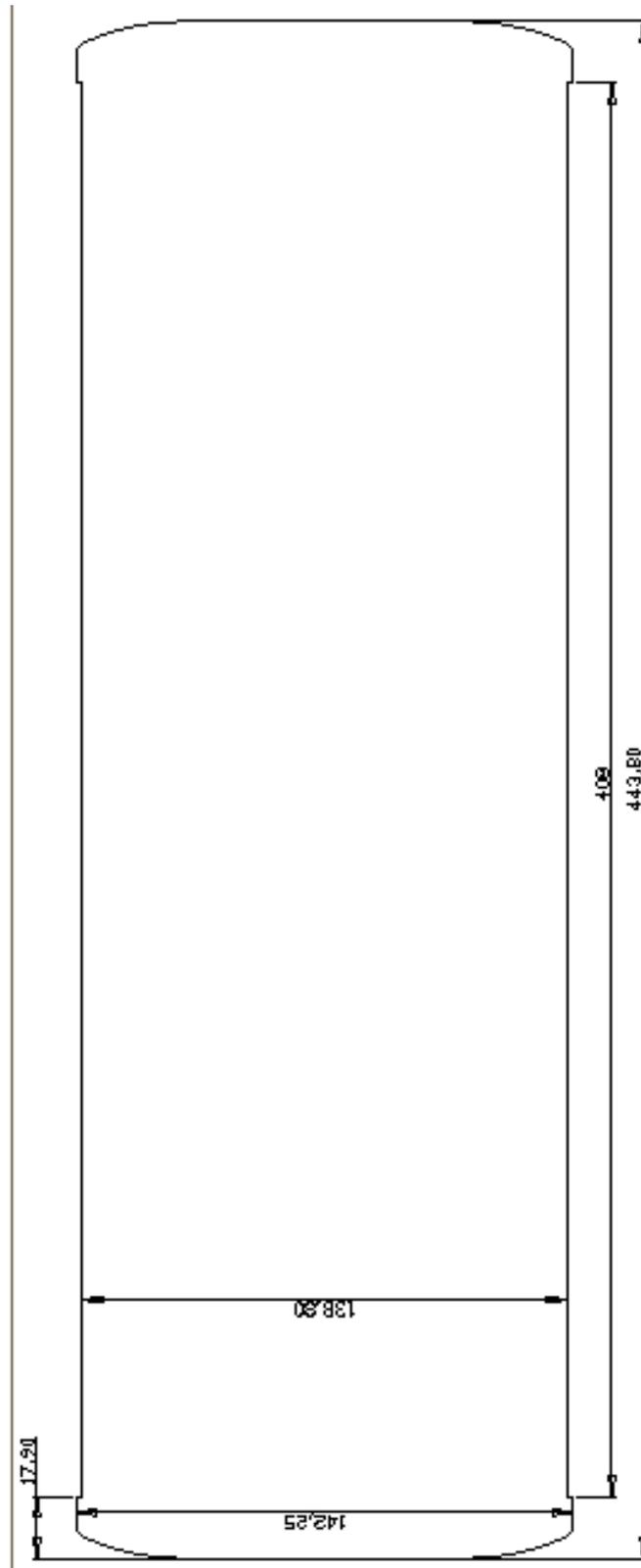


Diagram of the preamplifier switches.

### 7.3 FRONTPANEL DIMENSIONS

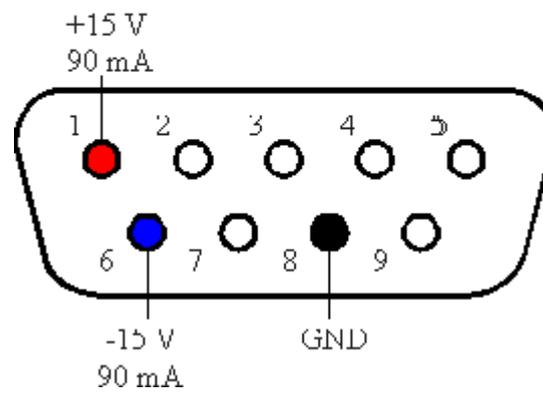


## 7.4 INSTRUMENT SPECIFIC TEST SHEETS

(insert with 10 pages of data)

## 7.5 PREAMPLIFIER CONNECTOR PINOUT

### Front-Panel “Pre-Amp”



## 7.6 CE CERTIFICATE

Anfatec Instruments AG  
Melanchthonstraße 28  
08606 Oelsnitz(V.)  
Germany

Traceable Calibration  
ELockIn .....  
Cal Date: .....

## CERTIFICATE OF TRACEABLE CALIBRATION

Certificate No: .....  
Manufacturer : Anfatec                      Model: eLockIn .....                      Serial No: .....  
Description: LockIn-Amplifier              Temperatur: ..... °C                      Humidity:..... %  
Cal Date: .....

Anfatec recommends Cal Interval : 12 Months

Instrument Condition: new product

Anfatec certifies above instrument meets or exceeds published measurement specifications and has been calibrated using standards traceable to the PRC National Institute of Metrology (NIM) and/or other National Metrology Institutes (NIST, NPL,PTB) that are linked to the international system of units (SI).

Calibration Procedure: eLockIn.seq, Version 1.0a

Calibration Equipment used:

Manufacturer/Model	Model Description	Serial Number
Tektronix AFG3021	Generator	C010315
Tektronix TDS3032B	Scope	B028247
Fluke 8845A	AC-Multimeter	9281063

Certified By: .....

Date Issued: .....

## 7.7 TRACEABILITY CHART

