

Lens Driver

Labview Instructions

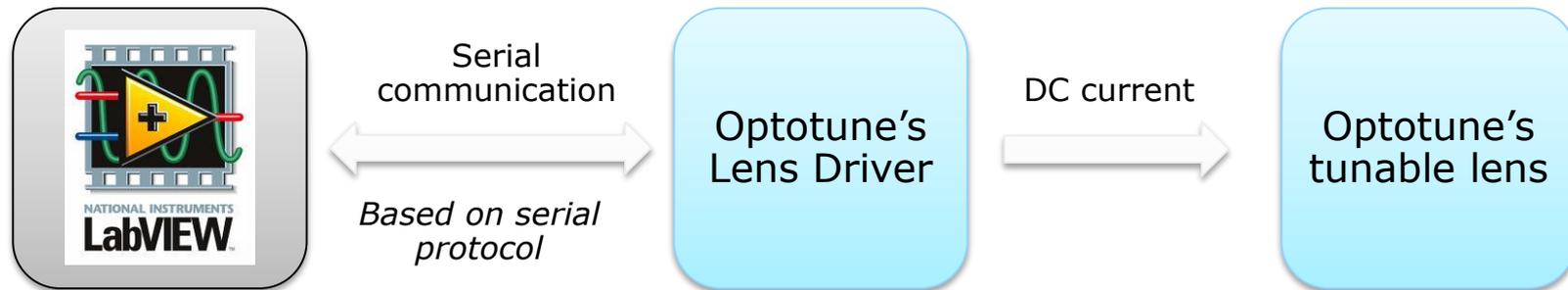
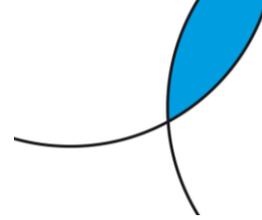
shaping the future of optics



May 2014
Dr. David Stadler
Application Engineer

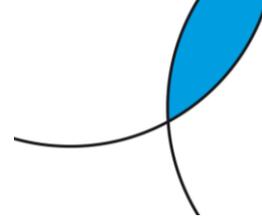
Bernstrasse 388 | CH-8953 Dietikon | Switzerland
Phone +41 58 856 3000 | www.optotune.com | info@optotune.com

Overview



- Before you start with the “Lens Driver” project, it is recommended to have a look at the “Hello Optotune Lens” program first
- This project is created in Labview 2013 SP1
- The project is a Labview implementation of Optotune’s Lens Driver software
- All Labview functions that are needed are included in the basic Labview package
- Communication is done via standard VISA port
- Contact sales@optotune.com for technical support

Front Panel



The screenshot shows the 'Lens Driver.vi' software interface. At the top, there is a menu bar (File, Edit, View, Project, Operate, Tools, Window, Help) and a toolbar with icons for running, pausing, and stopping. Below the menu bar, there are two status windows: 'error in (no error)' and 'error out', both showing a green checkmark and a source field. The main area is titled 'Optotune Lens Driver' and contains several sections: 'Signal A' and 'Signal B' settings, 'Focal Power' and 'Current dc B' sliders, 'Temperature low A' and 'Temperature high A' sliders, a 'Calibration 2' panel with various current and voltage settings, and a 'Data Log' panel at the bottom right. At the bottom of the interface, there are buttons for 'Get Temperature', 'Command', 'Fast Current', 'Reset', and 'End VI'.

If answer exists from Lens Driver it is displayed here

Note: 2 channels are implemented but Signal B is not available with current EL-E-4 hardware

Choose correct COM port

Choose operation mode

Set a focal power value. Range -5 to +10 dpt is not automatically adjusted based on temperature settings (like it is done in the Lens Driver software).

Temperature limits are required in focal power mode. Restricts the available range of diopters (see discussion in Lens Driver Manual).

Set DC current value

Set new Calibration values

Read Temperature from Sensor

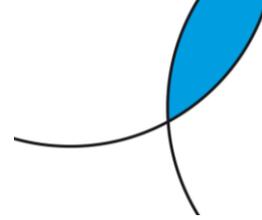
Shows the history of the serial commands sent to the Lens Driver.

Send command directly as byte command

Stop application

Possible fast DC current set

Front Panel

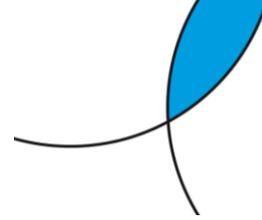


The screenshot shows the front panel of the 'Lens Driver.vi' software. The interface includes a menu bar (File, Edit, View, Project, Operate, Tools, Window, Help) and a toolbar with icons for running, pausing, and error handling. There are two error status boxes: 'error in (no error)' and 'error out', both showing a green checkmark and a status code. The 'error out' box shows code '107367629' and source 'VISA Read in write read Lens Driver.vi->Lens'. A 'remote' control is visible with a 'Data' field set to 'TB'. The main section is titled 'Optotune Lens Driver' and features controls for 'Signal A' (Sinus) and 'Signal B' (Square). Below these are frequency sliders for 'Frequency A' and 'Frequency B' (0.1 to 1000 Hz), and current limit sliders for 'Current hi A/B' and 'Current low A/B' (-200 to 200 mA). A 'VISA resource name' dropdown is set to 'COM4' with an 'ID' field set to '0'. A 'Calibration 2' panel shows values for Max Current A/B (292.84 mA), UCL A/B (292.769 mA), and LCL A/B (0 mA), with a 'Set Calibration' button. At the bottom, there are 'Temperature A' (24.6875) and 'Temperature B' (0 °C) readouts with a 'Get Temperature' button, an 'End VI' button, a 'Command' input field, a 'Fast Current' button, and a 'Reset' button. A 'Data Log' window at the bottom right lists parameters: MSB, MSA, MDB, MQB, TA, TB.

Different operation modes selected

Set modulation frequency

Set upper and lower current limit for frequency modes.



- The code contains the implementation of most of the serial commands
- Complete serial communication protocol can be obtained from Optotune
- “**Write Read Lens Driver**” vi implements communication at basic level
- Command list sent to Lens Driver only takes into account the values that have changed during run time
- This minimizes the required resources
- More detailed documentation is given **directly in the code**