Bore Controller

Licel GmbH

November 2, 2023

1 Introduction

In many LIDAR applications, daylight is among the major limiting parameters for the achievable signal range. By implementing a continuous monitoring and correction of the alignment, the telescope field of view can be reduced close to the laser beam divergence. This can improve operation of narrow field of view Raman or micropulse lidars and unattended operation. The LICEL bore sight alignment controller evaluates the image of the laser return from two user defined height ranges on a multi anode photomultiplier. It computes correction parameters for the beam steering.



The bore detection system consists of

- · the detector head and
- the Ethernet controller.

The detector head includes a multi-anode PMT, a high voltage supply and the counting electronics. The Ethernet controller sends the controlling data and receives the counting data via a 2m long VHDCI cable.

1.1 Software package

In addition to the hardware Licel provides the source code for 3 typical tasks,

- Automated alignment tracking and correction for a lidar (Boresite Alignment.vi)
- Live signal display (Bore_Live.vi)
- Pulse height distribution software to determine the optimum discriminator level (Bore_Pulse.vi).

1.1.1 Alignment Software

The alignment software *Boresite Alignment.vi* provides a standard shell that need to be adapted to the customer's drive control. The required changes are described in section 9 Software Preparation. Once the driver is adapted to specific drive and the opto-mechanical work outlined below is done, the lidar can be automatically kept at a predefined alignment. The idea behind the software is the following:

- The user defines an alignment as a known good position.
- The software will record the data and compute the current alignment.
- Based on this data the software will move the drives so that the system goes back to that known good position.

1.1.2 Live Display

The data is displayed as it arrives and shows how the signal on the four quadrants change when the lidar alignment changes.

1.1.3 Pulse Height

For a properly working system the discriminator level needs to be set. This level is dependent HV setting for the PMT and on the typical signal strength. In order to determine the optimum parameters the discriminator level can be varied when a constant signal level is applied and the resulting pulse height distribution can be used to determine the optimum parameter.

2 Preparations

2.1 Optical Layout

Before installing the bore sight module the optical and mechanical layout of the detection system should be defined. The system needs the field stop imaged on the cathode surface. **Note:** This is different from the usual requirement to image the primary mirror on the detector.

The lens parameters together with the position of the lens and the detector head define the image amplification. To ensure that the laser beam is not lost the full field stop image should be slightly less than the 4 quadrants of the bore sight controller.

2.2 Steering Mirror

The system provides the correction signals for a motorized steering mirror. If A and B Trigger are connected to two different lasers its possible to control two steering mirrors with one boresite detector. For a limited set of stepper drivers software modules are already available. These are currently:

- Newport ESP300 Series
- Newport Picomotor 8742
- Newport Conex-CC
- Newport SMC100
- PI QMotion E872

If your controller is not in this list you face two options:

- 1. Your write a shell around the driver commands as outlined in the section Software Preparation.
- 2. Licel writes this shell around the driver commands. This requires a quotation/order for the work and that you ship the controller to Licel. To work efficiently the controller should arrive at Licel 4 to 6 weeks before the intended shiping date of the boresite controller.

3 Wiring the Ethernet Controller



The Ethernet controller can be integrated into 19" racks (3 hight units). On the front panel 3 connectors are available:

- Trigger A input socket (Lemo Camac, 4 V @ 1 k Ω)
- Trigger B input socket (Lemo Camac, 4 V @ 1 kΩ)
- Ethernet connector, RJ45 connector for the connection to the controlling computer
- · Detector socket, connector for the VHDCI cable to the detector head

At the rear panel of the controller a power supply (+15 V, 200 mA / -5 V, 200 mA / +5 V, 200 mA) must be connected, either directly to the transient recorder supply or to an appropriate external supply.



4 Mechanical Dimensions

To verify the orientation of the PMT check the triangle mark at the rear side. It designates the -x axis.

5 Theory

The bore controller returns count rates for two height ranges for the four quadrants. This comes from the assumption that one part of the signal, typical the far field signal will be optimized. The background is a reference section and is needed for the offset correction. Each section has a start and a stop. Between them the counts are accumulated and averaged. So there are two sets of numbers. One set are the signal counts $S_{0...3}$ the other set are the background counts $B_{0...3}$. Both count rates can show pulse pile up and in this case need to be corrected for dead time effects.

To compute the background corrected count rates one has to subtract the B_i from the S_i first.

$$s_i = S_i - B_i \tag{1}$$

Assuming the coordinate (0,0) as the center of the four quadrant detector one can compute the center of gravity.

f gravity.

$$G_{x} = \frac{s_{1} - s_{2} + s_{3} - s_{4}}{(2)}$$

$$G_x = \frac{s_1 + s_2 + s_3 + s_4}{G_x}$$
(2)

$$f_y = \frac{s_1 + s_2}{s_1 + s_2 + s_3 + s_4} \tag{3}$$

A shift of the beam axis $d\alpha$ will result in a shift of the beam image of over the detector by $r * d\alpha$. This shift will cause a signal reduction in one section and a signal increase in another section so that

$$dG_x \propto 2 * d\alpha_x \tag{4}$$

$$dG_y \propto 2 * d\alpha_y \tag{5}$$

Each sector can have a slightly different sensitivity. So even for equal S_i there is not guaranty that the spot is in the center. After a proper alignment is found the current G_{xy} values are used instead as reference values. During the tracking the difference between the current alignment and the reference point is used as the control parameter for correcting the drives.

$$\Delta drive_x \propto G_x - G_{x0} \tag{6}$$

$$\Delta drive_y \propto G_y - G_{y0} \tag{7}$$

The coordinate system of the detector can be tilted against the movement axes of the drives. It is preferable to avoid this otherwise one has to perform a sinus cosines transformation of the correction factor.

6 Signal to Noise considerations

The alignment requires a valid signal. This can be defined in two ways:

- 1. The signal is significant, this means the Poisson noise of the signal is much smaller than the signal itself.
- 2. The signal is much stronger than the background.

The first item can be controlled by choosing a sufficient large signal region, the signal can be as much as 1024 bins of 100ns. Further increasing the number of acquired shots will increase the SNR. However it does not make sense to increase both parameters too much:

- If the signal region is too large to much of the background will be covered or to much of the near field will be included while the far field should be optimized.
- If the shot number is too large the reaction to changes will be slow.





The software uses:

$$C = \sum S_i * (MHz \text{ per mean count}) * (number of bins) * (number of shots)$$
(8)

to determine to overall number of counts in the signal. It computes the signal inherent SNR as

$$\frac{C}{\sqrt{C}}$$
 (9)

and the SNR with respect to the background as

$$\frac{\sum s_i}{B} \tag{10}$$

and applies the same threshold to both.

7 Operation

For a successful operation of the bore sensor the following steps need to be accomplished:

- 1. Make a ray tracing simulation to determine the optical setup to image the field stop on to the detector
- 2. Prepare the software modules necessary to control the stepper drivers (see section Software preparation)
- 3. Mount the detector.
- 4. Setup the network interface of the controller following the guidelines from the Licel Ethernet Controller - Installation and Reference Manual chapter 4. The bore controller chips with the default address of 10.49.234.234 as all Licel Controllers do. Please make sure that you can only setup one controller at a time as otherwise one would have two controllers with the same IP.
- 5. Align manually the system to a position, which will be used as the reference position
- 6. Turn the high voltage of the detector on (**recommended 850-900 V, maximum 1000 V**)
- 7. Set the discriminator level. Use the BorePulse height Module to optimize the threshold level.
- 8. Supply a trigger signal to the Bore-Controller
- 9. Verify with BoreLive that a valid signal is received on all 4 sectors of the detector.
- 10. Using BoreAlign take a signal profile to verify the optimum positions for the background and the signal.
- 11. Once all of those parameters are setup, use the BoreAlign module and activate the Keep Alignment button.

8 Multiple Mirror Configuration

The Bore controller has the opportunity to control two mirrors at the same time. To use multiple mirrors it is required to set the quantity of motors to 2 in the initialization file. Then the signals that go with Trigger A will control mirror 1 and signals that go with Trigger B will control the second mirror.

9 Software Preparation

9.1 Installing the Licel LabVIEW Sources

To install the Licel LabVIEW sources you may choose between the following options:

- Copy all files contained in the directory LabVIEW Files from the CD ROM to a directory of your choice.
- If you downloaded the Licel software from http://www.licel.com/lidar_components.html please unpack the content from the downloaded zip file and copy it to a directory of your choice (keep all directory hierarchies!).

Please note that in the case the software is copied from a CD you may have to unselect the "Read-only" attribute for the destination folder.

1. This is done by selecting the directory and right-clicking on it. Select **Properties** from the context menu.

x



2. Verify that the "Read-only" attribute is not checked, uncheck it if necessary. Click OK and check in the next dialog Apply changes to this folder, subfolders and files. Leave the dialog by clicking OK



Licel provides one or more user-defined LabVIEW error code files. LabVIEW will use these files to generate hints in error messages. Before you will have to copy these error code files to an appropriate location where LabVIEW will find them. For this

- 1. Locate the error code files in Licel's LabVIEW sources: they are located in the sub folder <LabVIEW Files Folder>\Files\user.lib\errors
- 2. Select all files *-errors.txt and copy them

Computer → OS (C	:) 🕨	LabVI	EW Fi	les ▶ Files ▶ use	er.lib 🕨 er	rors
Organize ▾ 🧊 Open ▾ Pr	int	ibrary	Ŧ	Share with 🔻	Burn	New to
S (C:) Apps Drivers Littel LabVIEW Files Cogs PerfLogs Program Files Program Files (x86) temp Users Windows C DVD RW Drive (E:) Licel Softward		Nar	Licel	Print Edit Open with Restore previous Send to Cut Copy Create shortcut Delete Rename	versions	Þ
Licel-errors Date mode	ified Size	: 24.08. : 10,6 K	2017 1 B	Properties [1:12 Date	e created: 1	9.08.2020

3. Navigate to the folder

<LabVIEW installation directory>\user.lib\errors, create the sub folder errors if necessary. Paste the copied file(s) to that directory



4. All copied LabVIEW error code files should be seen now:



Now you should be able to run all the files. If you are still having problems, apply a mass compile to the directory where the software was extracted to:

- 1. Start LabVIEW.
- 2. Select the menu entry Open Project ... in the File menu

tabVIEW	-		Х
File Operate Tools Help			
New VI Ctrl+N New Open Ctrl+O W* 2016 Search		(R
Create Project			
Open Project Recent Project Becent Files Open Existing			
Exit Ctrl+Q All Recent Files		•	
Blank VI			
v			Ŧ
 Find Drivers and Add-ons Connect to devices and expand the functionality of LabVIEW. Community and Support Participate in the discussion forums or request technical support. Welcome to La Learn to use LabVIE from previous version 	bVIEW W and u	ıpgrade	
LabVIEW News			

3. Navigate to the folder where you copied the Licel LabVIEW sources to

📴 Select a File to	Open							Х
$\leftarrow \rightarrow \cdot \uparrow$	K Lice	I Bore → project		~	Ū		project	
Organize 🔻	New folder							
📥 OneDrive	^	Name	^		Date mo	odified	Туре	
💻 This PC		👪 Bore_src			12/6/202	21 12:31 PM	LVPROJ	File
3D Object	s			Type: LVPI Size: 37.3 Date mod	ROJ File KB	/6/2021 12:31 PM	4	
🔮 Documen	ts			buternoo	incar re,	0,2021120111	<u> </u>	
🕹 Download	ls							
🁌 Music								
Pictures								
Videos 🛛								
📒 OS (C:)								
🛖 Shared Fo	lders (
i Network	v ·	C						>
	File nar	ne: Bore_src			~	All LabVIEW	Files (*.vi;*.c	:tl;*.vit;
						ОК	С	ancel

4. Open the project Bore_src.lvproj in the subfolder project



5. Select the menu Tools, then Advanced, and finally Mass Compile....

Bore_src.lvproj - Project Explorer	− □ × Tools Window Help Measurement & Automation Explorer	
Items Files Files Project: Bore_src.lvproj Wy Computer Uicel Bore Acquisition Witual Controller Uitual Controller Dependencies Build Specifications	Instrumentation MathScript Window Compare Merge Profile Security User Name Build Application (EXE) from VI Source Control	
	VI Analyzer LLB Manager Import Shared Variable Distributed System Manager Find VIs on Disk	
	Prepare Example VIs for NI Example Finder Remote Panel Connection Manager Web Publishing Tool Control and Simulation Create Data Link Find LabVIEW Add-ons VI Package Manager Vision Assistant	
	Advanced > Options	Mass Compile Clear Compiled Object Cacheks Edit Error Codes Edit Palette Set Create or Edit Express VI Export Strings Import Strings

6. You will be asked to select a folder, select the **source** folder under the target directory of your LabVIEW files.

📭 Choose a directory to c	ompile				×
← → • ↑ 📙 « L	icel Bore	e > source > v	کې کې	o source durchs	uchen
Organisieren 🔻 Neu	er Ordne	er		==	- 🔳 🕜
project ^	Na	me	1	Änderungsdatum	Тур ^
len OneDrive		BoreAnalysis.IIb	:	23.06.2022 13:55	Dateio
		BoreLive.IIb	1	23.06.2022 13:55	Dateio
Dieser PC		BorePulse.IIb	1	23.06.2022 13:55	Dateio
🧊 3D-Objekte		BoreRemoteControl	1	23.06.2022 13:55	Dateio
E Bilder		GCS_Q-Motion	1	23.06.2022 13:55	Dateio
E. Desktop		LicelFile.llb	1	23.06.2022 13:55	Dateio
Dokumente		LicelModule.llb	1	23.06.2022 13:55	Dateio
		LiceITCPIP.IIb	1	23.06.2022 13:55	Dateio
Musik		LiceITCPIP_API.IIb	1	23.06.2022 13:55	Dateio
		LicelUtil.IIb	1	23.06.2022 13:55	Dateio
Videos		Picomotor_8742_Drive_dll		23.06.2022 13:55	Dateio
🛄 OS (C:)		SearchControllers.llb	1	23.06.2022 13:55	Dateio
🛖 Shared Folders (-	Bore Drive		16.06.2022 15:11	LabVIE 🗸
×	<				>
Date	iname:				~
		Verzeichnis wa	ählen	Öffnen	Abbrechen

7. Press Mass Compile in the next dialog.

📭 Mass Cor	mpile	🗙 📑 Mass Compile	×
Options	Status	Options Status	
Directo C:\L	ory to cor .abVIEW\L	npile #### Starting Mass Compile: Do, 23. Jun 2022 14:19:54 Directory: "C:\LabVIEW\Licel Bore\source" #### Finished Mass Compile: Do, 23. Jun 2022 14:20:03	•
Log R	esults		
Cache	e VIs 10	⊠ Append	
	de warning	Is for read-only VIs Mass Compile Done Help Mass Compile Done Help	•

8. Later the mass compile status will be shown.

Source Organization

The Bore LabVIEW project consists of several virtual directories:



- Licel Bore Acquisition
 - TCPIP TCPIP Utilities used for the the Network Setup in the Ethernet Manual
 - Bore main Bore applications Bore Live, Bore Pulse, and Boresite Alignment (described in the following sections)
 - Bore Drive Controls contains LLBs or directories for the control of different stepper drives. Each of them has a control VI as described in the following subsections
- Virtual Controller simulation program
- user.lib error definition files (see above)

In the following subsections the standard control VIs contained in Bore Drive.llb will be described. The bore controller gives the input for a steering system. The alignment application *Boresite Alignment* and a set of VIs is used to bring the system to control the stepping drives. All these VIs call *the same control VI* and pass a requested action to it. Dependent on this action the control VI organizes the communication with the drive controller.

9.2 Control VI

The control VI is loaded on start by Boresite Alignment.vi. The following entry in the initialization file BoreAnalysis.ini or BoreAnalysis2.ini defines the path of the control VI. The VI given here is found in any IIb or directory residing in the Bore directory.

```
[Drive]
ControlVI="BoreDrive NoDrive Control.vi"
;; ControlVI = "Newport Control.vi"
;; ControlVI = "NewportConexCC Control.vi"
;; ControlVI = "NewportSMC100 Control.vi"
;; ControlVI = "Picomotor_8742_Control.vi"
;; ControlVI = "QMotion_E872_Control.vi"
```

Shown here is the standard entry which will result in loading

Bore Drive.llb\BoreDrive NoDrive Control.vi. The other supported control VIs are commented out with the ;; at the beginning of a corresponding line. The active BoreDrive NoDrive Control.vi will establish no communication with any drive controller but is intended to be used as a template for adding your own code in a LabVIEW case structure depending on the requested action. Furthermore it can be used when one wants to inspect the communication of the *Boresite Alignment* to the Bore controller for testing purposes but don't want to use active steering.

The control VI must have the following wiring scheme:



The place to insert specific LabVIEW code (e.g. sub-VIs from a driver-IIb) is found in the block diagram of the control VI. Each action corresponds to a case of the inner case structure.



The shell VIs calling the control VI are integrated into the *Boresite Aligment* module. In addition they have simulation code that interfaces with the Virtual Controller and allows to see the VI's in action without a controlling a real system.

The control VI is called by the following Shell VIs:

VI name		Action
BoreDrive	InitializeDrive.vi	
BoreDrive	GetCurrentPosXY.vi	
BoreDrive	LimitSwitch.vi	
BoreDrive	StartToXY.vi	
BoreDrive	Brake.vi	
BoreDrive	CloseDrive.vi	
BoreDrive	GetStatus.vi	
BoreDrive	StartHomeRun.vi	

Furthermore the shell VI BoreDrive DeviationToXY.vi is provided.

9.3 Initialize – BoreDrive InitializeDrive.vi

This vi contains the initialization routines for the xy translation stages. It should include any required initial movement or should return information that a home run is required to run the translation stage into a referenced position. For this the VI returns the device status as <code>BoreDrive GetStatus.vi</code> does, as well.

Any required references (e.g. for a VISA session) should be kept in shift registers inside the used control VI. Some legacy control VIs use global variables for this.

The global variable *StepperDriverSim* should be set to false to work with a real controller. If the simulation is activated the software will try to connect to a ethernet controller. This should be the Virtual Controller (a software module simulating the hardware module - see Licel Ethernet Controller Installation and Reference Manual chapter 8).

9.4 BoreDrive GetCurrentPosXY.vi

This VI should return the positions of the xy translation stages in their own coordinates (e.g. stepper counts, encoder counts...)

9.5 BoreDrive LimitSwitch.vi

This VI should return the status of the limit switches. A limit switch is considered active if a true boolean value is returned.

9.6 BoreDrive StartToXY.vi

Starts the movement of one or both drives to the desired positions

9.7 BoreDrive Brake.vi

This VI contains the routines to stop the drives.

9.8 BoreDrive CloseDrive.vi

This VI contains the routines for the xy translation stages that need to be called on exit.

9.9 BoreDrive DeviationToXY.vi

Compute the new target positions for both drives based on the current position, the target point, and the current alignment point. This VI does not have a direct hardware interface but it relates drive coordinates and the angular alignment parameter as described in Eqn. 6.

9.10 BoreDrive GetStatus.vi

The VI returns the device status with the following bitwise information:

- 0x01 0 needs initialization, 1 initialized
- 0x02 0 homing required, 1 no homing required (referenced)
- 0x03 0 no home run active, 1 home run active

9.11 BoreDrive StartHomeRun.vi

This VI starts a home run of the translation stages. This may be necessary for the Newport SMC100 and Newport Conex-CC controllers. After starting a home run the device status should frequently be read until it has the value 0x0011.

10 BoreLive.vi

You need to setup first the controller IP

Live Display TCP/IP			
IP Address	Port	Timeout	
10.49.234.234	2055	5000	Reconnect

Run the VI using LabVIEW's run 🖄 button.

If the IP Address and Port match the settings of the Boresite Controller and the network is correctly working

the TCPIP LED indicator should change: 6 TCPIP

If not, please check the IP

TCPIP

Address and Port, you may modify them while the VI is running.

If you recognize that you are not connected to the Ethernet Controller you would like to use (because you are operating two Boresite Controllers), just

- 1. Enter the *IP Address* and *Port* of that controller you really would like to connect to and
- 2. press the Reconnect button.

Also use the *Reconnect* button after changing the *IP Address* and/or *Port* while no connection is active.

Turn the detector *High Voltage* on (a recommended value is around 850-900 V). Set the *Discriminator* level, select the *Trigger* and adjust the signal and background ranges to your needs.



This module will give you a live sensor signal without any controlling of the drives. The Trigger selector will switch between the signals for Trigger A and B.

11 BorePulse.vi

This VI measures a pulse height distribution (PHD) for the detector. From the PHD one can determine the optimum discriminator level. The software is equivalent to the pulse height distribution for transient recorder. You need to setup first the controller IP

Pulse Distribution	TCP/IP			
IP Address		Port	Timeout	
10.49.234.234		2055	5000	Reconnect

Run the VI using LabVIEW's run 🔄 button.

If the IP Address and Port match the settings of the Boresite Controller and the network is correctly working

the TCPIP LED indicator should change:

TCPIP \rightarrow O TCPIP	If not, please check the IP
-----------------------------	-----------------------------

Address and Port, you may modify them while the VI is running.

If you recognize that you are not connected to the Ethernet Controller you would like to use (because you are operating two Boresite Controllers), just

- 1. Enter the IP Address and Port of that controller you really would like to connect to and
- 2. press the Reconnect button.

Also use the *Reconnect* button after changing the *IP Address* and/or *Port* while no connection is active. Once the connection with the controller is established the front panel will appear as follows:



Choose the *Trigger* channel, set the *High Voltage* of the detector, the signal region where you want to have the two range bins and the number of *Shots to Average*. After pressing the *Start* button a pulse height distribution (PHD) at the selected *Trigger* will be recorded.

12 Dual Boresite Alignment.vi

This VI is considered to start multiple instances of BoreSiteAlignment.vi using the selected initialization files. With this it is possible to control maximum 4 mirrors (2 mirrors per Boresite Alignment Application). It is important to choose *different* initialization files for each *BoreSiteAlignment* instance before running the Application.



Run Dual Boresite Alignment.vi using LabVIEW's run 🔂 button.

12.1 Initialization File

The following settings are used in the initialization file Boreanalysis.ini or BoreAnalysis2.ini

```
[TCPIP]
UseValues = TRUE
IPAddress = "10.49.234.235"
Port = 2055
[TCPIP_API]
Active = TRUE
Port = 2060
[Signal]
backgroundStart = 5
backgroundStop = 10
SignalStart = 18
SignalStop = 25
Shots = 1
[PMT]
HV = 800
[Reference]
X = 0.000000
Y = 0.00000
XB = 0.000000
YB = 0.000000
[MotorStep]
X = 100,000000
Y = 100,000000
XB = 100,000000
YB = 100,000000
[Tracking]
MinSNR = 0,000000
```

[Drive]

```
Simulation = FALSE
SimulationIP = 127.0.0.1
ControlVI = "BoreDrive NoDrive Control.vi"
;; ControlVI = "Newport Control.vi"
;; ControlVI = "NewportConexCC Control.vi"
;; ControlVI = "NewportSMC100 Control.vi"
;; ControlVI = "Picomotor_8742_Control.vi"
;; ControlVI = "QMotion_E872_Control.vi"
MotorQuantity = 1
ToleranceX = 0.0005
ToleranceY = 0.0005
ToleranceXB = 0.0005
ToleranceYB = 0.0005
MinimumStepTime = 1.000000
FactorX = 0.01
FactorY = 0.03
FactorXB = 0.01
FactorYB = 0.03
DeadbandX = 0.00000
DeadbandY = 0.00000
```

The initialization file keys are read by the *Boresite Alignment* software right after starting it. Most of the parameters do not need to be edited manually in the initialization file– they will be set by the program on change. These keys are marked with as asterisk (*) in the following list:

[TCPIP]	TCP/IP related settings, read by Windows applications, only:
	 UseValues*: Use TCPIP values from the initialization file IPAddress*: TCPIP address Port*: TCPIP Port
[TCPIP_API]	TCP/IP API settings
[Signal]*	Background and Signal Start Stop Parameters
[PMT]*	${\tt HV}^{\star}$ is the gain voltage to be applied to the specified device
[References]*	 X, Y*: Point of reference alignment for mirror A, mirror A will read on signals with Trigger A
	• XB, YB*: Point of reference alignment for mirror B, mirror B will read on signals with Trigger B
[MotorStep]	motor steps to move from one alignment step to the next in motor-specific units (stepper or encoder counts)
	• X, Y: motor steps for mirror A
	• XB, YB: motor steps for mirror B
[Tracking]*	MinSNR* is the required signal/noise ratio for tracking operation
[Drive]	• Simulation: Simulate drive or not
	• SimulationIP: IP address of the Virtual Controller
	• ControlVI: Control VI to use. By default BoreDrive NoDrive Control.vi is enabled, the other supported VIs are included in the initialization file but commented out(;;).
	• MotorQuantity: Number of driven mirrors/sytems (value range from 1 to 2)
	• ToleranceX, ToleranceY*: Tolerances in motor coordinates (steps, encoder counts,) for mirror/system A, the motors will move if the the deviation of the current and target positions are larger

- ToleranceXB, ToleranceYB*: Tolerances in motor coordinates (steps, encoder counts, ...) for mirror/system B, the motors will move if the the deviation of the current and target positions are larger
- MinimumStepTime*: Minimum Time between subsequent motor movements
- FactorX, FactorY*: Factor to convert Bore detector coordinates to motor units for mirror/system A. These factors can be set negative!
- FactorXB, FactorYB*: Factor to convert Bore detector coordinates to motor units for mirror/system A. These factors can be set negative!
- DeadbandX, DeadbandY*: X- and Y-deadbands to detect zero movements (change of coordinates ; deadband)

13 Boresite Alignment.vi

This VI contains the driver control and is an example for the tracking software. It has two sections the boxcar integrator mode and the tracking mode

First one needs to setup the IP address for the connection

В	eam Position Boxcar Analysis	TCP/IP	extenal Control		
	IP Address		Port	Timeout (ms)	
	10.49.234.234		() 2055	5000	Reconnect

Run the VI using LabVIEW's run 🖄 button if the VI is not started by Dual Boresite Alignment.vi.

13.1 First start of Boresite Alignment.vi

During the initialization of *Boresight Alignment* the software checks some settings in the initialization file: if the following setting is present

```
[Drive]
Simulation = FALSE
ControlVI = "BoreDrive NoDrive Control.vi"
```

the software assumes that a different control VI should be used for the alignment operation or that the simulation mode should be switched on. Therfore, the following dialog comes up:

-	Select Bore Drive Control VI	\times						
	[Drive] ControlVI = "BoreDrive NoDrive Control.vi"							
	has been found in the initialization file BoreAnalysis.ini. This setting is used for simulation and testing.							
	Do you want to simulate Bore detector signals and drive movements? Then cancel this dialog (x), set Simulation = True in "BoreAnalysis.ini", Start the Virtual Controller and run Boresight Alignment again.							
	Do you want to test Bore detector signals but do not need active beam steering? Then actively reselect BoreDrive NoDrive Control.vi in the selection list below. Later when using active beam steering you will have to manually edit the initialization file. Confirm with <ok>.</ok>							
	Do you finally want Bore detector signals and active beam steering now? Then select one of the available drive Control VIs other than BoreDrive NoDrive Control.vi or browse for a custom drive control VI. Confirm with <ok></ok>							
	After following one of these options this dialog will not be shown again.							
	Please Select a Drive Control VI							
	ОК							

The user has 3 options now:

1. Simulate Bore detector signals and drive movements

in this case the dialog should be canceled (top right X), and the initialization file BoreAnalysis.ini should manually be changed before the next program start:

```
[Drive]
Simulation = TRUE
ControlVI = "BoreDrive NoDrive Control.vi"
```

To switch to non-simulation mode the user would have to edit the initialization fiel again and set Simulation = FALSE.

2. Test Bore detector signals but do not activate the beam steering

then,BoreDrive NoDrive Control.vi has to be re-selected in the selection list below.



, the dialog can then be confirmed by clicking OK.

The program will add UserSelection = 1 to the initialization file to avoid to display the start dialog again. Later, before using active beam steering, the initialization file has to be changed again (remove UserSelection = 1, then the dialog will be shown again or enter the correct name of the desired control VI).

3. Finally use the Bore detector signals for active beam steering

Then, a drive Control VI other than BoreDrive NoDrive Control.vi should be selected from the list above or the browse button can be used to select a custom drive control VI.

13.2 Operate Boresite Alignment.vi

If the IP Address and Port match the settings of the Boresite Controller and the network is correctly working

the TCPIP LED indicator should change:



Address and Port, you may modify them while the VI is running.

If you recognize that you are not connected to the Ethernet Controller you would like to use (because you are operating two Boresite Controllers), just

1. Enter the IP Address and Port of that controller you really would like to connect to and

Reconnect 2. press the button.

Also use the Reconnect button after changing the IP Address and/or Port while no connection is active. In Boxcar mode the signal and the background bins are shifted over the signal to give a impression of the signal and to help to make a decision where to place the signal and the background region. First one needs to turn the high voltage on, select trigger, set the discriminator level and after pressing the start button an acquisition will start. This assumes that a trigger is supplied.



Once these regions are known, press the start button. The movement of the center of gravity position should be visible.



The next steps assume that you filled the VI's described above with your routines to control the stepper driver.

1. if you are working with a controller where a homing may be required (like Newport SMC100, Newport Conex-CC) an additional button will appear:



You can start a home run manually or by using the TCP/IP API. Without the home run you should not be able to proceed.

- 2. the current drive positions should be shown
- 3. press Start to start the acquisition at the alignment detector
- 4. multiple curves can be shown by connecting the second motor and the Quantity is set to 2 in the initialization file. Trace of Channel A has the color white and from Channel B the Trace Color is orange.
- 5. assuming that you have enough signal the SNR is OK should become bright green. If it is not raise the number of shots to accumulate, increase the signal region or provide a better alignment.
- 6. with motor Control one can choose the motor to control.
- 7. once the SNR is good, you can define the current alignment as a reference point by pressing the target button. Therefore you have to select the mirror from Motor Control drop down menu before pressing the set target button. Green dot is the target point of mirror A and red dot is the one of mirror B.
- 8. If this is done the tracking is activated by pressing the Keep Alignment button.
- 9. watch closely if the system starts to drift away stop the tracking by pressing the Keep Alignment button again so that it comes out. Probably the mathematical relation ship that you entered at BoreDrive DeviationToXY.vi needs to be corrected.
- 10. press Stop to stop the acquisition at the alignment detector
- 11. press Clear to clear the graphics

13.3 Remote Control

Typically the support for stepper driver is excellent for MS Windows OS but is less good for other OS. In this case the remote control of the Boresite Alignment VI might help. It requires that a Windows PC controls the boresite detector and the steering mirror stepper drives and is connected via Ethernet to a master controller where any OS can run, Linux for instance.

The remote control opens two TCPIP ports which need to be defined in the BoreAnalysis.ini

[TCPIP_API] Active = TRUE Port = 2060

The application will then use 2060 and 2061 for communication. The available TCP/IP API commands are described below. Port 2060 is used for synchronous commands and responses while port 2061 mirrors the data from the bore sight detector and the motor commands.

14 Simulation

The whole Alignment Module can be also tested in Simulation mode. One needs a Virtual Controller. This is part of the executables in http://www.licel.com/download/ethernet/LVInstaller.zip. An executable for Windows also comes with the Bore LabVIEW sources, it needs the LabVIEW Runtime Engine 1016. It should start up as

General PMT Bore System	
Lidar computing and electronics	TCP/IP TCPIP Connection 1 TCPIP Connection 2
Capabilities	Operation ON Power

Please note the green Bore switch, indicating the support for the Bore commands. The general capabilities are described at in the Licel Ethernet Controller Installation and Reference Manual chapter 8. In addition to the there described capabilities is one tab with the Bore support. This also provides as stepper driver simulation coupled with a alignment computation. Only one stepper driver is supported.

General PMT Bore System				
Signal Regions		Signal Displacement	Picture	
Paderound Start	BORE	X		
	Interrupt	v		
Background Stop 0		0.00		
Signal Start 0	•			
Signal Stop 0	BoreMode	Background Displacement		
	. A	×		
Shots to Average 0		y		
Number of Cycles 0	Done 0	0.00		
Counter	BoreSigns	Signal		Background
7 0-1-1		topLeft	topRight	topLeft topRight
telescope field of view	w (mrad)	0.19 bottomLeft	0.31 bottomRight	0.19 0.31 bottomLeft bottomRight
beam divergence	o (mrad) 40.5	0.25	0.30	0.25 0.30
Background	Size (%)		Background	Signal [MHz]
Signal	Size (%)			Signal[MHz]
Signal	40.00			<i></i>
	Motor Simulation	on		

To run the simulation start first the Virtual Controller.exe. Set the StepperDriverSim variable to true inside BoreDrive InitializeDrive.vi. Enter 127.0.0.1 as the IP number in the Alignment Software and start it. Once it runs the TCPIP indicators at the entrance panel should become green. Change the BoreMode to get alternating signals or only from channel a or b.

Repeat then the steps described above in the section BoreSite Alignment.vi. Press the *Start* button and see the simulation coming into live.

15 Low Level TCPIP access

15.1 Commands

The syntax of the commands is described in the Licel Ethernet Controller Installation and Reference Manual. The bore controller implements the general commands:

- *IDN?
- CAP?
- TCPIP
- LOGON
- WHITELIST

The HV control is controlled via the PMT commands:

- PMTSTAT?
- PMTGAIN

DeviceNumber should be **0** for this as there is only one HV integrated into the head. The discriminator is controlled via

• DISCRIMINATOR

The following commands are unique for the controller:

- ALIGNDATA
- ALIGNSIGN
- ALIGNSIGNB
- ALIGNTIME

15.2 Align Sequence

The following steps need to be executed to run the controller

- 1. Open the command socket typical at 2055
- 2. Open the push socket one port above the command socket
- 3. Set the signal and the background region with the ALIGNTIME command
- 4. Set the discriminator DISCRIMINATOR command
- 5. Turn the PMT high voltage on with the PMTGAIN command
- 6. Activate the data transmission with the ALIGNDATA command
- 7. Read the data from the push socket
- 8. If necessary change the alignment
- 9. In order to avoid to react on data that was recorded before the alignment change send the ALIGNSIGN or ALIGNSIGNB command and wait till the counter sign in the received data changes.
- 10. Once the measurement is finished, stop the data with the ALIGNDATA command, turn the HV off with the PMTGAIN command.

The bore aligment vi implements this. If you plan to work with C the bore application in the bore_src directory of the C Driver shows how that could be implemented. The program assumes a Virtual controller running. The Virtual Controller also contains a stepper driver simulation that is coupled to the aligment simulation. The Virtual Controller.exe requires a working LabVIEW Runtime Engine 2010 SP1 (see http://licel.com/soft_tcp.html for the link).

15.3 LabVIEW TCPIP Bore VIs

LiceITCPIP BoreGetData.vi

Read data transmitted by the bore sight detector.

The data has the format Align Info:id sig1 sig2 sig3 sig4 bg1 bg2 bg3 bg4 for Trigger A or the format Align Info_B:id sig1 sig2 sig3 sig4 bg1 bg2 bg3 bg4 for Trigger B, where id is a counter which absolutely increases with each transmitted data set. The sign of the counter can be toggled via the ALIGNSIGN command. sig1..4 and bg1..4 are the averaged signal and background values and for the involved channels and corresponding to the background and signal regions defined by the ALIGNTIME command. Align Info_B marks that the incoming data is from channel B otherwise the data is from channel A. The information from which channel the data is coming is outputted by Channel



LiceITCPIP BoreSetRanges.vi

Configure the signal and background ranges at the bore sight detector using the ALIGNTIME command. The input cluster **Signal Regions** contains the elements:

Background StartStart bin of the background rangeBackground StopStop bin of the background rangeSignal StartStart bin of the signal rangeSignal StopStop bin of the signal rangeThe following restrictions apply:

- 0 < Background Start < 1024
- Background Start < Background Stop < 1024 + Background Start
- Background Stop < Signal Start < 1024 + Background Stop
- Signal Start < Signal Stop < 1024 +Signal Start

where Background Start and Background Stop may pairwise be interchanged with Signal Start and Signal Stop. A range bin corresponds to 15m.



LiceITCPIP BoreSign.vi

Changes the sign of the bore counter, The Channel input selects the signal whose sign of the counter id will be toggled



LiceITCPIP BoreStart.vi

Start the acquisition at the bore sight detector.

Shots is the number of shots to acquire every cycle

Cycle specifies the number of acquisition cycles. When the number is reached, the controller will stop the acquisition. A value of -1 (default) has the meaning of infinite cacles i.e. the acquisition will be active until it is explicitly stopped.



LiceITCPIP BoreStartMemory.vi

Start the acquisition at the bore sight detector for box car analysis. An additional input (Trigger) is added to control the input channel

Shots is the number of shots to acquire every cycle

Cycle specifies the number of acquisition cycles. When the number is reached, the controller will stop the acquisition. A value of -1 (default) has the meaning of infinite cacles i.e. the acquisition will be active until it is explicitly stopped.

Trigger select the channel from which the boxcar Analysis should be done.



LiceITCPIP BoreStop.vi

Stop the data acquisition at the bore sight detector

15.4 Drive Simulation commands

The Virtual Controller can simulate the behavior of a stepper drive in x and y direction. All simulation commands start with **STEPPERDRIVESIM** and are terminated by a CRLF.

15.4.1 EXIT

EXIT **return** EXIT

15.4.2 HOME

HOME <AXIS> return HOME executed

15.4.3 GETPOS

GETPOS <AXIS> return AXIS: %d POS: %d

15.4.4 BREAK

BREAK return BREAK executed

15.4.5 STATUS

STATUS <AXIS> return AXIS: %d STATUS: %d

15.4.6 GETSPEED

GETSPEED <AXIS> return AXIS: %d SPEED: %d

15.4.7 SETPOS

SETPOS <AXIS> <NEWPOS> return SETPOS executed

15.4.8 MOVETO

MOVETO <AXIS> <NEXTPOS> return MOVETO executed

15.4.9 SETSPEED

SETSPEED <AXIS> <NEWSPEED> return SETSPEED executed

15.5 BoreRemoteControl Syntax

15.5.1 TCP/IP API Command List

The following list contains the supported commands. The commands must be sent with an additional <CRLF> (0x0D0A) and the responses will end with a <CRLF>, as well.

• *IDN?

Parameters Description Reply	Return the controller information example for a simulation: Licel Virtual Controller Software (version 1.51.10) rev. 20.09.2023
CAP?	
Parameters	
Description Reply	Return the Boresite Alignment software capability BoreRemoteControl
PMTSTAT?	
Parameters Description Reply	Return information about the high voltage PMT 800.0 off remote
PMTGAIN	
Parameters Description	0 <hv> Set the high voltage to HV V, e.g. PMTGAIN 0 850. HV = 0 will switch off the high voltage</hv>
Reply	PMTG executed
DISCR	
Parameters Description	<discriminatorlevel> Set the discriminator level to the given value. Allowed values must lie in the range 0 63</discriminatorlevel>
Reply	DISCR executed

• MINSNR

Parameters	<pre><min_snr></min_snr></pre>
Description	noise ratio), e.g. MINSNR 20
Reply	MINSNR executed

• SHOTSAVG

Parameters	<shotstosverage></shotstosverage>				
Description	Set the number of shots to average, e				
	SHOTSAVG 10				
Reply	SHOTSAVG executed				





-Motor Alignment			Discriminator
Current Position	Current Position B		
х ү	Х Ү	Reset	
0.00 0.00	0.00 0.00		Charles Annual Ale
Target Position	Target Position B	Set Target	Shots to Average gr
<u>x y</u>	Х Ү	Motor Control	150.00-
0.00 0.00	0.00 0.00	Mirror A	
Motor Coordinates	Motor Coordinates B		125.00 -
Х Ү	Х Ү		100.00
0 0	0 0		
	,		/5.00-
Emergency	0.00 Min SNR		50.00 -
Brake		V	25.00
Keep Align	ment 🕘 SNR is OK		23.00
			0.00-
Start	Stop Clear		



Motor Alignment Current Position B					Discriminator	
x 0.00	Y 0.00	x 0.00	Y 0.00	Reset	days a second second	
Target Pos	sition	Target P	osition B	Set Target	Shots to Average gr	
0.00	0.00	x 0.00	0.00	Motor Control	150.00	
Motor Coo	rdinates	Motor Co	ordinates B		125.00	
X	Y	X	Y lo		100.00	
					75.00	
	imergency Brake	0.00	Min SNR		50.00	
Keep Alignment SNR is OK					25.00	
Keep Augument					0.00-	
Start	S	top	Clear			



• START

Parameters	
Description	Start the acquisition at the Boresite con-
	troller (emulate a click on the <i>Start</i> button)
Reply	START executed

• STOP

Parameters	
Description	Stop the acquisition at the Boresite con- troller (emulate a click on the <i>Stop</i> button)
Reply	STOP executed

• TARGETA

Parameters <targeta_x> <targeta_y></targeta_y></targeta_x>					>		
Description	ription Set the target values Target						
	TargetA_Y for the system A, e.g. TARGETA						
	0.1 0.11						
Reply	TARGETA executed						

• TARGETB

Parameters <targetb_x> <targetb_y></targetb_y></targetb_x>					>
Description	Set	the	target	values	TargetB_X
	Tarc	getB_Y	for the s	ystem B, e	e.g. TARGETB
	0.08	8 0.12	2		
Reply	TARG	GETB (execute	d	

• MOTORPOS?

• SIGNALREG

Parameters

Description

Parameters Description	<a b> Request the motor positions (in motor coor- dinates system A (red) or B (blue)</a b>
Reply	MOTORPOS x y where x and y are the cor-
	responding motor coordinates



<backgroundstart></backgroundstart>	Signal Regions (km
<backgroundstop> <signalstart> <signalstop></signalstop></signalstart></backgroundstop>	Background Start
Set the background and signal ranges, e.g. background 20.2 27.5 km, signal 2.5	Background Stop
15.2 km , i.e. SIGNALREG 20.2 27.5 2.5 15.2	Signal Start
SIGNALREG executed	Signal Stop 🖌 15.200

• ALIGNMENT

Reply

Parameters	<0 1>
Description	Start (1) or stop (0) the alignment
Reply	ALIGNMENT executed

s to Average 🚽 1
150.00-
125.00
100.00
75.00
50.00
-

• DRIVESTATUS?

Parameters Description

Request the drive status from the used controller. This includes the information about whether or not the control VI is initialized, a home run is completed, and/or a home run is active

example: DRIVESTATUS INITIALIZED=1 HOMEOK=0 HOMERUN=0 indicates that the Control VI is initialized, that a home run is required, and that a home run is currently not active

• GOHOME

Reply

Parameters Description Reply	Will start a home run if this is supported by the Control VI and the used controller ${\tt GOHOME}\xspace$ started
MOVEABS	
Parameters Description Reply	<a b><x y> <position> Let the axis X or Y at the system A or B move to the given Position MOVEABS started</position></x y></a b>
VER?	
Parameters Description Reply	This command will return the software version of the Boresite Alignment software Boresite Alignment (version 2.01.10/TCPIP 3.00.08)

15.5.2 BoreRemoteControl Sample Script Perl

```
use IO::Socket;
use File::Basename;
use feature "switch";
use strict;
```

```
use warnings;
my $dummy;
my $oldIP = $ARGV[0];
my $resp;
# _____
                 _____
# tcp/ip send
# _____
                     _____
sub sendCommand {
 my ($socket, $command) = @_;
 print $command . "\r\n";
 print $socket $command ."\r\n";
 $dummy = <$socket>;
 print $dummy;
 $dummy;
}
# --
                        _____
# go
# ______
 print "Connecting to: " . $oldIP . "\n";
 my $sock = new IO::Socket::INET (
 PeerAddr => $oldIP,
 PeerPort => '2060',
 Proto => 'tcp', );
 die "Could not create socket: $!" unless $sock;
 my $push = new IO::Socket::INET (
 PeerAddr => $oldIP,
 PeerPort => '2061',
 Proto => 'tcp', );
 die "Could not create socket: $!" unless $push;
 $resp = sendCommand($sock, "*IDN?");
 $resp = sendCommand($sock, "CAP?");
 $resp = sendCommand($sock, "PMTSTAT? 0");
 $resp = sendCommand($sock, "PMTGAIN 0 850");
 $resp = sendCommand($sock, "PMTSTAT? 0");
 $resp = sendCommand($sock, "DISCR 8");
 $resp = sendCommand($sock, "MINSNR 20");
 $resp = sendCommand($sock, "SHOTSAVG 10");
 $resp = sendCommand($sock, "TARGETA 0.1 0.1");
 $resp = sendCommand($sock, "TARGETB 0.08 0.12");
 $resp = sendCommand($sock, "SIGNALREG 20.2 27.5 2.5 15.2");
 $resp = sendCommand($sock, "START");
 my $i;
 $dummy = "";
 for ($i = 0; $i < 5; $i++) {
   $dummy = <$push>;
   print $dummy;
 }
 $resp = sendCommand($sock, "ALIGNMENT 1");
 for ($i = 0; $i < 1000; $i++) {
   $dummy = <$push>;
   print $dummy;
  }
 $resp = sendCommand($sock, "ALIGNMENT 0");
```

```
$resp = sendCommand($sock, "STOP");
close($push);
close($sock);
```

15.6 BoreRemoteControl Sample Log

The following section shows a run a of the BoreRemoteControl.pl against a Boresite Alignment.vi which is connected to a Virtual Controller and uses the Stepper Drive Simulation as a steering mirror driver. The Boresite Alignment.vi runs on 192.168.178.28. The following actions are executed

- 1. Connect to the VI on Port 2060 and 2061
- 2. Ask the ID so that we are sure that we control the right VI.
- 3. Ask if is really a bore system
- 4. Get the current HV
- 5. Turn the HV on and set it to 850V
- 6. Get the current HV
- 7. Set the discriminator level of the Boresigh detector
- 8. Define the minimum SNR used in the alignment procedure
- 9. Set the number of shots to average
- 10. Set the desired center of gravity for Mirror A
- 11. Set the desired center of gravity for Mirror B
- 12. Set the background region to 20.2 27.5km and the signal region to 2.5 15.2km.
- 13. Start the data acquisition
- 14. log 5 datasets
- 15. start the alignment

16. Log the data and the drive commands and responses.

```
D:\Licel\CProjekte\BoreRemoteControl>perl BoreRemoteControl.pl 192.168.178.28
Connecting to: 192.168.178.28
*TDN?
Boresite Alignment (version 2.0.2/TCPIP 2.59.02 rev. 1167)
CAP?
BoreRemoteControl
PMTSTAT? 0
PMT 950.0 off remote
PMTGAIN 0 850
PMTG executed
PMTSTAT? 0
PMT 850.0 on remote
DISCR 8
DISCR executed
MINSNR 20
MINSNR executed
SHOTSAVG 10
SHOTSAVG executed
TARGETA 0.1 0.1
TARGETA executed
TARGETB 0.08 0.12
TARGETB executed
SIGNALREG 20.2 27.5 2.5 15.2
SIGNALREG executed
```

START START executed Align Info:0 5.004947 4.999078 5.025416 5.001313 0.000000 0.000000 0.000000 0.000000 Align Info:-1 4.980025 5.042612 5.012152 4.989077 0.000000 0.000000 0.000000 0.000000 Align Info:-2 4.952141 5.045466 5.045222 5.038550 0.000000 0.000000 0.000000 0.000000 Align Info:-3 4.985275 4.949838 4.960628 5.050857 0.000000 0.000000 0.000000 0.000000 Align Info:-4 5.047578 4.951504 5.045131 4.997110 0.000000 0.000000 0.000000 0.000000 ALIGNMENT 1 ALIGNMENT executed Align Info:-5 5.030637 4.974953 4.950946 5.040252 0.000000 0.000000 0.000000 0.000000 STEPPERDRIVESIM GETPOS 0 STEPPERDRIVESIM GETPOS 1 AXIS: 0 POS: 50000 AXIS: 1 POS: 50000 STEPPERDRIVESIM MOVETO 0 50010 MOVETO executed STEPPERDRIVESIM MOVETO 1 50010 MOVETO executed STEPPERDRIVESIM GETPOS 0 STEPPERDRIVESIM GETPOS 1 AXIS: 0 POS: 50010 AXIS: 1 POS: 50010 Align Info:6 5.119856 4.963286 5.050348 4.864851 0.000000 0.000000 0.000000 0.000000