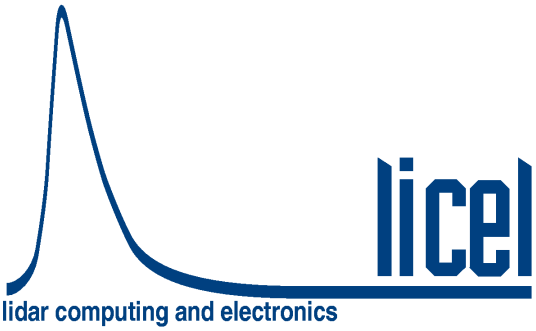


Licel Ethernet Controller – Installation and Reference Manual

Licel GmbH

December 5, 2019

Contents

Chapter 1

Introduction

The ethernet-based control modules for Licel detection systems open the path for truly remote controlled experiments. The Licel Ethernet Control Modules use a TCP/IP based protocol with a syntax similar to conventional GPIB based instruments. Each module use an ASCII command set with a structure similar to SCPI in order to be compatible with common measurement devices. A LabVIEW driver library for easy integration is supplied. The modules can either be operated using a static IP address or a dynamically assigned IP address (DHCP).
In the next chapter the control modules currently available at Licel are introduced. Then the installation of the software and setup of the network is described. The following chapter gives an introduction to the acquisition software. After that acquisition tutorial further software control modules are described. Finally the appendix contains information about the TCP/IP command set, the file format, initialization files, and the LabVIEW TCP/IP driver library.
The most up to date version of this manual can be found at http://www.licel.com/software.htm.

Chapter 2

Licel Control Modules

Currently several modules are available to control transient recorders, APDs, PMTs, the trigger timing/gating and the recording of laser power. Two integrated packages to make polarization sensitive measurements and to remotely control the detector alignment extend the family of Ethernet controllers. The control modules are equipped with a twisted pair ethernet connector which allows for 10/100 Mbit network based access.

2.1 The Transient Recorder Control Module

The Licel transient recorder control module can control up to 16 transient recorders. It translates the ASCII based commands received via TCP/IP into low level transient recorder commands. The data from the transient recorders is then sent back to the PC. This eliminates the need of a special interface card to control the transient recorder. The typical data transfer rate is 200 kb/sec. This is lower than for a PCI-DIO-32HS but offers a cost sensitive solution for small systems.
The transient recorder ethernet control module introduces a new data transfer mode: the push mode. In the push mode the transient recorders get their start, stop, and readout commands from the ethernet controller without any direct interaction with the PC. The ethernet controller then pushes the data to the PC. At the PC level, a periodic task reads the data when it becomes available from the TCP/IP buffer. This frees the PC from controlling the transient recorders by itself and reduces the communication load. The advantages of the push mode are important for single shot acquisitions.

2.2 The Photomultiplier High Voltage Control Module

The Licel photomultiplier high voltage control module can control up to 8 PMT modules. The control voltage ranges between 0 and 1V, which is generated by a precision DAC and monitored. PMTs with an activated high voltage are indicated by a LED.

2.3 The APD High Voltage Control Module

The Licel APD high voltage control module can control up to 4 APD modules. The control voltage ranges between 0 and 1.8V, (which corresponds to 0 to 450V APD HV), which is generated by a precision DAC and monitored. APDs with an activated high voltage are indicated by a LED (1-4). The thermoelectrical cooler can also be remotely activated. Once a stable temperature is reached the T_{Set} LED is activated.

2.4 The Licel Trigger Module

The Licel Trigger Module incorporates one trigger input and 4 different outputs to build up compact detection systems. The trigger input can be used to synchronize the system to an external laser flash lamp or Q-switch trigger. The module can also run internally triggered. The Licel Trigger Module consists of a timing sub-board which is able to generate:

- a lamp trigger
- a pretrigger for the transient recorder
- a Q-Switch trigger
- and a Gating trigger for gated PMT-Modules.

All timings are derived from a quartz based oscillator ensuring nanosecond timing stability.
Optionally the Licel Trigger Module can be extended with more timing sub-boards. Then, each of the sub-boards is able to generate the trigger outputs listed above. When programming such a multi-board Trigger Module the parameter `BoardID` in the TRIGGERMODE and TRIGGERTIME must be used.

2.5 The Licel Power Meter Controller

Monitoring every laser shot is the optimum approach to detect laser pulse energy fluctuations, SHG and THG efficiency changes and flashlamp degradation of your laser . A laser spot reflection can be measured using a photodiode or a laser power meter head. The photo diode signal or the output of the power meter head is directly analyzed at the Power Meter Controller.
After an external trigger is received a short trace from detector will be analyzed. The light pulse is measured and the result together with a time stamp is sent to the PC over a Ethernet connection. A trace mode is available to inspect a single pulse.

2.6 The Licel Polarotor

The Licel Polarotor adds (de-)polarization measurements to multispectral detection systems. A rotating stepper motor driven Glan Thompson prism is used to separate p- and s-polarized signal contributions. The integrated polarotor trigger generator synchronizes

- a lamp trigger
- a Q-Switch trigger
- a pretrigger for the transient recorder at the s polarization detection chain
- and a pretrigger for the transient recorder at the p polarization detection chain.

A laser repetition rate of up to 50 Hz is supported. All timings are derived from a quartz based oscillator ensuring nanosecond timing stability.
Like the Trigger Module the Licel Polarotor can optionally be extended with additional timing sub-boards. When programming those sub-boards the `BoardID` in the TRIGGERMODE and TRIGGERTIME must be used.

2.7 The Licel Bore Sight Alignment Controller

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.

Chapter 3

Software Installation

Licel provides a package of software modules for setting up the Licel Ethernet Controller for network operation, and for operating the Licel Control Modules. These software modules are written in LabVIEW's G language. The software is provided as LabVIEW source for users who have LabVIEW (beginning with version 7.0) installed, or alternatively as a set of Windows applications. The Windows applications come within a Windows Installer package for an easy installation on your Windows (95/98|NT|2000|XP|Vista) computer. Licel provides the software on a CD ROM and for download (http://www.licel.com/software.htm).

3.1 Preparation

Windows Application Users

If you have used older versions of Licel Windows applications it is recommended to backup existing initialization files (`*.ini`).
Search the existing installation directory of the older version of Licel Windows applications (standard: `<Program Files Directory> \Licel`) and backup all files with the ending `*.ini` to an archive file (zip, ARJ, TAR, etc...) or onto a CD ROM.

LabVIEW Users

If you have used older versions of Licel LabVIEW libraries it is necessary to remove and backup older versions.

1. Backup all your current Licel software libraries, in case you want to restore them, by either compressing them (zip, ARJ, TAR, etc...) or burning them onto a CD ROM.
2. Scan your disks to find all versions of the following files and delete them once you have made backups of them

```
Advanced Viewer.llb
ControlAPD-PMT.llb
ControlTiming.llb
Datafile.llb
Licel Acquis.llb
Licel Module.llb
Licel TCPIP.llb
Licel Util.llb
Postan.llb
Power Meter.llb
```

```
Search Controllers.llb
TCPIP-Acquis.llb
M-Acquis.llb
TCPIP-LiveDisplay.llb
TCPIP-MPush-Acquis.llb
TCPIP-Pulse.llb
TCPIP-Track.llb
Licel Main.vi
Licel Main-M.vi
Multi Power Meter Control.vi
```

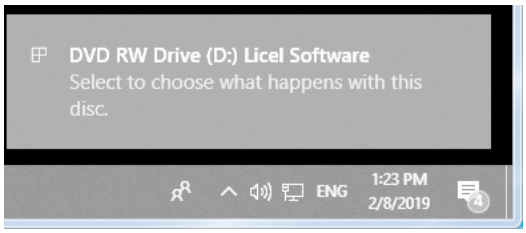
Please note: Licel may have provided individual software solutions with additional or less LabVIEW library files than noted in the list above.

3. Search the directory your older version of Licel LabVIEW libraries reside and backup all initialization files (`*.ini`).
4. The LabVIEW sources are delivered including the following files and directories:
   - `Installation.txt` a short description file
   - `source` a directory containing the above listed LaVIEW LLBs and VIs and initialization files
   - `project` a directory containing the LabVIEW project `LicelTCPIP_src.lvproj`
   - `Files\user.lib\errors\Licel-errors.txt` Licel error code file

3.2 The Licel CD ROM

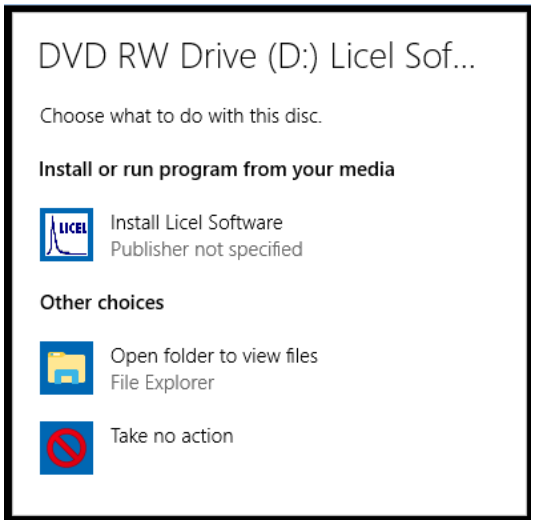
The standard CD ROM provided by Licel contains both, the LabVIEW sources and the Windows Installer for installing the Windows applications, and furthermore a documentation folder. Licel may add customer specific components on the CD ROM.

1. Insert the Licel CD into your CD ROM drive.
2. In Windows 10 you will normally be notified by a pop-up message at the bottom right corner of the main monitor.



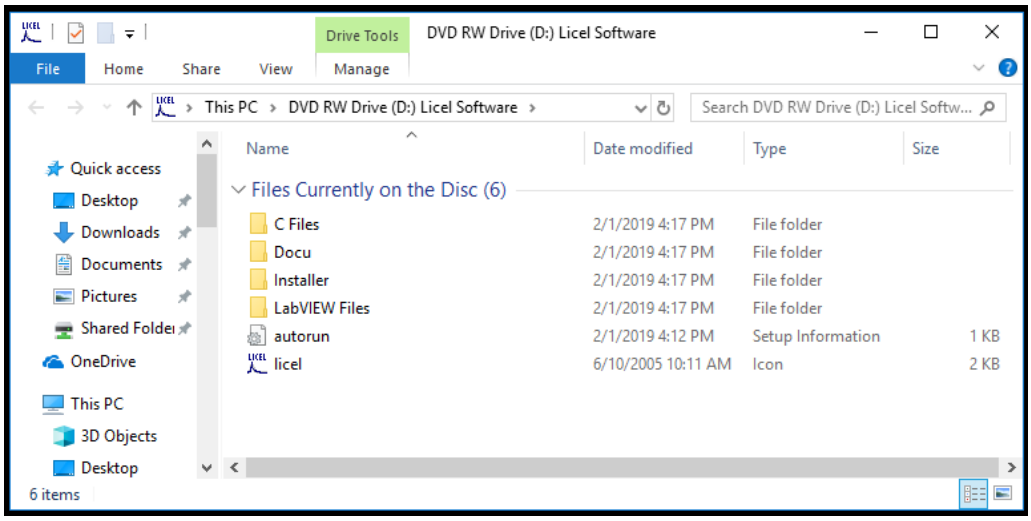
Please click on the pop-up message.

3. The following selection dialog should appear:

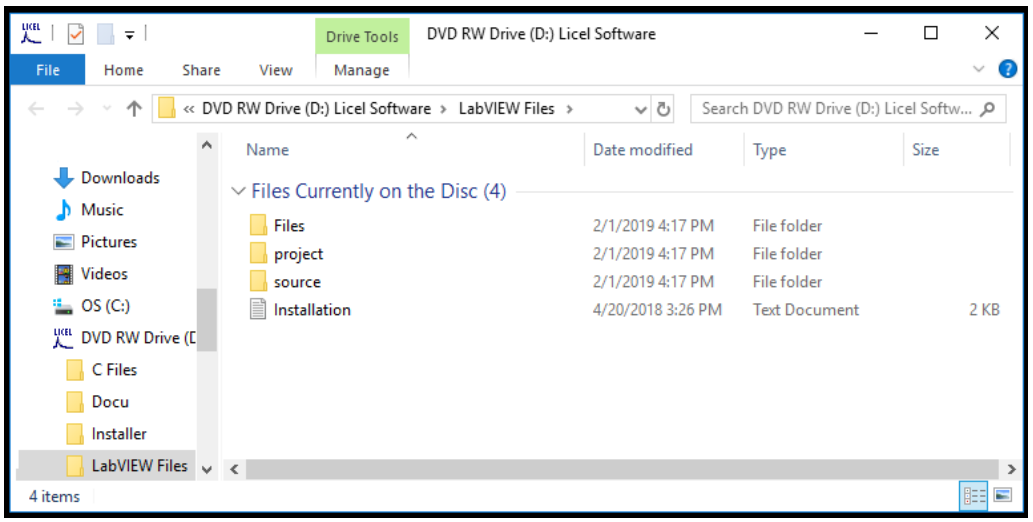


In older Windows operation systems a similar dialog will automatically come up.

- Press *Install Licel Software* to start the Windows Installer which will guide you through the installation of the Licel Applications. Please proceed to the section 3.4.
- Press *Open folder to view files* to start the File Explorer (Windows Explorer) to see the content of the CD:



  - The LabVIEW source files are located in the folder *LabVIEW files*. From there you may copy them to a directory of your choice on your local PC.

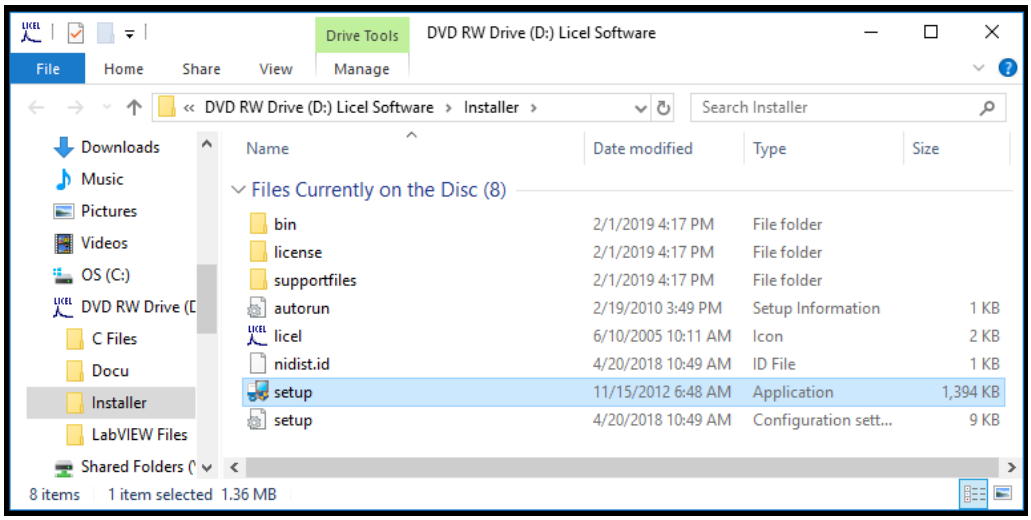


Please note the remarks according to existing LabVIEW library files. Please refer to the section 3.5 for further details.

– In the folder *Docu* you will find some documentation.
– The folder *C Files* contains Licel's C sources.

4. If the selection dialog does not come up automatically after inserting the CD into your CD/DVD drive, please manually open the File Explorer (Windows Explorer) and navigate to the CD/DVD drive of your PC.

   • Either go to the folders *LabVIEW Files*, *Docu*, or *C Files* to get the LabVIEW source files, read the documentation, or copy the C source files,
   • or open the folder *Installer* and run *setup.exe* by double click to start the Windows Installer.



Please proceed to the section 3.4 afterwards.

3.3 Download

The Licel software is frequently maintained. The most recent version is available on the download page (http://www.licel.com/software.htm). Licel provides both packages described in this chapter, the LabVIEW sources as well as the Windows installer to deploy the Windows applications. The packages come as zipped archive files, `TREthernet.zip` contains the LabVIEW sources, while `LVInstaller.zip` is the corresponding zip archive with the Windows installer. Note that you may have changed these files names while downloading the archives.

Unpacking the Windows Installer

If you downloaded the Windows Installer package (`LVInstaller.zip`) please unzip all files to a temporary directory. Locate the setup routine `setup.exe` in that directory and run it by double-clicking the program entry in the Windows Explorer. Please proceed to the section 3.4.

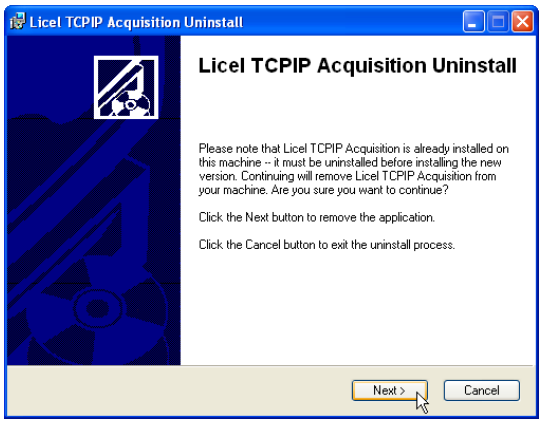
Unpacking the LabVIEW Sources

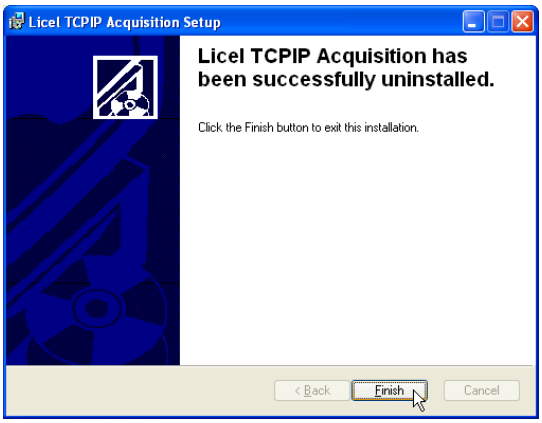
The Licel LabVIEW libraries and initialization files contained in the zip file `TREthernet.zip` may directly be unzipped to a destination folder of your choice. Please note the remarks according to existing LabVIEW library files. Please refer to the section 3.5 for further details.

3.4 Installing the Windows Applications

This subsection describes the installation process for the Licel Windows applications. To operate the Licel Windows applications a LabVIEW runtime environment needs to be installed, as well. The Windows applications together with the LabVIEW runtime environment come as a Windows Installer package. For the installation of the LabVIEW runtime part of the installer package local administrator privileges are required.
The following items describe the installation process after starting the Windows Installer's setup routine (`setup.exe`). The setup program is automatically started when using the CD ROM and pressing **Install Applications** in the setup selection dialog. `setup.exe` is located on the Licel CD ROM in the subdirectory `Installer` or in the temporary directory you unzipped the downloaded Licel Installer package. You may directly start the setup routine from the corresponding directories.
The Windows Installer dialogs will guide you through the installation process.

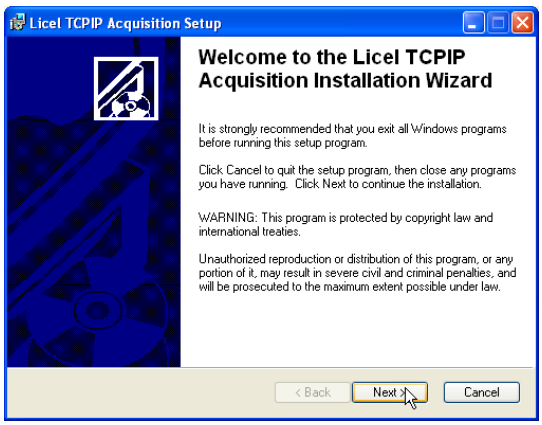
1. If an older version of the Licel software is detected, the install utility will first remove the old components.

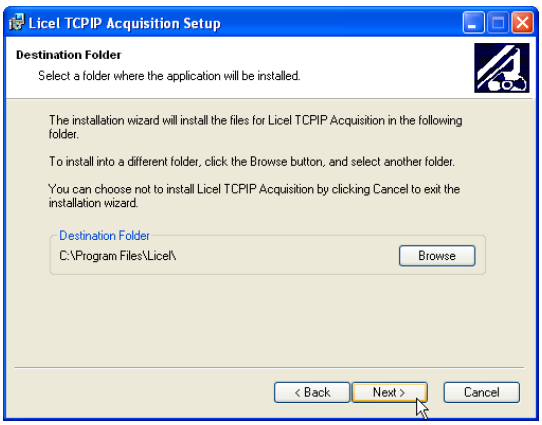




   Click **Next**, and after the uninstall process has terminated **Finish**. Then, you will have to run `setup.exe` again.

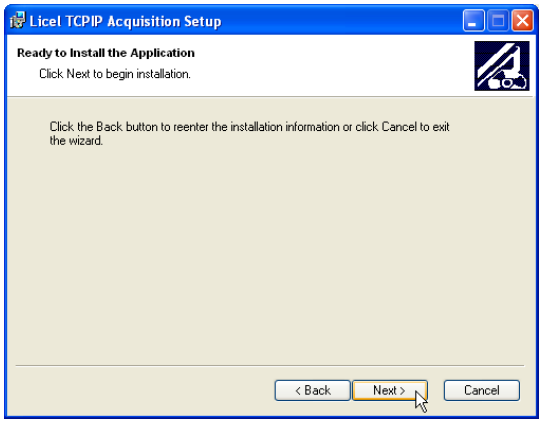
2. If no older version is detected a welcome screen will appear. Please click **Next** to proceed. On the next screen you may choose the installation directory (standard: `<Program Files Directory> \Licel`).

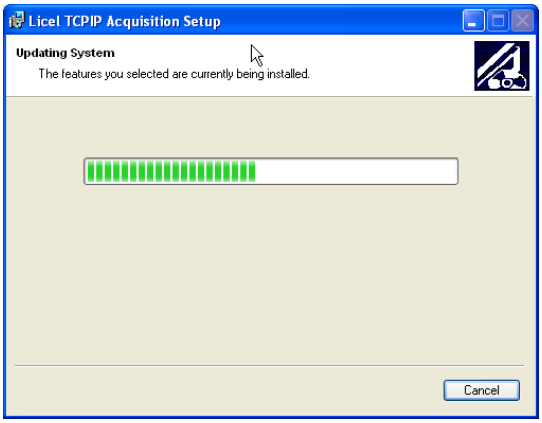




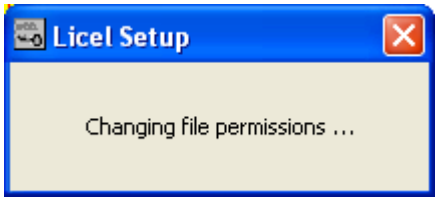
If you would like to change the installation directory click **Browse** and choose or create a directory of your choice. Click **Next** to proceed.

3. Confirm the next dialog by clicking **Next** or click **Back** to change your installation settings. After starting the installation progress is indicated by a progress bar.

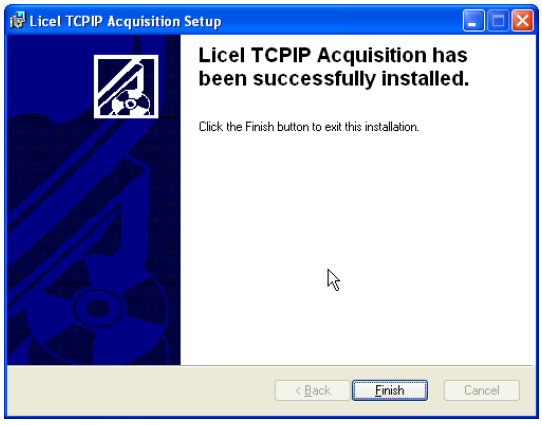




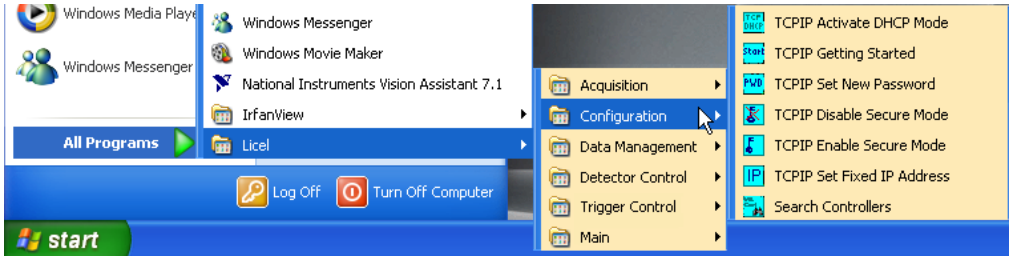
At the end of the installation process permissions on some files will be set.



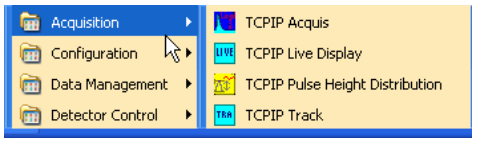
4. A successful installation will be shown in the next screen. Please click **Finish** to proceed.

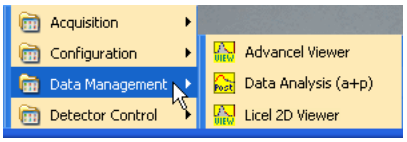


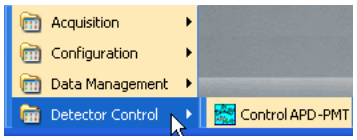
5. After the installation has successfully been completed you are able to start the Windows applications through the corresponding entry in the program group **Licel** in the Windows start menu:

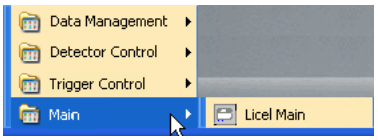


The links to the applications are grouped into the subfolders **Configuration**, **Acquisition**, **Data Management**, **Detector Control**, **Trigger Control**, and **Module Control**. and **Detector Control**.



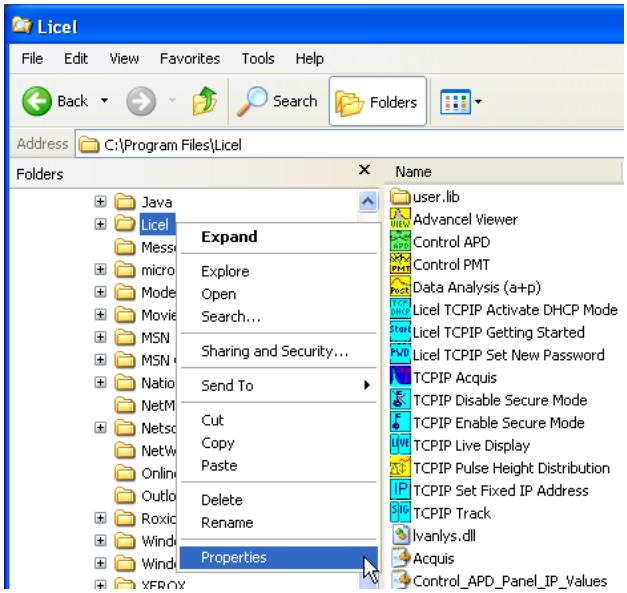


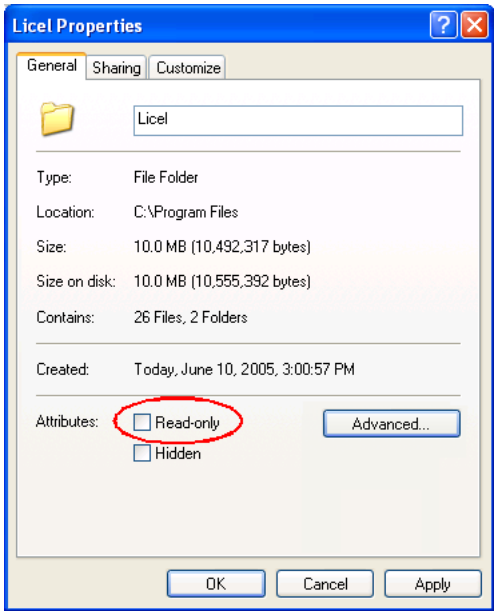




The standard software package includes the Licel Virtual Controller: the following start menu item will be available as well:

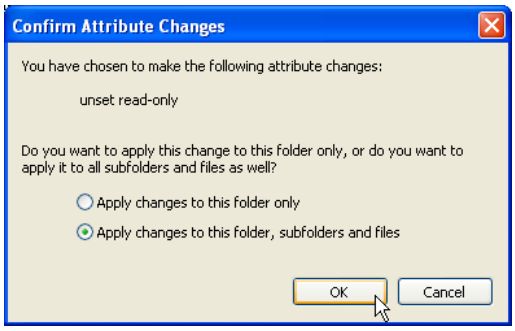
6. Please note that the Licel software needs write permissions for the initialization files located in the installation directory. Normally sufficient permissions are set during the installation process. If any problems indicating missing permissions occur select the directory and right-click on it. Select **Properties** from the context menu.





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*



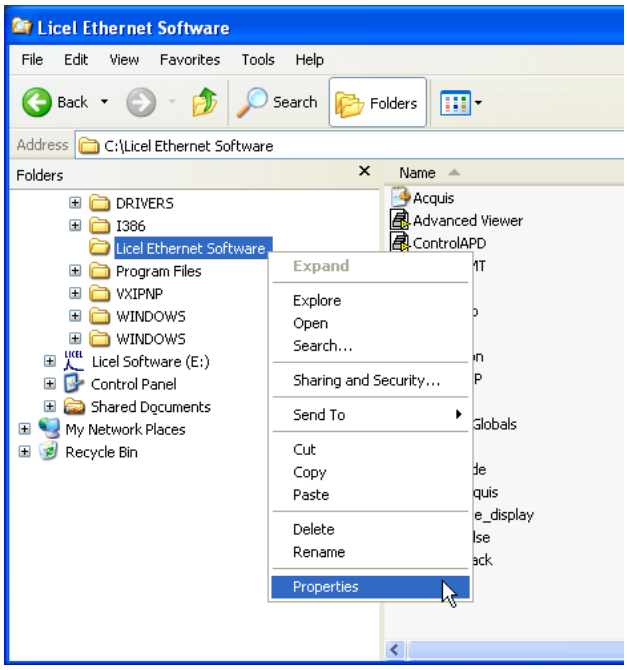
7. If you have backuped your initialization files from an older version of Licel Ethernet Software you may copy the TCP/IP parameters from the corresponding old initialization files to the files of the current installation. Please note that copying information from older to new initialization files should be done value by value (line by line).

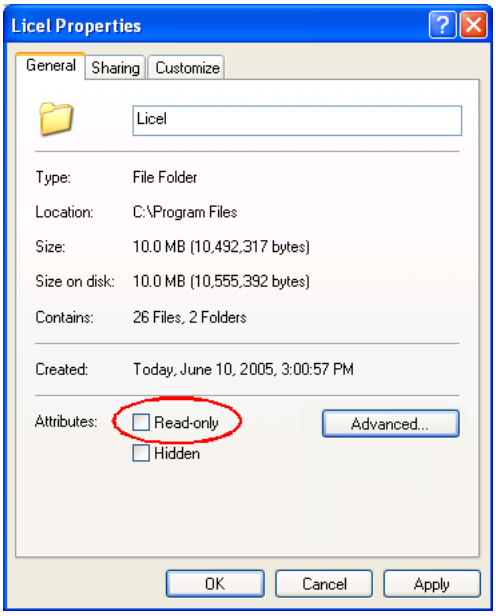
3.5 Installing the Licel LabVIEW Libraries

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

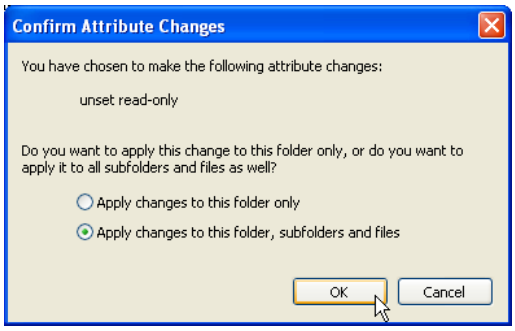
- The Licel LabVIEW Libraries will be copied automatically from the Licel CD ROM by pressing **Copy LabVIEW Source** in the setup selection dialog. You will be asked to select or create a target folder.
- You may manually copy all files contained in the directory `LabVIEW Files` on the CD ROM to a directory of your choice.
- If you downloaded the Licel software from http://www.licel.com/software.htm please unpack the content from the downloaded zip file and copy it to a directory of your choice.

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



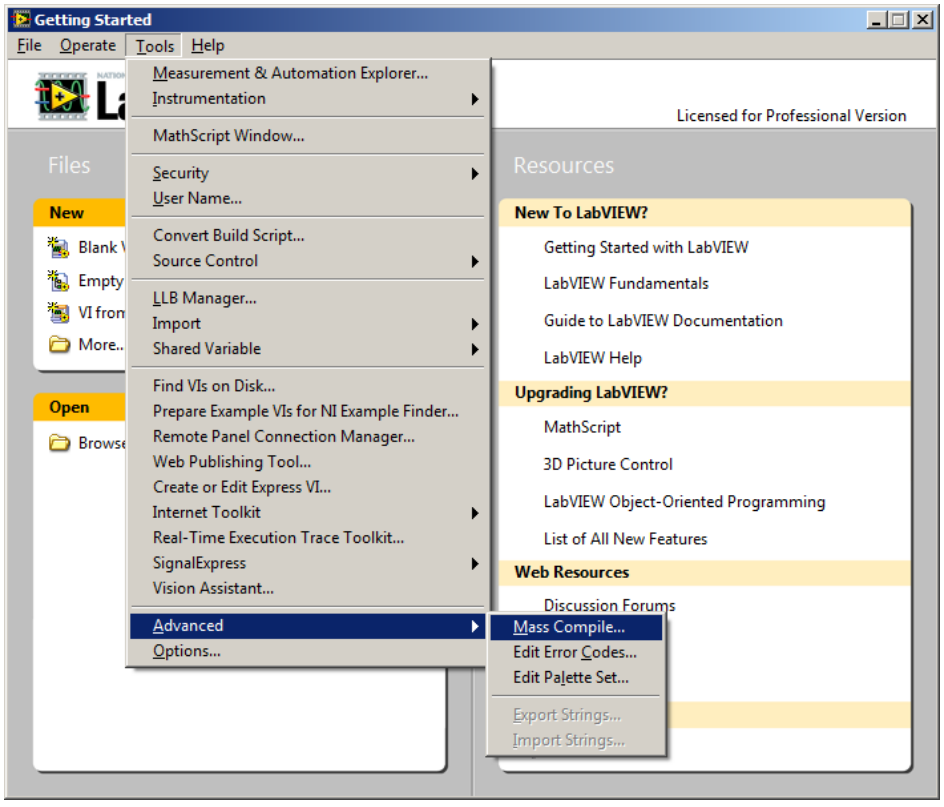


Verify that the "`Read-only`" attribute is not checked, uncheck it if necessary. Click *OK* and check in the next dialog *Apply changes to this folder, subfolders and files*. Leave the dialog by clicking *OK*

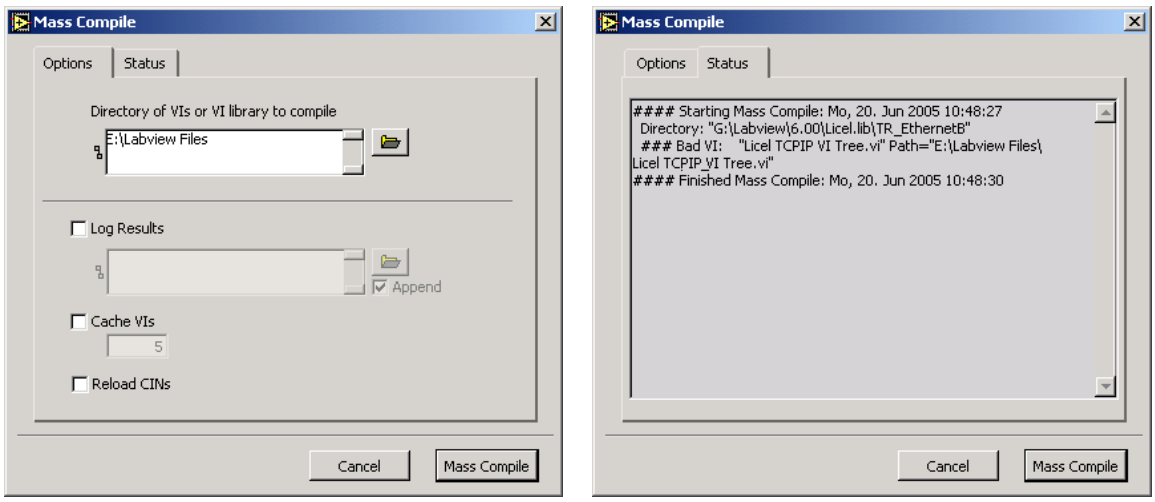


Now you should be able to run all the files. If you are still having problems, apply a mass compile to the directory where the software was extracted to:

1. Start LabVIEW.
2. Select the menu **Tools**, then **Advanced**, and finally **Mass Compile...**.



3. You will be asked to select a directory, select the target directory of the LabVIEW source files.
4. Press *Mass Compile* in the next dialog.



5. Later the mass compile status will be shown. Please ignore that the vi `Licel TCPIP VI Tree` is indicated to be a ”bad vi”.

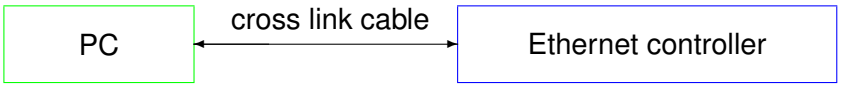
Please note that the removal of older libraries is a necessity, since LabVIEW often links to various libraries with the same name. As a result, if a library is installed twice, one can not be certain which library is actually being used.
If you still have any problems, please contact Licel for further assistance.

Chapter 4

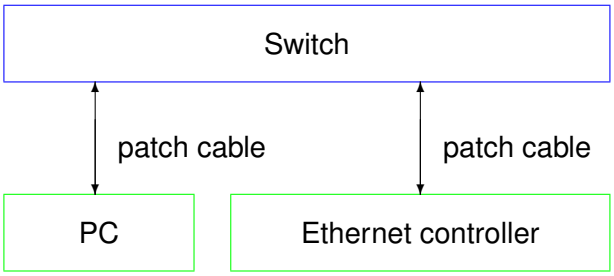
Setting up the Network

4.1 Network Introduction

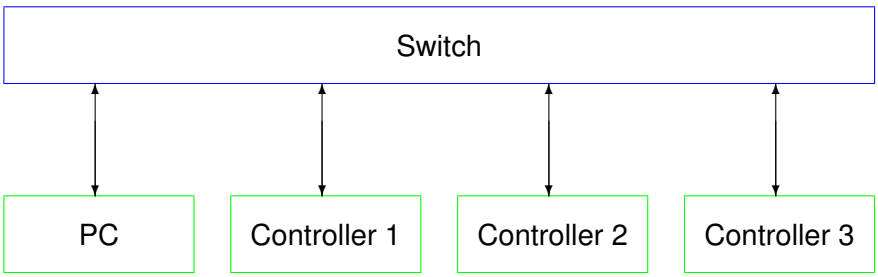
To control a Licel Ethernet controller a working TCPIP connection is required. This can be reached by two ways, using a cross link cable, which creates a one to one connection between the PC and the Ethernet Controller or with patch cables and a switch



The cross link cable might be a perfect setup for single controller, but as soon as the PC needs to communicate over the same network connector with other nodes locally or the Internet the usage of a switch is mandatory.



This configuration has the big advantage that it is easily scalable if more than one controller needs to be connected.



There are two concepts for the switch either:

- Use the local infrastructure, this requires coordination with your local network administrator as

she/he will define network addresses to be used for the PC and the Ethernet controllers or require DHCP for the nodes to be used.

- add a second Ethernet controller to the PC, so that Ethernet controllers can be moved to a private network and you become the administrator of this private network.

  http://en.wikipedia.org/wiki/Private_network describes the available address ranges, selecting a network subset in the 192.168.0.0 192.168.255.255. seems like a good choice

In all of these configurations the PC and the controllers should be finally in the same subnet but have different IP addresses within this subnet. To achieve this, each controller needs to be specially setup as all controller ship with the same default network address. If more than controller needs be setup the procedure below needs to be repeated for each controller individually. **Never** connect more than one controller with the factory default to a network.

4.2 Preparations

To operate the Licel Ethernet Controller in your local network you will have to carry out the following required steps described in the corresponding subsections:

1. Get the required **Network Information**.
2. Prepare the PC to communicate with the Ethernet controller using a cross-link cable (**Network Preparation**).
3. Setup the Ethernet controller for your local area network either by setting a fixed IP address or by activating the DHCP mode (**Network Setup**).
4. **Reconfigure the PC** for your local area network and test the communication with the Ethernet controller.

4.3 Network Information

The Licel Ethernet Controller is shipped with a default static IP address. The default parameters are:

IP address	10.49.234.234
network mask	255.255.255.0
gateway	
port	2055

The network parameters should be aligned according to your local network environment. Before doing this, the system administrator should be contacted. He should provide the following information:

1. Should the Ethernet controller use a dynamically assigned IP address (DHCP)?
   (a) If yes, the network parameters will be set by a DHCP server residing in your LAN. Refer to the subsection DHCP Mode (4.5.2) to enable the Licel Ethernet Controller to automatically receive the network parameters from the DHCP server.
   (b) If a static address configuration is to be used,
      i. the IP address,
      ii. the network mask,
      iii. and the gateway

      should be set by yourself. Refer to the subsection Fixed IP Address (4.5.1).
2. The default ports used by the Ethernet controller are 2055 and 2056. Can these ports be used?
3. Is it necessary to change the configuration of any firewall in the case you need to access the controller outside of the LAN boundaries?

4.4 Network Preparation

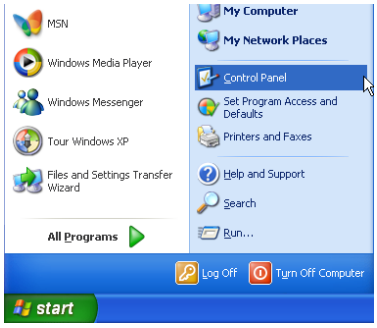
After having installed the Licel Windows applications or the Licel LabVIEW modules on your PC you are ready to change the network configuration parameters of the Licel Ethernet Controller according to the local network settings described in the previous section.

4.4.1 Establish the Connection

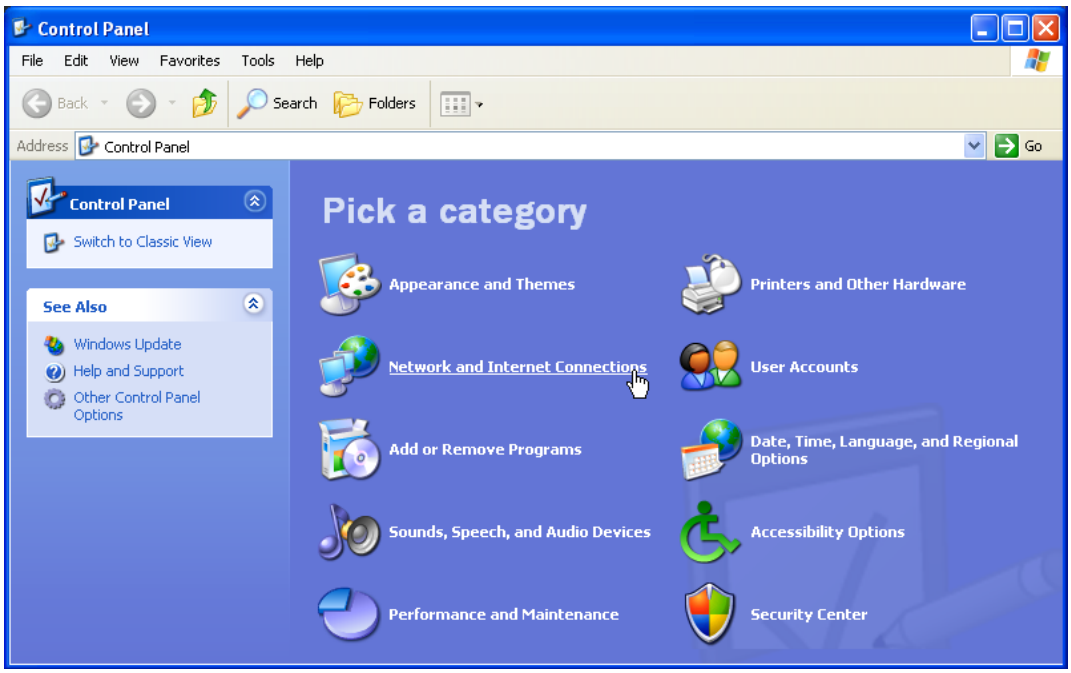
A straight-forward way to do this is the following procedure. You will need local administrator rights on your PC for the following steps:

1. Disconnect the PC from the local network.
2. Open the **Properties** dialog of the network connection your Ethernet adapter is assigned to. Usually you will find the appropriate network connection by opening **Network Connections** from the Windows start menu or the System Settings. The following list shows the steps to follow on a Windows XP operating system:

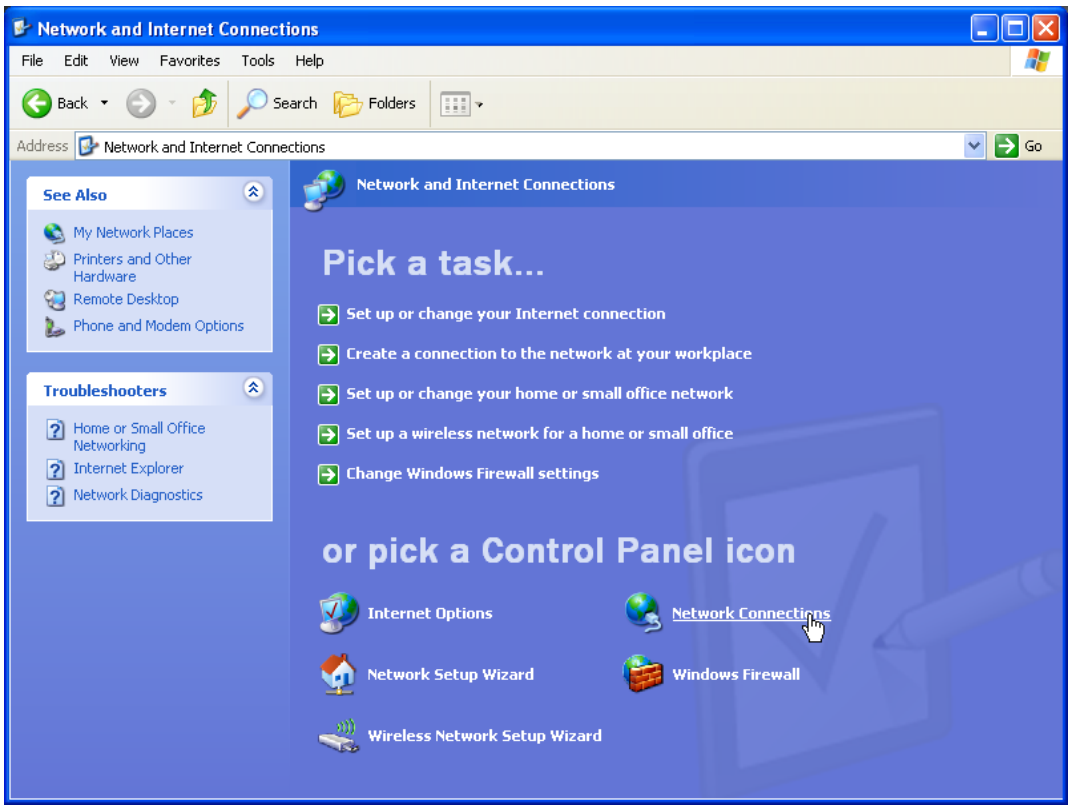
   (a) Click on the start button, and then on *Control Panel*.



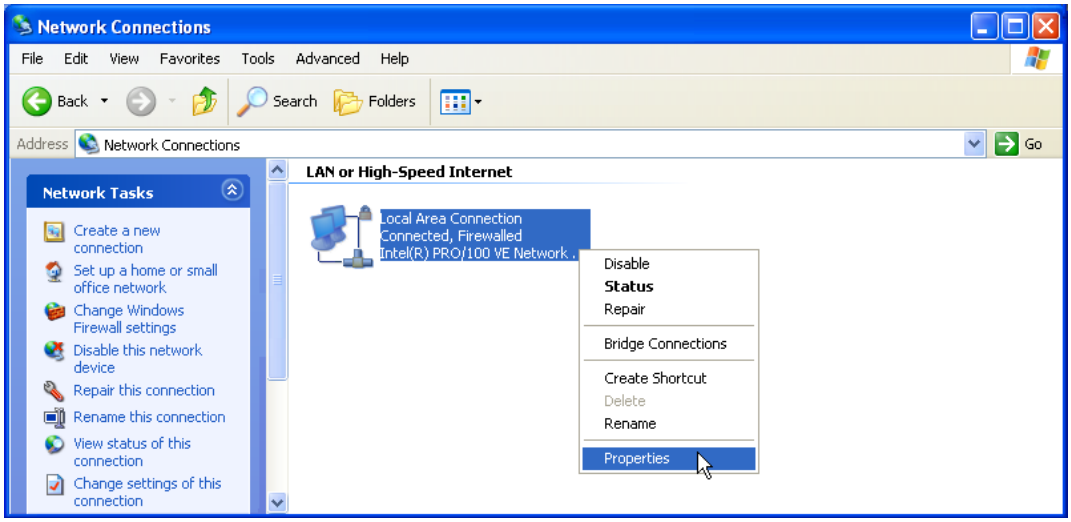
   (b) Once the control panel has come up click on *Network and Internet Connections*.



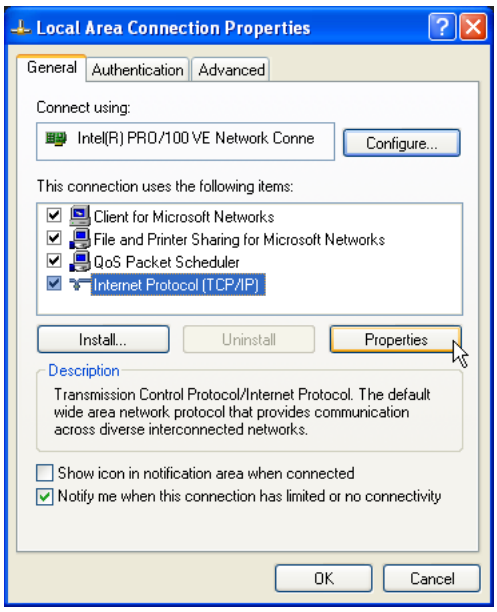
   (c) In the next window click on *Network Connections*.



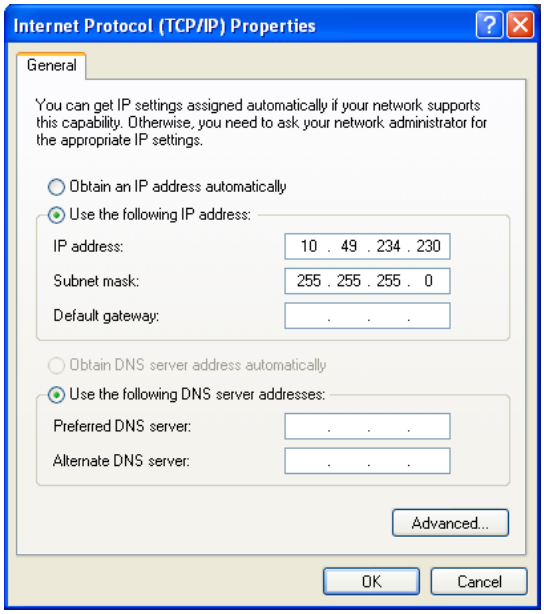
(d) The installed network connections will be shown, right-click on the local Ethernet connection to be used with the Licel Ethernet Controller and choose **Properties** from the context menu.



3. Click on the TCP/IP protocol entry in the lists of components used by the assigned Ethernet adapter card / LAN connection and press the *Properties* button.



4. Write down your current TCP/IP settings i.e. all settings seen in the following graphics. You will need this information to reconfigure your PC to access the LAN again.



5. If activated disable DHCP (checkbox *Obtain an IP address automatically*) and manually assign an IP address within the default address range of the Licel Ethernet Controller. A good choice would be `10.49.234.230`. **Never use the default address (`10.49.234.234`) of the Licel Ethernet Controller as IP address for your PC.**

6. Quit the dialog by pressing *OK*.

7. Reboot your PC.

8. Power up the rack with the Licel Ethernet Controller and connect the PC with the controller using the red **cross-link cable** shipped together with your hardware.

Now you should be able to access the Licel Ethernet Controller via your Ethernet card. Please test this first connection with the methods given in the next section.

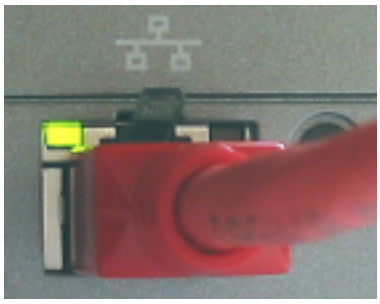
4.4.2 Diagnostics

Please carry out the following steps to verify that the connection of the Licel Ethernet Controller with the PC is established.

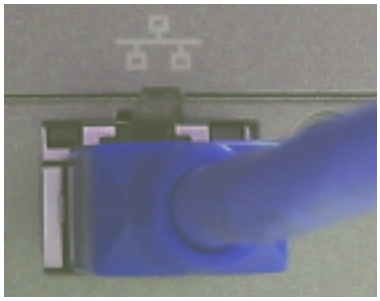
1. Verify that the green **LNK** LED lights up indicating a correct electrical connection.
2. Verify that in case of a 100Mbit Ethernet connection the **Spd** lights up.
3. Verify that the network settings of your PC have changed according to your settings:
   (a) Open a command prompt window (DOS box).
   (b) Type `ipconfig` and press enter. At least one of the Ethernet adapters should show the address that you previously set (`10.49.234.230`). The response should be similar to the following:

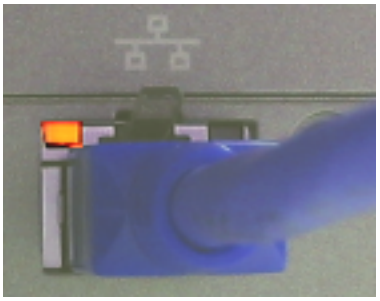
```
1 Ethernet Adapter :
        IP-Address. . . . . . . . . : 10.49.234.230
        Subnet Mask . . . . . . . . : 255.255.255.0
        Standard-Gateway. . . . . . :
```

4. Verify that the Licel Ethernet Controller is accessible via the network now:
   (a) Open a command prompt window (DOS box) or use the one from above.
   (b) Type `ping 10.49.234.234` and press enter. The Licel Ethernet Controller should respond without loss of any packet. If the controller is not responding check if the network cable is correctly mounted and that an appropriate cable is used, i.e. a cross-link cable when working with a direct connection from the computer. Most Ethernet adapters indicate a correct connection with a green LED:



A non-existent or incorrect connection is often identified by an unlighted LED (left) or red LED (right).





Please note that these indicators may be different on your PC.

   (c) If the network cable connection is correct and the controller is still not responding execute a hardware reset and repeat the procedure with the default IP address.

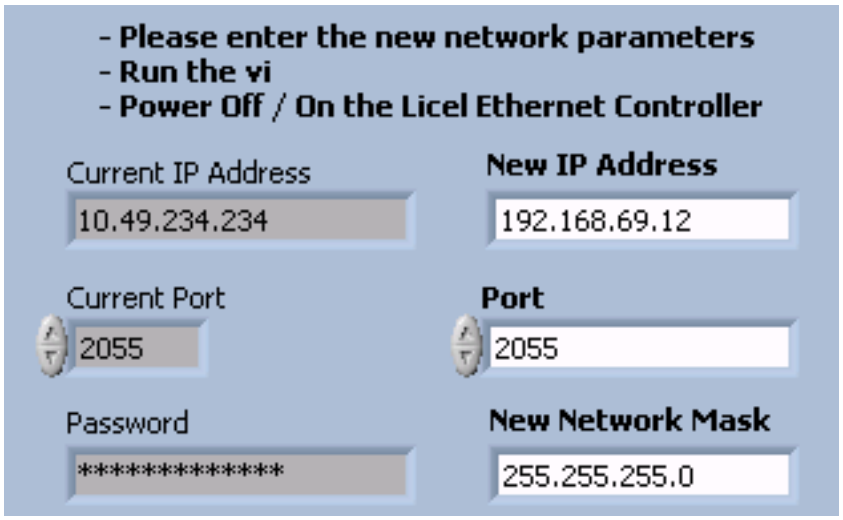
4.5 Network Setup

In order to configure the Ethernet controller, you need either to set the controller to a fixed IP address or invoke the DHCP Mode. Whether a fixed or dynamic (DHCP) mode is used or not will depend upon your network type. Dependent on this, please refer either to the subsection Fixed IP Address or DHCP Mode and skip the corresponding other subsection. Please contact your administrator if you have not yet requested the information described in the above subsection Network Setup.
Afterwards you will have to reconfigure your PC for operating in the local network.
Once you have set the **IP Address** and **Port** for the Licel Ethernet Controller you should define these values to be used by the software.

4.5.1 Fixed IP Address

If you need to set the controller to a fixed IP address carry out the following steps. Skip the steps described in next subsection DHCP Mode.

1. Open `Licel TCPIP Set New Fixed IP Address.vi` or the corresponding Windows application from the Windows start menu.

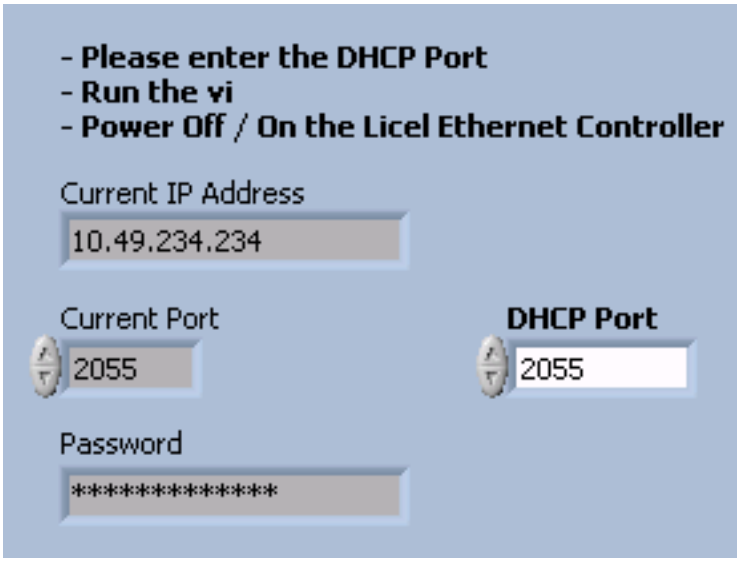


2. Set the desired network parameters in the fields **New IP Address**, **Port**, and **New Network Mask**.
3. Do not forget to enter the correct administrator **Password**.
4. Run the vi by pressing the start  button. It should finish without opening an error message dialog.
5. Turn the Licel Ethernet Controller off and switch it on again. Wait **approximately 20 – 30 seconds**.
6. A `ping 10.49.234.234` executed from a command prompt (DOS box) should now time-out.

4.5.2 DHCP Mode

In order to configure the Licel Ethernet Controller for DHCP operation carry out the following steps. You must have skipped the steps described in the last subsection Fixed IP Address.

1. Open `Licel TCPIP Activate DHCP Mode.vi` or the corresponding Windows application from the Windows start menu.

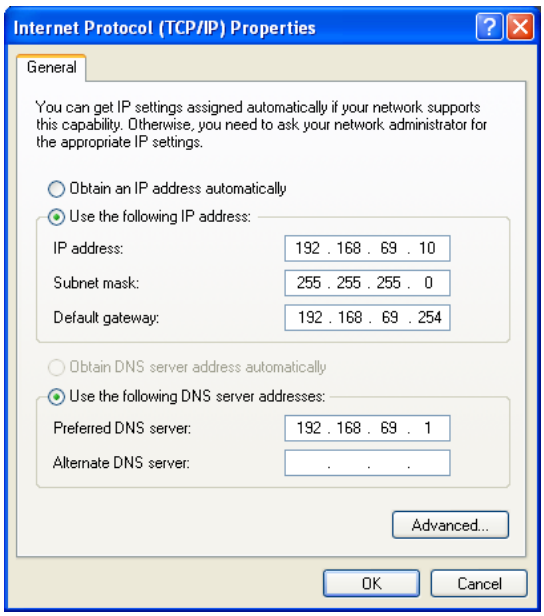


2. Set the desired **DHCP Port** number.
3. Do not forget to enter the administrator administrator **Password**.
4. Run the vi by pressing the start button. It should finish without opening an error message dialog.
5. Turn the Licel Ethernet Controller off and switch it on again. Wait **approximately 20 – 30 seconds**.
6. A `ping 10.49.234.234` executed from a command prompt (DOS box) should now time-out.

4.6 Reconfigure the PC

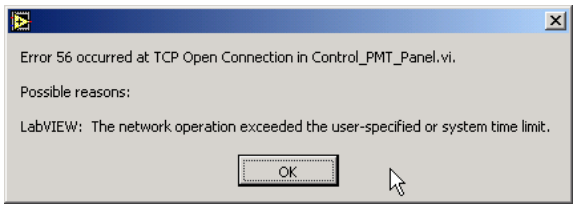
After you successfully configured the Licel Ethernet Controller the following last steps have to be carried out to reconfigure your PC for the local network and to test the connection to the Licel Ethernet Controller:

1. Reconnect the PC to the local network.
2. Open the **Properties** dialog of the network connection your Ethernet adapter is assigned to. A more detailed instruction has been given above.
3. Open the **Properties** dialog of the TCP/IP protocol entry in the lists of components used by the assigned Ethernet adapter card.
4. Reset your current TCP/IP settings to the values you recorded while processing the subsection to establish a network connection.



Note that the values shown here are just example settings. You must exactly use the settings present on your PC before configuring the Licel Ethernet Controller.

5. Quit the dialog by pressing *OK*.
6. Reboot your PC.
7. Connect the Licel Ethernet Controller with your local network through a hub or switch using an **ordinary patch cable**.
8. Execute a `ping` command from a command prompt (DOS box). Use the IP address you assigned to the Licel Ethernet Controller. If the Ethernet controller is in DHCP mode, you need to ask your system administrator for the assigned network address. The `ping` command's response should indicate a correctly working connection.
9. Test the access using `Licel TCPIP Getting Started.vi` or the corresponding Windows application to be started from the Windows start menu.
10. A TCP/IP timeout error with LabVIEW's error code 56 may be caused by a wrong IP address.



Please check carefully that the values for **IP Address** and **Port** match with the parameters set at the Licel Ethernet Controller. Set the correct values as defaults for future operation. Other reasons for errors with code 56 are non-existing connections (check if the cable in use is correct) or unstable network operation.

4.7 TCP/IP Connection Parameters

To work properly with the Licel Ethernet Controller both the Windows applications and the LabVIEW software must be able to establish a TCP/IP connection. The user of the software must define the **IP Address** and **Port** – these values must be equal to the parameters that have been for the Licel Ethernet Controller following the network setup section.
Defining the **IP Address** and **Port** is different for the Windows applications and the LabVIEW sources.

Windows Applications: Initialization Files

The Windows applications communicating with the Licel Ethernet Controller use initialization files to read their TCP/IP parameters **IP Address** and **Port**.
An example for an initialization file holding the TCP/IP information is given below:

```
[TCPIP]
UseValues=TRUE
IPAddress=10.49.234.234
Port=2055
```

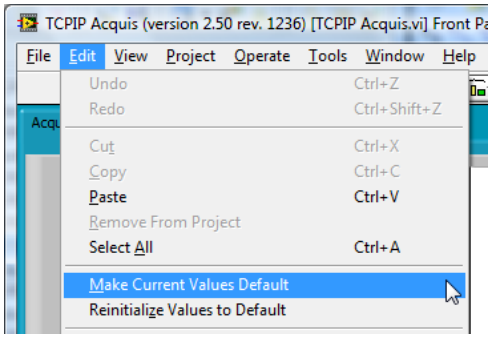
You may edit this file using a text editor like `Notepad` which is installed by default when setting up a Windows operating system. You may use `Notepad` as well to create a required initialization file if it does not exist in the installation directory. Make sure that you save the file before leaving the editor. You must change the values for IP address and port to the values you will set following the Instructions in the network setup section.
Here is an overview of the initialization file names used by the Licel Windows applications for reading the TCP/IP information:

Windows Application	**Initialization File**
`Control APD-PMT.exe`	`Control APD-PMT.ini`
`Control Timing.exe`	`Control Timing.ini`
`Power Meter Control.exe`	`Power Meter Control.ini`
`TCPIP Acquis.exe`	`Acquis.ini`
`M-Acquis.exe`	`Acquis.ini`
`TCPIP Live Display.exe`	`TCPIP Live Display.ini`
`TCPIP Pulse Height Distribution.exe`	`TCPIP Pulse Height Distribution.ini`
`TCPIP Track.exe`	`TCPIP Track.ini`
`Licel Main.exe`	`Licel Main.ini`

LabVIEW: Setting Default TCP/IP Parameters

The initialization files described above are necessary for the Windows applications because there it is not possible to set specific values as default values for control fields. However, when running the software within a LabVIEW development environment, default values can be defined for controls on the panel of a LabVIEW vi. This is especially convenient and recommended for the TCP/IP parameters **IP Address** and **Port**. Change the values to the values you set following the Instructions in the network setup section.

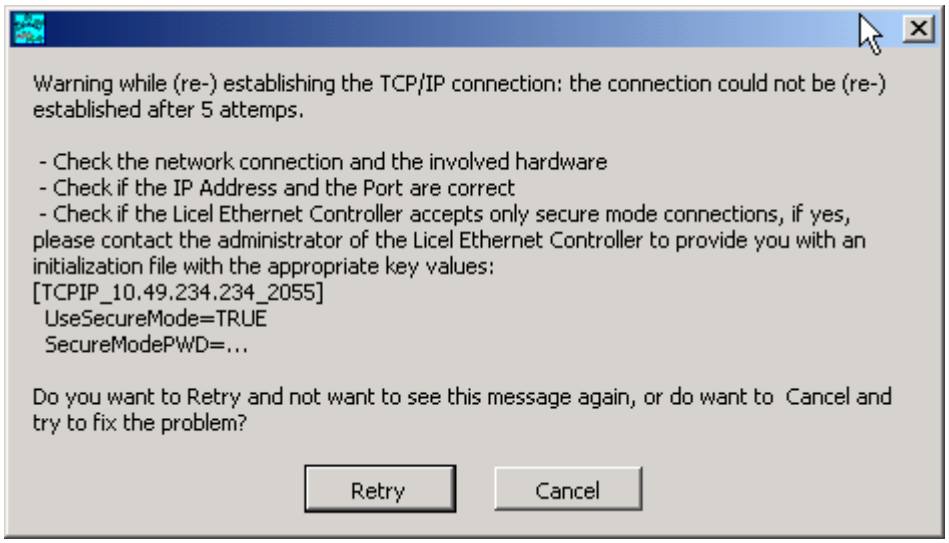
1. Open the vi using LabVIEW, do not run the vi.
2. Enter the value for the IP address into the control named **IP Address**.
3. Right-click on the control **IP Address** ⟶ the context menu opens.
4. Select **Data Operations** ⟶ a sub menu opens.
5. Select **Make Current Value Default**.



6. Repeat this procedure for **Port**.
7. Save the vi.

4.7.1 TCP/IP Connection Problems (Software)

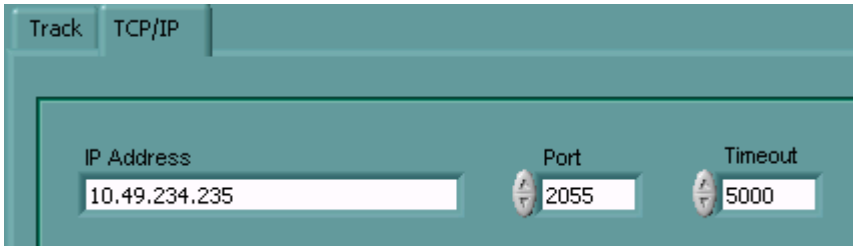
The parent application *Licel Main* as well as the software modules when run stand-alone (*Track*, *Live*, *Acquis*, *Control APD-PMT*, and *Control Timing*) have a built-in mechanism to re-establish the TCP/IP connection to the Licel Ethernet Controller when the connection is lost or when the connection is not successful after the program start.
If the reconnection mechanism is not successful after 5 attempts the software assumes that some basic TCP/IP settings may be incorrect. Therefore the following error message is displayed:



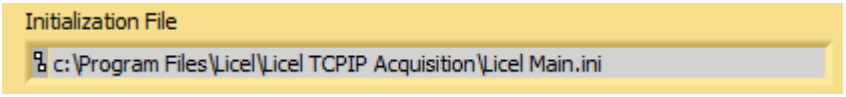
In the case that this dialog comes up please

- check the network connection and the involved hardware. Check whether the Licel Ethernet Controller and all other Licel hardware is switched on. Check that the Ethernet cable is plugged correctly, and that the correct Ethernet cable is used.
- check whether the **IP Address** and the **Port** the software is using equal to the values of the Licel Ethernet Controller (refer to the network setup).

  1. Before you start please enter the correct values for the **IP Address** and **Port**. You should already have set these values for the Licel Ethernet Controller
     – Using the LabVIEW vi, just enter the required values on the *TCP/IP* page and save them as defaults.



     – If you run a Windows application you should set the values in the corresponding initialization file. You will see the full path of the file in a file path indicator on the *TCP/IP* page.



While a Licel Windows application **is running** (and has not yet a TCP/IP connection) you may enter the **IP Address** and the **Port** directly. If a connection can be established (i.e. the values are correct) the parameters will be written to the appropriate initialization file when the program terminates without an error.

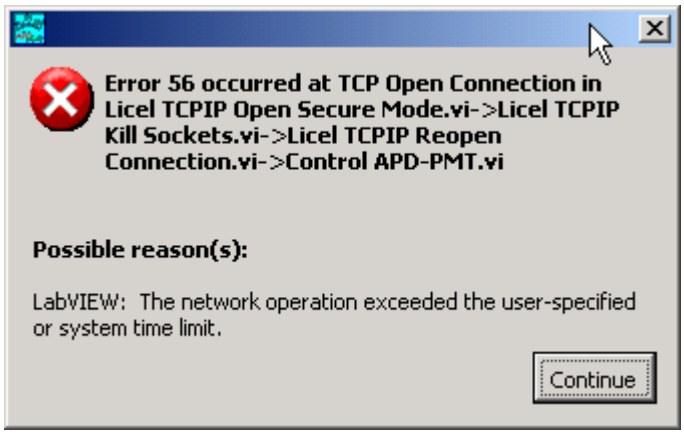
- check whether the Licel Ethernet Controller operates in secure mode. If secure mode is used please check the following section in the file `LicelTCPIP.ini`:

  ```
  [TCPIP_<Controller-IP-Address>_<Controller-Port>]
  UseSecureMode=TRUE
  SecureModePWD=<SecureModePassword>
  ```

  where `Controller-IP-Address` and `Controller-Port` are the IP address and port of the Licel Ethernet Controller, respectively. If necessary, ask your administrator for the correct password for usage in secure mode.

You have two choices to leave the message dialog:

1. Click *Retry* to continue to reconnect to the Licel Ethernet Controller.
2. Click *Cancel* to exit. The program will display an error message (here an example for *Control APD-PMT*, the dialog's appearance may depend on the LabVIEW version):



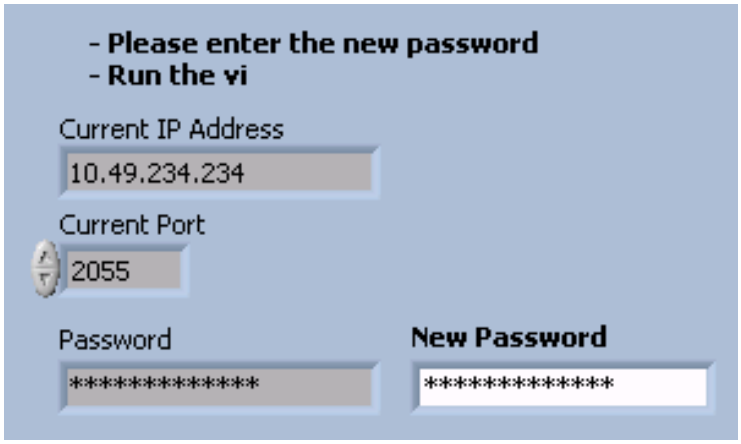
4.8 Network Security

The Licel Ethernet Controller provides two levels of network security. Certain administrative tasks use an administrator password. An example is the change of the IP address of the controller. The administrator password has to be sent with the related commands. Furthermore a *Secure Mode* based on an encryption mechanism is available.

4.8.1 Changing the Administrator Password

The Licel Ethernet Controller is shipped with the default administrator password "'*Administrator*"'. In order to change this password which grants administrative access to the controller, please carry out the following steps:

1. Open `Licel TCPIP Set New Password.vi` or start the corresponding Windows application from the Windows start menu.



2. Enter the current administrator **Password**.
3. Enter the **New Password**.
4. Run the vi by pressing the start button. It should finish without opening an error message dialog. Please note that the password is case sensitive.

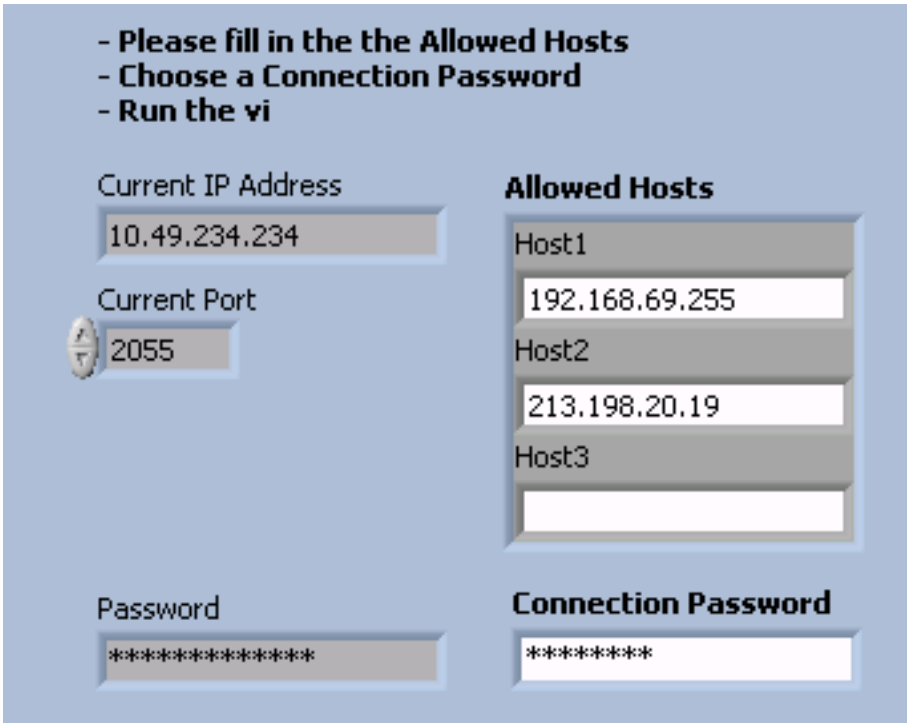
4.8.2 Secure Mode

The Licel Ethernet Controller might be the target of an attack. The best protection against this is to run the controller with a private IP address beyond a firewall. Firewalls are designed to protect against various types of attacks that can not be covered by the Ethernet controller. Licel strongly recommends the use of a firewall/router combination to prevent unauthorized use of the hardware.
Starting with firmware versions from 2005-02-22 (`state53`) the Licel Ethernet Controller has an additional level of security that can be additionally used.
This secure mode combines whitelisting of allowed hosts with an encrypted password transmission scheme.

Enabling the Secure Mode

In order to enable the Secure Mode for the Licel Ethernet Controller carry out the following steps:

1. Open `Licel TCPIP Enable Secure Mode.vi` or the corresponding Windows application from the Windows start menu.



2. Set the desired whitelist of allowed host IP addresses or address ranges. An entry in the list of **Allowed Hosts** is either

   - a host specified by its IP address `xx.xx.xx.xx` ,
   - an IP address range `xx.xx.xx.255` ranging from 0 to 255, or
   - empty .

   At least one valid entry must be submitted. Do not forget to include the IP address of the PC you are currently using.

3. Set the **Connection Password**. This password must be used by clients accessing the Licel Ethernet Controller as long as the secure mode is enabled. Refer to the `LOGON` command for further details. Please note that the password is case sensitive.

4. Do not forget to enter the administrator **Password**.

5. Run the vi by pressing the start  button. It should finish without opening an error message dialog.

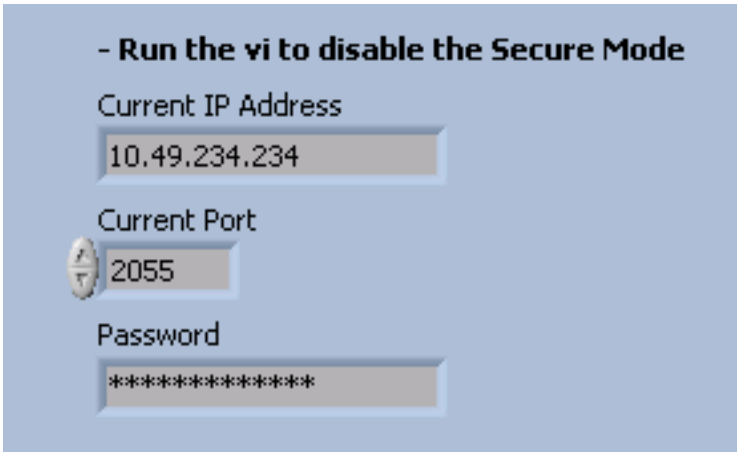
6. The vi will write an initialization file `LicelTCPIP.ini` with appropriate keys and values. These values are used by the sample applications while establishing a connection to the controller. Distribute the initialization file to all PCs the sample programs are installed on.

7. Test the access using `Licel TCPIP Getting Started.vi` or the corresponding Windows application to be started from the Windows start menu.

Disabling the Secure Mode

In order to disable the Secure Mode for the Licel Ethernet Controller carry out the following steps:

1. Open `Licel TCPIP Disable Secure Mode.vi` or the corresponding Windows application from the Windows start menu.



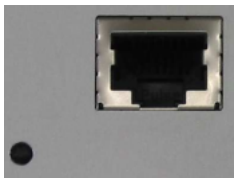
2. Do not forget to enter the administrator administrator **Password**.

3. Run the vi by pressing the start button. It should finish without opening an error message dialog. Note that this vi can only be used if the Licel Ethernet Controller is running in secure mode and if a valid initialization file `LicelTCPIP.ini` is located in the directory where the vi's library resides.

4. The vi will update the initialization file `LicelTCPIP.ini` with appropriate keys and values. These values are used by the sample applications while establishing a connection to the controller. Distribute the initialization file to all PCs the sample programs are installed on.

4.9 Hardware Reset

A reset is performed by pressing the reset switch while powering up the controller. The reset switch is located inside a hole close to the RJ45 connector.



To reset the system

- turn off the controller unit
- press the switch inside the hole with a small screw driver, Allen key or anything similar
- turn the rack on while keeping the switch pressed, release the switch 5 seconds after switching the unit on, wait for 45 seconds.

After a reset

- the controller has the default IP address
- the port number is reset to the default value
- the controller operates in its fixed IP address mode
- the password is reset to the default password.

Chapter 5

Licel TCP/IP Software Modules

Licel provides a number of standard software modules or example applications to communicate via TCP/IP with the available control modules. The software modules are described later in this manual. All these TCP/IP software modules are capable to run stand alone.

5.1 Licel Main

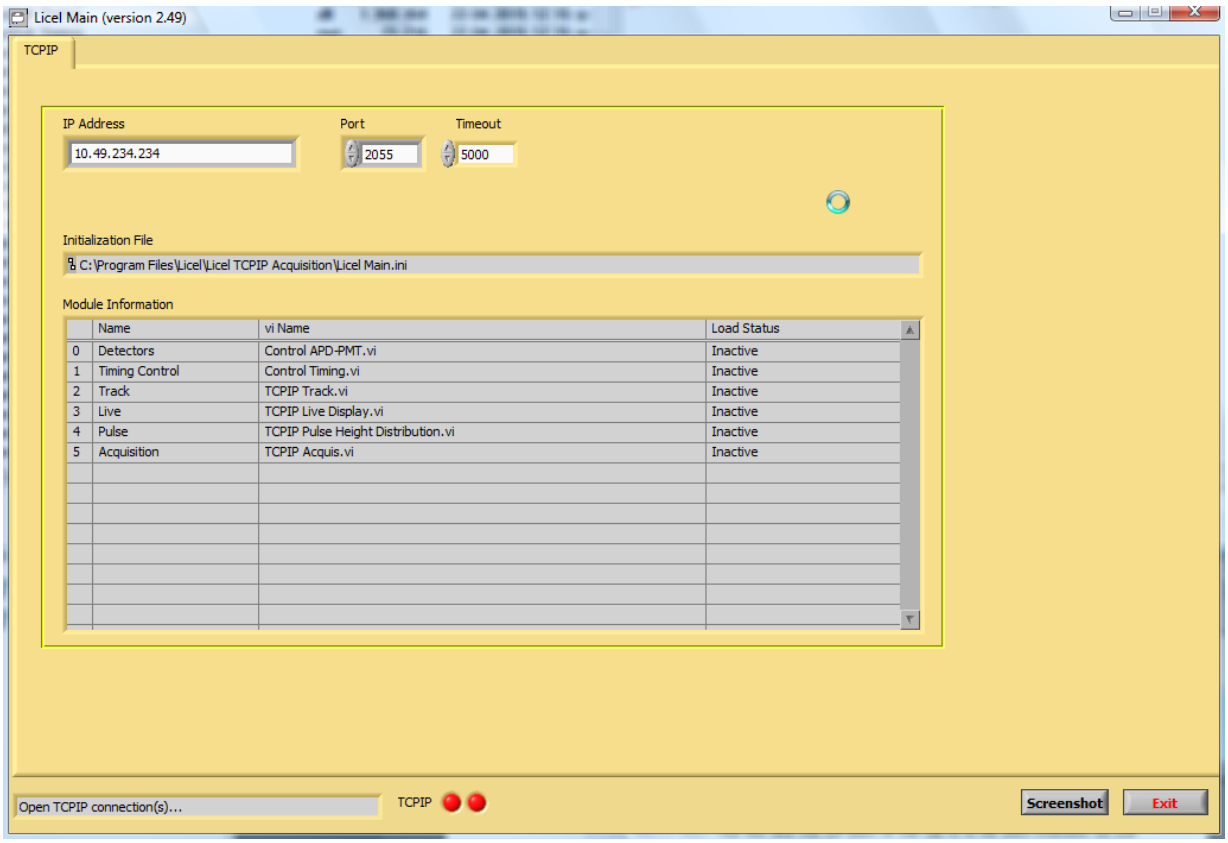
In many cases a single Licel Ethernet Controller supports the control of different control modules, for example transient recorder and PMT high voltage control modules. To support all these capabilities attached to one IP address with a single software component a parent application named `Licel Main` has been designed. `Licel Main` embeds one or more TCP/IP software modules and is responsible for the handling of the TCP/IP connection.
Since the release of Licel Acquisition Software TCPIP version 2.49 `Licel Main` is capable to manage the control of up to two Licel Ethernet Controllers each having it's own IP address.
With the installation of the Licel Acquisition Software TCPIP `Licel Main` is installed in the following variants:

1. `Licel Main.exe`: classic variant for one rack with one Ethernet controller at one IP address
2. `Licel Main-M.exe`: for up to six Ethernet controllers at six different IP addresses.

A Licel TCP/IP software module is capable to run within a sub panel on a tabulator page of the parent application `Licel Main`.

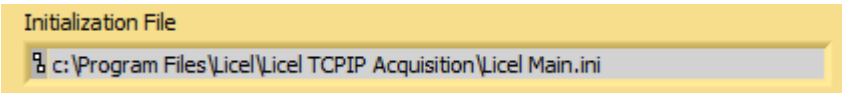
5.1.1 Starting Licel Main

You can load Licel Main by selecting the virtual instrument `Licel Main.vi` in your LabVIEW folder. If you installed the Windows applications please start the program by selecting the corresponding entry in the Licel section of the Windows Start menu. After loading a screen like the following will be visible (here the Windows application is shown).

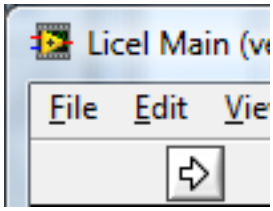


1. First of all the **IP Address** and **Port** have to be set. You should already have set these values for the Licel Ethernet Controller following the network setup section above.

   - Using the LabVIEW vi, just enter the required values on the *TCP/IP* page and save them as defaults.
   - If you use the Windows application you must directly enter the correct values into the corresponding control fields on the *TCP/IP* page. On exit the values will be saved to the initialization file in the case that the TCP/IP connection is established.
     You may also set the values in the initialization file `Licel Main.ini`. You will see the full path of the file in a file path indicator on the *TCP/IP* page.

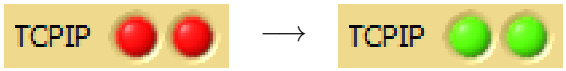
2. To start the program press the **Run** button at the top left of the screen.

   The Windows application will start automatically when called for the first time.

3. After a short time the **TCPIP** indicator at the bottom should change its color from red to green indicating a successful connection with the Licel Ethernet Controller. If the indicator remains red and/or an error is indicated, please check the values for **address** and **Port**, change them (on the program's panel or in the initialization file) if necessary. Check if the Licel Ethernet Controller is running and that all network connections are correct. The LED of the transient recorder(s)

should be lit up if transient recorder control modules are loaded. If TCP-Live Display is loaded, a second LED indicator as seen below will be visible.



After starting the application will load all requested software control modules to the sub panels on the different tabulator pages if

1. the corresponding vi has been found
2. the controller's capabilities support the requested capabilities
3. the module could be successfully initialized.

Please refer to the initialization file `Licel Main.ini`.
The status of the software modules `Licel Main` attemps to load and run can be viewed in the table on the *TCP/IP* page. A software module can be loaded when the corresponding hardware is found. In the last column it is indicated whether or not a software module is supported by the current Licel Ethernet Controller. In the example below `Control Timing` is not supported because the addressed Licel Ethernet Controller is installed at a Rack6 equipped with transient recorders and a PM-Remote control.

Module Information

	Name	vi Name	Load Status
0	Detectors	Control APD-PMT.vi	Running
1	Timing Control	Control Timing.vi	Not Supported
2	Track	TCPIP Track.vi	Running
3	Live	TCPIP Live Display.vi	Running
4	Pulse	TCPIP Pulse Height Distribution.vi	Running
5	Acquisition	TCPIP Acquis.vi	Running

5.1.2 Operating Licel Main

The following controls are available while operating `Licel Main`:

- **IP Address**: IP address of the accessed controller. It is recommended to set this value before starting the program, or to enter it to the initialization file `Licel Main.ini` for operating the windows application.
- **Port**: TCP/IP port for the communication with the accessed controller. It is recommended to set this value before starting the program, or to enter it to the initialization file `Licel Main.ini` for operating the windows application.
- **Timeout**: Timeout in milliseconds for the TCP/IP communication (default value: 5000 ms).
- Status indicators at the bottom:

  - **TCPIP** LEDs: TCP/IP alive indicators (see above)
  - **Status**: Status message indicator. After starting the program the following message sequence will be shown: *Open TCPIP connection(s)...*, *Search Modules...*, *Load Modules...*, *Initialize Module* <...> for all modules to be loaded, *Ready*.
- **Screenshot** A click on this button will send a screenshot of the current sub panel to the standard printer.

- **Exit** A click on this button will exit the application.

The different software modules may be accessed just by clicking on the corresponding tab pages. The software work as if they would run stand-alone. Only the monitoring of the TCP/IP connections is managed by `Licel Main`. If a TCPIP connection is lost `Licel Main` will take the control from the currently active vi, try to reconnect to the Licel Ethernet controller, reset all currently used values (PMT voltage, timing values, etc.) after a successfully established connection, and return the control to the currently active vi.

5.1.3 Software Module Load Definition

The initialization file `Licel Main.ini` contains information for `Licel Main` about the software modules to load and run. The information is organized in terms of initialization key blocks like the following:

```
[Module2]
Active=TRUE
Path=TCPIP Track.vi
Name=Track
Controllers=0
CAPs=TR
Push=FALSE
IndependentTCPIP=FALSE
```

The load instructions for `Licel Main` may be set following the instructions above:

`ACTIVE`	`TRUE` | `FALSE`	load the vi or not
`Path`	<vi path>	path of the vi (without llb)
`Name`	a chooseable name	is displayed as tab label
`Controllers`	controller index	
`CAPs`	comma-separated list of capabilities	Licel Main will check for these entries (`OR`)
Push	`TRUE` | `FALSE`	push socket in use (TCPIP Live Display, MPush Acquis)
IndependentTCPIP	`TRUE` | `FALSE`	the loaded module will use a different Ethernet Controller not managed by `Licel Main`, the feature is supported by the software modules `Control APD-PMT` and `Power Meter Control`

Here, the controller index is always 0 (`Controllers` = 0), i.e. `Licel Main` will always use the same Licel Ethernet Controller for each software module.

5.1.4 Licel Main-M

`Licel Main-M` is a variant of `Licel Main` to support the communication and data acquisition with up to six Licel Ethernet Controllers with individual IP addresses. `Licel Main-M` is available as a LabVIEW VI (`Licel Main-M.vi`) and as a Windows application (`Licel Main-M.exe`). Both use the initialization file `Licel Main-M.ini`. With this initialization file `Licel Main-M` will work with two racks, each equipped with transient recorders and PM-Remote controller. This ini file may be exended for the use of up to six Licel Ethernet Controllers.
The behavior of `Licel Main-M` and the loaded software modules is defined in the initialization file `Licel Main-M.ini`. Each software module section `[Module`*n*`]` contains the entry `Controllers` to specify wether the first rack (`Controllers` = 0) or the second rack (`Controllers` = 1) is used. Also specified is the definition for Licel's multi-rack acquisition software `M-Acquis`:

```
[Module8]
Active = TRUE
Path = M-Acquis.vi
Name = Acquisition
Controllers = 0;1
CAPs = TR
Push = FALSE
IndependentTCPIP=FALSE
```

Here, `Controllers = 0;1` determines that both racks are addressed by the software module. The user interface of `Licel Main-M` is equal to that of `Licel Main` with the exception of the IP addresses and ports and the status indicator. The controls allow to set all IP addresses and ports corresponding to the definition in the initialization file `Licel Main-M`.ini.

Additional TCP/IP indicator LEDs are shown for the second TCP/IP connection.

Chapter 6

Transient Recorder Software Tutorial

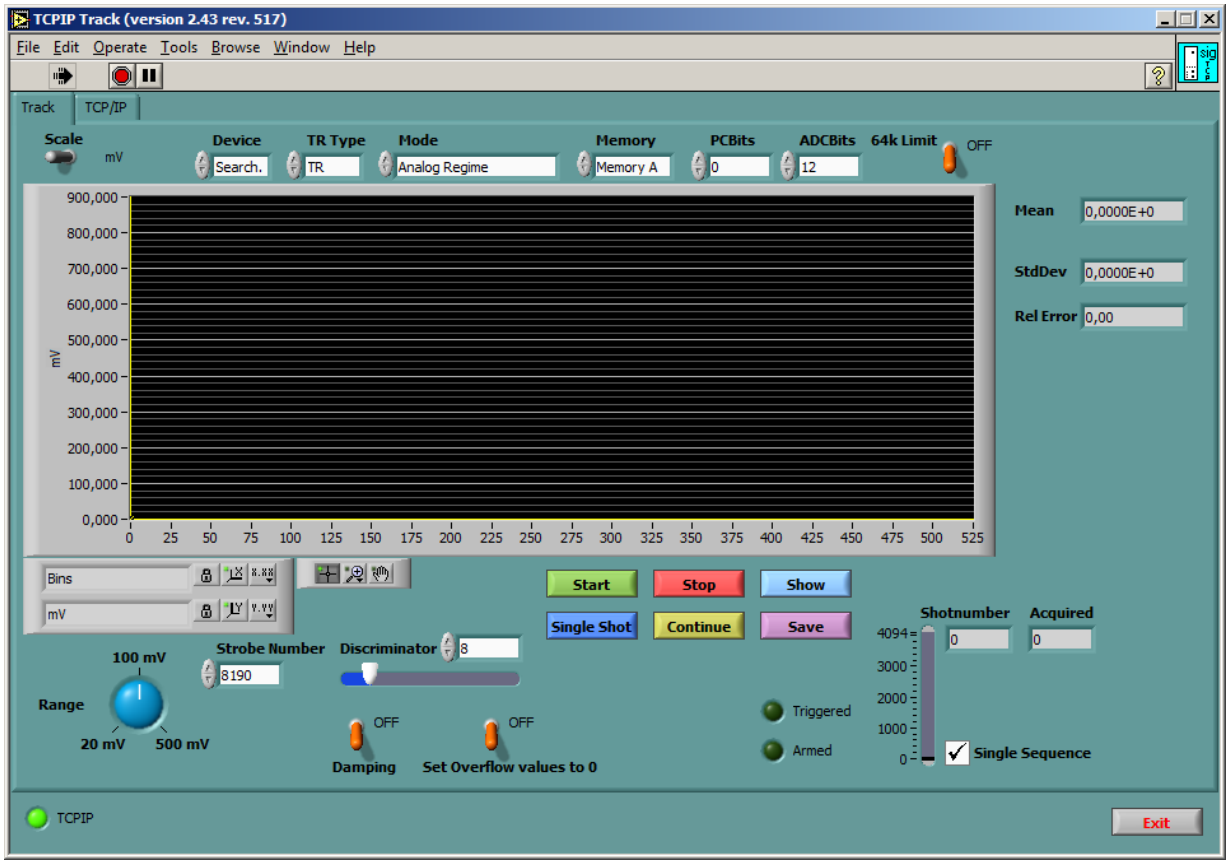
6.1 Overview

This software tutorial describes how to use the data acquisition software as well as the functions of the individual controls and indicators. In order to actually try the information in this tutorial, the hardware and network setup must be completed. This tutorial is broken into two parts. The quick tour gives a brief introduction to recording spectra with the software module TCPIP Track and TCPIP Live Display. The Acquisition with TCPIP Acquis contains instructions for recording your first spectra using TCPIP Acquis.

6.2 Quick Tour

6.2.1 TCPIP-Track

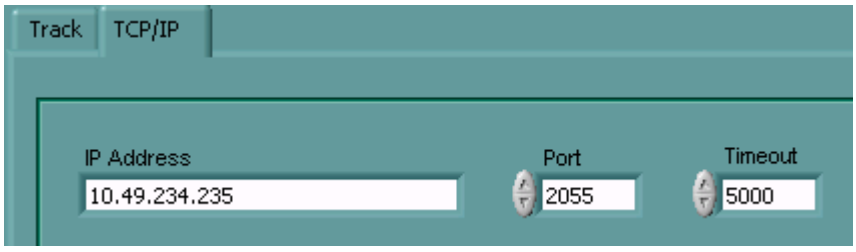
You can load TCPIP Track by either double clicking on the `TCPIP-Track.llb` or by selecting the virtual instrument `TCPIP Track.vi` in `TCPIP-Track.llb`. If you installed the Windows applications please start the program by selecting the corresponding entry in the Licel section of the Windows Start menu. After doing so, you should see a screen similar to the one below. TCPIP Track is a program that can be used to access all the individual functions of the transient recorders. It allows you to control one individual recorder at a time.



If you have completed the hardware setup and configured the network you should be ready to experiment with the software. Do the following steps to get a brief introduction of the software TCPIP Track.

1. First of all the **IP Address** and **Port** have to be set. You should already have set these values for the Licel Ethernet Controller following the network setup section above.

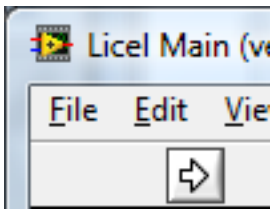
   - Using the LabVIEW vi, just enter the required values on the *TCP/IP* page and save them as defaults.



   - If you use the Windows application you must directly enter the correct values into the corresponding control fields on the *TCP/IP* page. On exit the values will be saved to the initialization file in the case that the TCP/IP connection is established.
     You may also set the values in the initialization file `Licel Main.ini`. You will see the full path of the file in a file path indicator on the *TCP/IP* page.

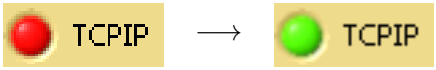
   - If you run `TCPIP Track` within a sub panel on a page from `Licel Main` the latter is responsible for managing the TCP/IP connection.

2. To start the program press the **Run** button at the top left of the screen.



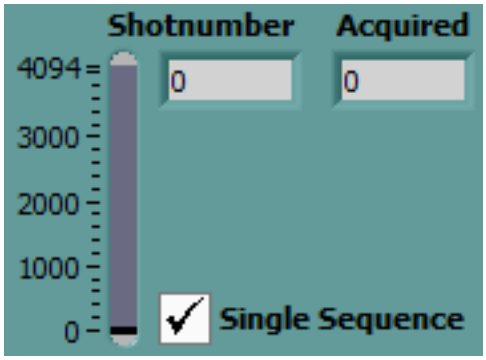
The Windows application will start automatically when called for the first time.

3. After a short time the **TCPIP** indicator should change its color from red to green indicating a successful connection with the Licel Ethernet Controller. If the indicator remains red and/or an error is indicated, please check the values for **address** and **Port**, change them (on the program's panel or in the initialization file) if necessary. Check if the Licel Ethernet Controller is running and that all network connections are correct. The LED of the transient recorder should be lit up.



4. Press the **Start** button directly below the waveform graph.

5. After pressing Start, the **Shotnumber** should start increasing. The shot number is increased by one for every trigger pulse that is received.



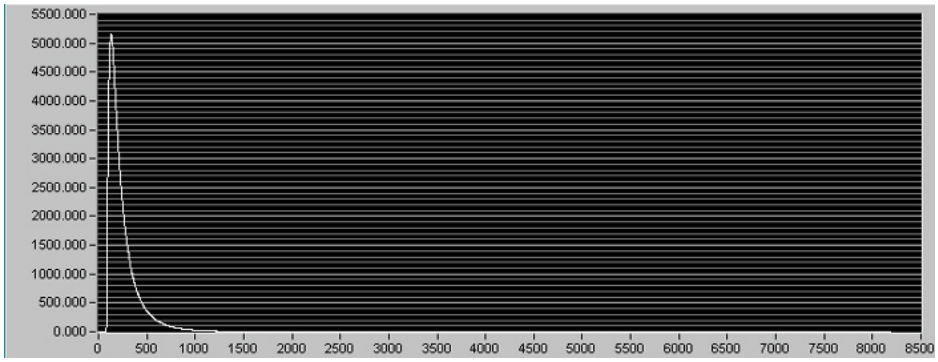
If the 

-option is checked the acquisition will stop at 4094 (or 65534 if **64k Limit** = *ON*) received trigger pulses. The behavior of the unchecked option will be explained below.

6. Press the **Stop** button after a few seconds to stop the acquisition

7. Press the **Show** button to display the results.

8. You will see the acquired signal in the graph window. Your signal will differ depending upon your system configuration. The example below is the simulation of a well aligned low noise system.



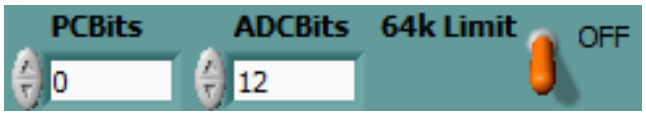
9. The data set that is displayed after pressing the **Show** button is selected by the **Device**, **TR Type**, and **Mode** switches at the top of the waveform graph.

Device refers to the hardware address of the transient recorder. Before selecting the **Mode** you must set the **TR Type** in compliance with the external hardware module that you are addressing. The transient recorder is either a TRxx-xx or PRxx-xx, **TR Type** must be set to *TR* or *PR*, respectively. Both types differ in their memory layout as you might see in the programming manual (http://www.licel.com/manuals/programmingManual.pdf, page 16 ”Memory organization”). **Mode** depends on the **TR Type**:

TR Type	**Mode**	
TR	*Photon Counting*	the accumulated data from the counting chain is displayed
	Analog	the ADC data is displayed
PR	*PR Photon Counting*	the photon counting data is displayed

The data set is further specified by the **Memory** switch. *Memory A* corresponds to acquisitions which were triggered by Trigger A, and *Memory B* corresponds to acquisitions triggered at input B. If only one trigger input is connected to the trigger source, only one memory can hold data different from 0.

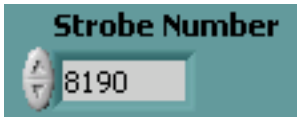
10. Further settings for the selected transient recorder (**Device**) are available at **PCBits**, **ADCBits**, and **64k Limit**.



The numbers of photon counting bits and analog bits (**PCBits** and **ADCBits**) of the selected transient recorder are read from the Licel Ethernet Controller (supported since spring 2011). In the case that the controller does not support the request the numbers of photon counting bits and analog bits can be set by the user. The values must correspond to the installed transient recorders.

With the **64k Limit** the maximum number of acquirable shots (shot limit) is selected: *OFF* and *ON* correspond to the shot limits 4094 and 65534, respectivlely. **64k Limit** = *ON* is supported only by 16 bit transient recorders connected to an appropriate controller (available since 2011).

11. The horizontal scale is given in bins. The number of displayed bins is controlled by the **Strobe Number** control. This number should not exceed the memory length of the transient recorder; for a TR-xx-80 this would be 8190 and for a TR-xx-160 16380 bins.



12. The vertical scale has different meanings for the analog and the photon counting mode.

    - For photon counting data the mean number of counts per bin is displayed. For a given range bin the number of accumulated counts is divided by the shot number.
    - For the analog data the display can be either in mV or in least significant bits (LSB) and may be selected using the most top-left switch:

      The LSB scale corresponds to the ADC reading, for a 12 bit transient recorder it can vary between 0 and 4095.

13. Press **Continue** to continue accumulation without clearing the memory. Pressing **Start**would clear both memories.

14. After a few seconds press the **Stop** button followed by the **Show** button and notice how the signal-to-noise ratio has improved.

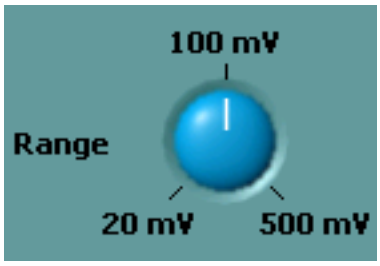
15. The acquired data can be saved using the **Save** button.

    The data is written to a binary file (single precision 4 byte floating point) using LabVIEW's `vi.lib\Utility\file.llb\Write To SGL File.vi`.

16. If the 

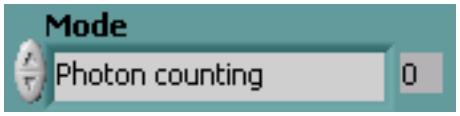
-option is not checked the acquired data will be read from the transient recorder and displayed once the shot counter reaches 4094 (or 65534 if **64k Limit** = *ON*). The acquired shot number will be displayed in the **Acquired** field. Then the transient resorder is restarted to acquire the next up to 4094 (64534) shots. The next time the counter reaches 4094 (64534) the data read from the transient recorder will be added to the data acquired before.

17. Change the input sensitivity with the **Range** knob located at the bottom left.



The three displayed millivolt values indicate the full scale negative voltages.

18. Take a new acquisition by repeating steps 1 through 7 to see the influence of the input range.
19. Press the **Stop** button to stop the acquisition.
20. Connect a photomultiplier to the signal input on the transient recorder and switch to photon counting mode.



21. Using the **Discriminator** slide, you can set the discriminator level between 0 and 63.

Change to the photon counting mode and make a few acquisitions to see how the count rate is influenced by the discriminator setting.

22. Change back to analog mode
23. Turn the **Damping** switch on and make a new acquisition

This reduces the counting rate since the discriminator level is set four times as high.

24. The **Set Overview Values to 0** switch helps you to analyze whether the signal exceeds the acquisition range.

Once you have acquired a real signal and the overflow at the transient recorder flashes switch the **Set Overflow Values to 0** switch and see if you see spikes towards zero. Those spikes will give you the exact position where the overflow occurred. With this information you can decide if you have overflows or underflows. Overflows can be avoided by increasing the signal input range or lowering the amplitude of the input signal underflows might require recalibration of the instrument as shown in http://www.licel.com/manuals/TRInstallation.pdf (section 5.3 Calibration).

Underflows are a serious thread to signal integrity as they are hard to notice and might distort the background computation. They should be fixed as otherwise the far field signal will be incorrect.

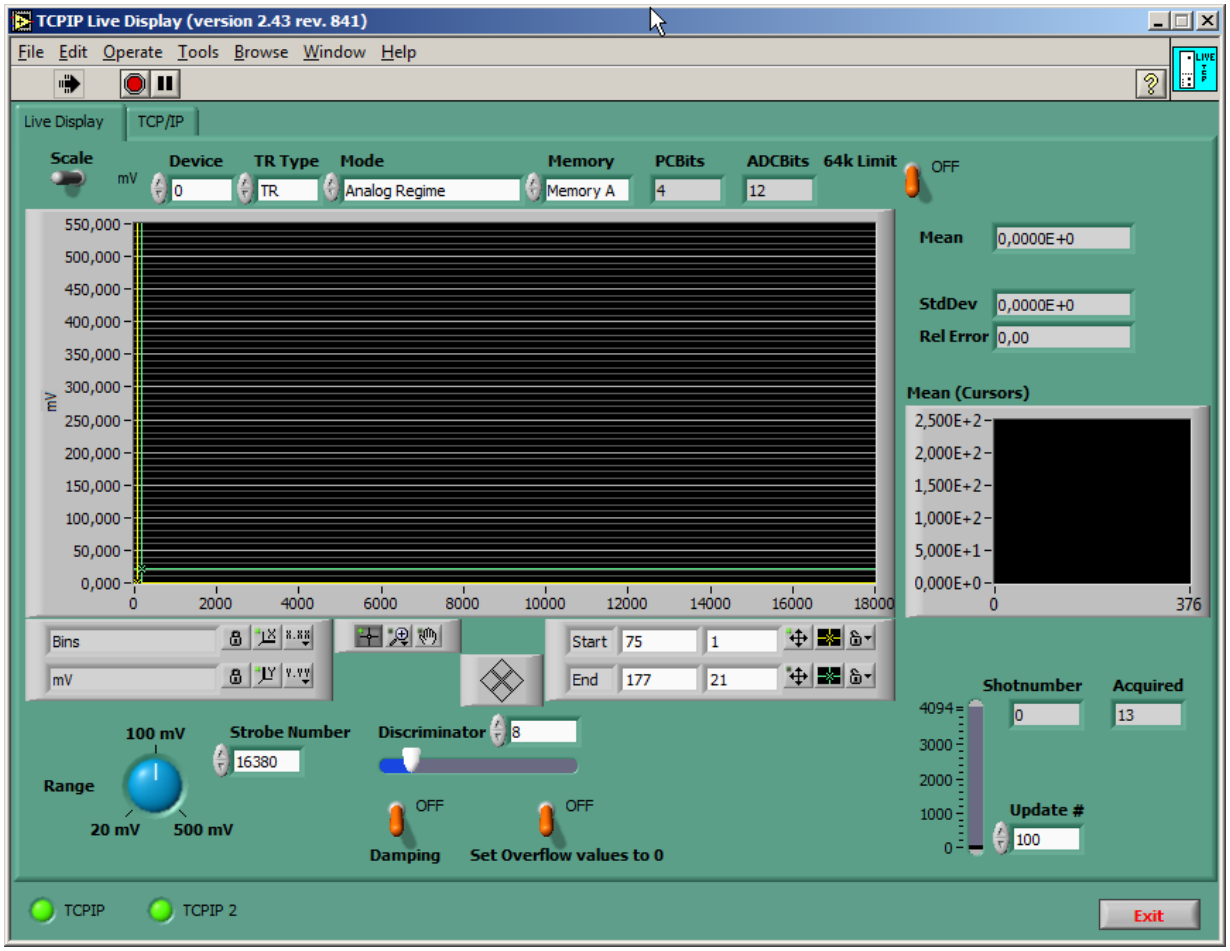
25. The **triggered** lamp is turned on if a shot is acquired while the program makes a status request. The **armed** lamp below the trigger lamp lights up when the transient recorder is waiting for the next trigger event while the program makes a status request.

26. Use the **Exit** switch to stop the program.

That's the end of the short introduction into the capabilities of the data acquisition software TCPIP Track.

6.2.2 TCPIP Live Display

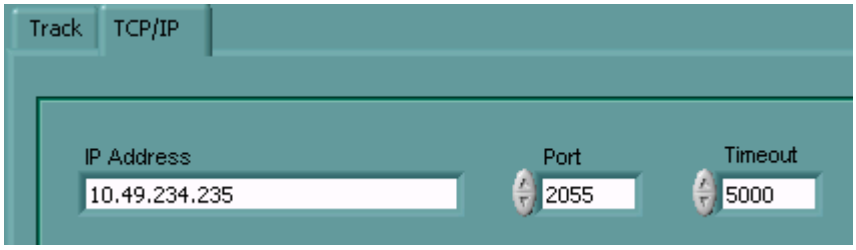
TCPIP Live Display allows you to operate the transient recorder in an oscilloscope mode, where the display is updated every X number of shots. This mode is very useful when you are trying to align the optics on your system and would like to see how the changes affect the signal. To load TCPIP Live Display, either double click on the `TCPIP-LiveDisplay.llb` or open the file `TCPIP Live Display.vi` in the `TCPIP-Live Display.llb`. If you installed the Windows application please start the program by selecting the corresponding entry in the Licel section of the Windows Start menu. After this the following interface appears:



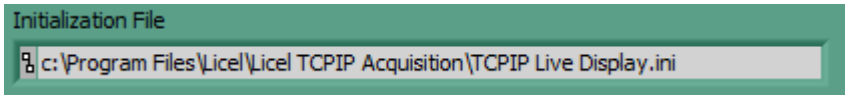
As you can see, the interface is quite similar to that of TCPIP Track. The main difference is that the button controls start, stop, show, single shot, continue and save buttons are missing. Instead, there is an **update #** control and a second graph called **mean2**.

1. First of all the **IP Address** and **Port** have to be set. You should already have set these values for the Licel Ethernet Controller following the network setup section above.

   - Using the LabVIEW vi, just enter the required values on the *TCP/IP* page and save them as defaults.

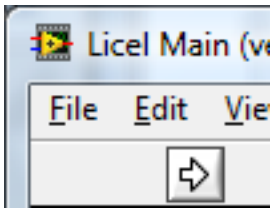


   - If you use the Windows application you must directly enter the correct values into the corresponding control fields on the *TCP/IP* page. On exit the values will be saved to the initialization file in the case that the TCP/IP connection is established.
     You may also set the values in the initialization file `Licel Main.ini`. You will see the full path of the file in a file path indicator on the *TCP/IP* page.



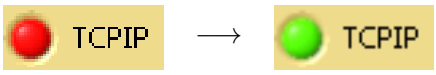
   - If you run `TCPIP Live Display` within a sub panel on a page from `Licel Main` the latter is responsible for managing the TCP/IP connection.

2. To start the program press the **Run** button at the top left of the screen.



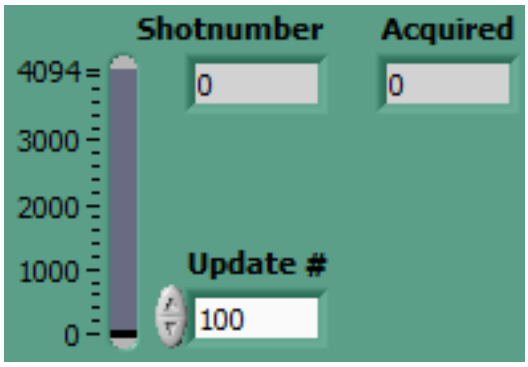
The Windows application will start automatically when called for the first time.

3. After a short time the **TCPIP** indicator should change its color from red to green indicating a successful connection with the Licel Ethernet Controller. If the indicator remains red and/or an error is indicated, please check the values for **address** and **Port**, change them (on the program's panel or in the initialization file) if necessary. Check if the Licel Ethernet Controller is running and that all network connections are correct. The second TCPIP indicator should change to a green color after some time, too. This second connection is used to directly transfer data from the transient recorders to the acquisition computer if the **update #** is smaller than 15.



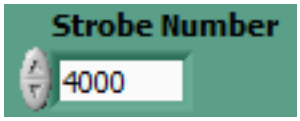
4. Notice that the **Shotnumer** indicator immediately starts to increase. When **Shotnumber** is equal to **update #**, the signal data is read from the transient recorders and displayed.

5. Set the **update #** to 100.



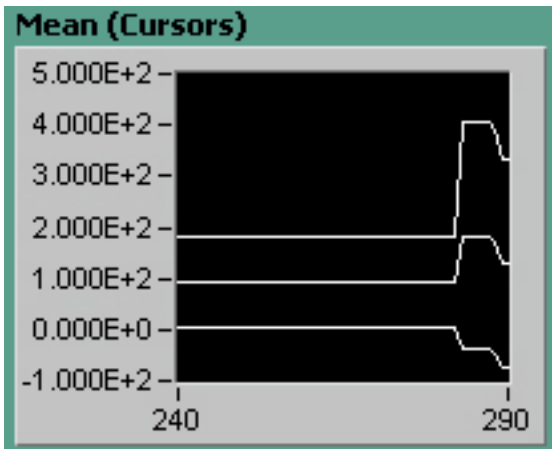
If your trigger is operating at 10Hz, the data display will now be updated every ten seconds. By changing this value, you decide how many shots will be taken between subsequent updates of the display. If the shot number is set a value larger than 4094 the transient recorder's data is read when reaching a multiple of 4094 and immediately displayed. The transient recorder is restarted, and the next acquired data is added until the target shot number is reached. The shot number corresponding to the last displayed data is shown in the **Acquired** field.

6. Set the Strobe Number to 4000



notice that the length of the signal on the x-axis is now 4000. This indicates that only the first 4000 bins of the transient recorder memory are being read out and displayed.

7. Move the cursors to select part of the signal on the waveform graph. The region between the two cursors is averaged and displayed in **Mean (Cursors)**. Additionally the +/- one standard deviation lines are shown.



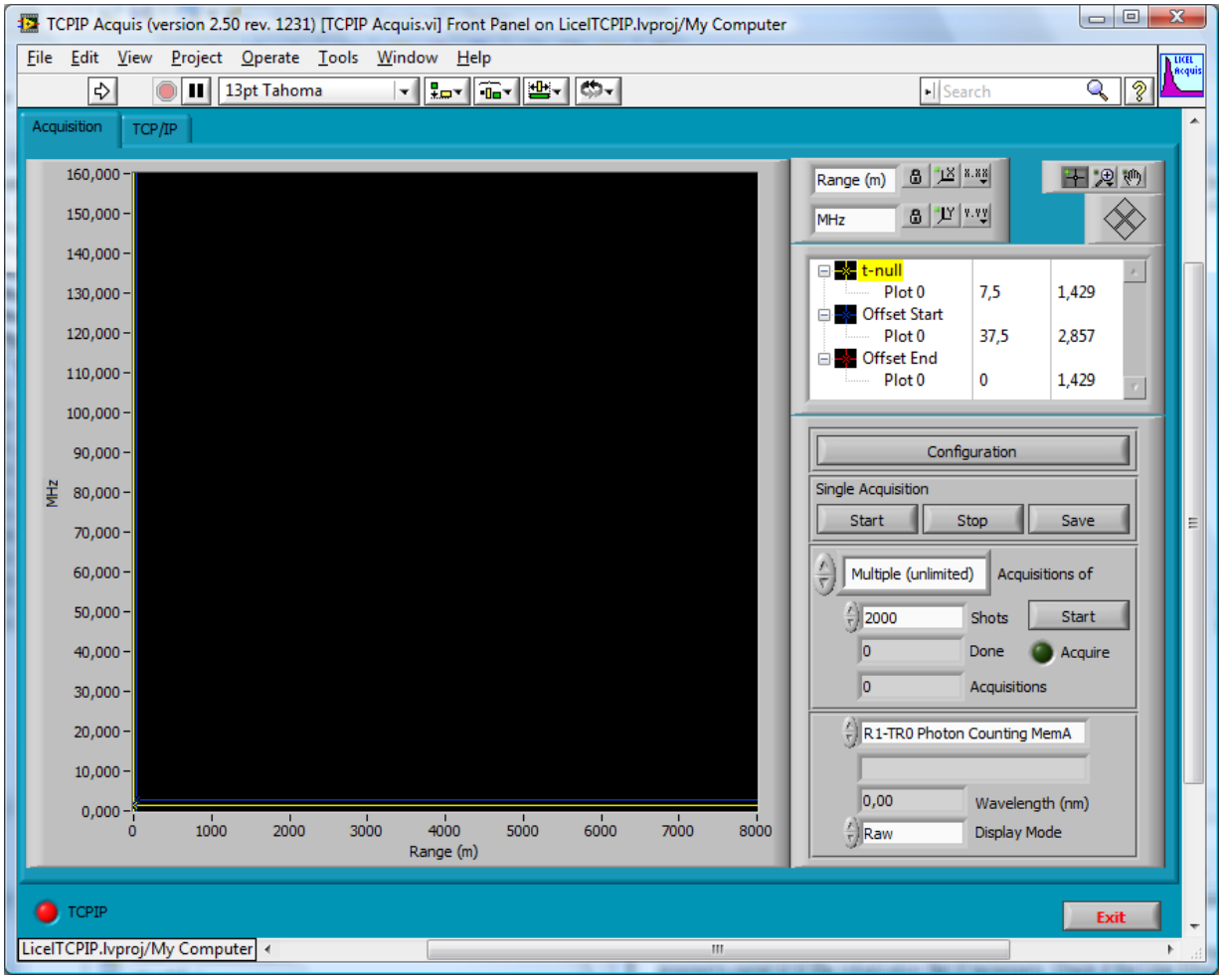
If you have questions about the other controls, please consult the previous section about TCPIP Track.

6.3 Acquisition Software

In this section, you will be introduced to the TCPIP Acquis software module. Before starting acquisitions, you should configure the default global information to correspond to geographical details of your location, so that this information can be properly included into the headers of the data files. After that, you will set up the Transient Information which is specific to your data sets. If you have already aligned these parameters, you can directly jump to the TCPIP Acquis section to make an acquisition. Please note that `TCPIP Acquis` uses the initialization file `acquis.ini`.

6.3.1 Starting TCPIP Acquis

To load the acqusition program TCPIP Acquis, either double click on the `TCPIP-Acquis.llb` or open the file `TCPIP Acquis.vi` in the `TCPIP-Acquis.llb`. If you installed the Windows applications please start the program by selecting the corresponding entry in the Licel section of the Windows Start menu.
The Front Panel of TCPIP Acquis is seen in the next picture:



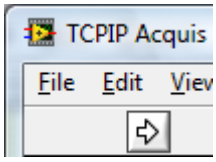
1. First of all the **IP Address** and **Port** have to be set. You should already have set these values for the Licel Ethernet Controller following the network setup section above.

   - Using the LabVIEW vi, just enter the required values on the *TCP/IP* page and save them as defaults.

- If you use the Windows application you must directly enter the correct values into the corresponding control fields on the *TCP/IP* page. On exit the values will be saved to the initialization file in the case that the TCP/IP connection is established.
  You may also set the values in the initialization file `acquis.ini`. You will see the full path of the file in a file path indicator on the *TCP/IP* page.

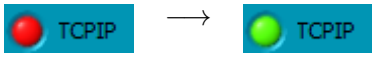
- If you run `TCPIP Acquis` within a sub panel on a page from `Licel Main` the latter is responsible for managing the TCP/IP connection.

2. To start the program press the **Run** button at the top left of the screen.



   The Windows application will start automatically when called for the first time.

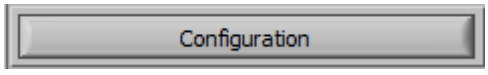
3. After a short time the **TCPIP** indicator should change its color from red to green indicating a successful connection with the Licel Ethernet Controller. If the indicator remains red and/or an error is indicated, please check the values for **address** and **Port**, change them (on the program's panel or in the initialization file) if necessary. Check if the Licel Ethernet Controller is running and that all network connections are correct. The LED of the transient recorder should be lit up.



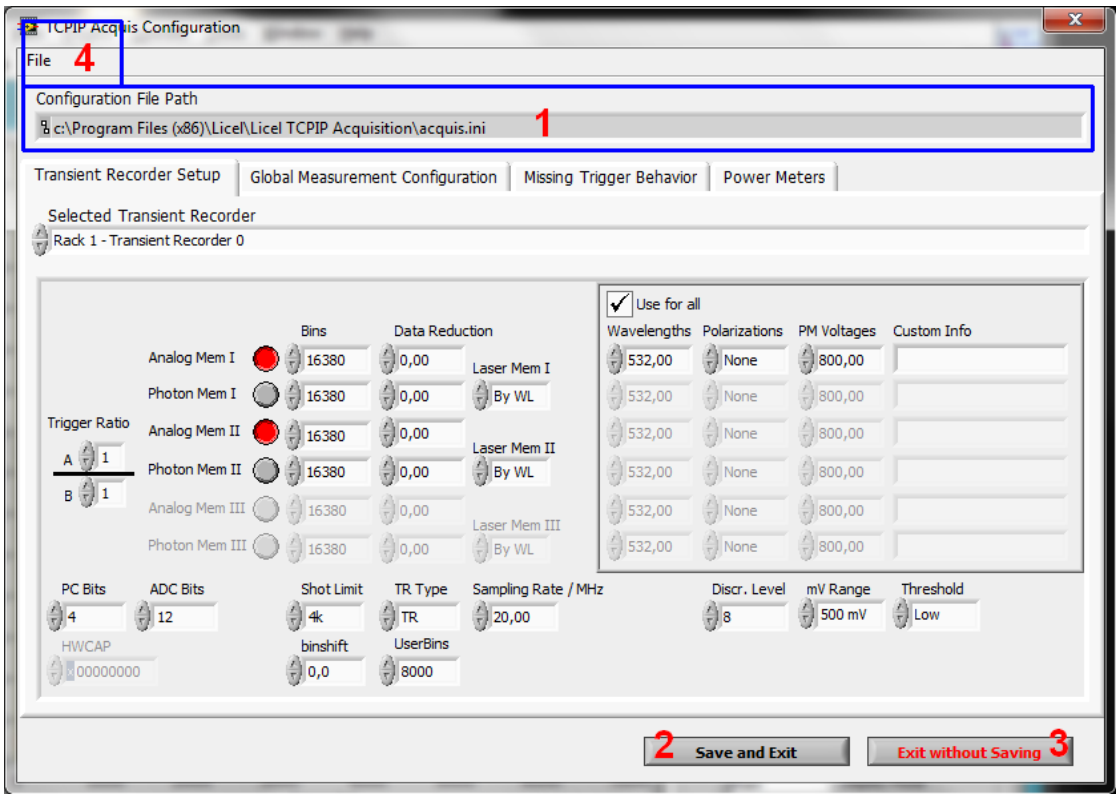
If the program is already running and not acquiring data then continue. Otherwise press the **Stop** button and then you may continue with the following steps.

6.3.2 The Configuration Dialog

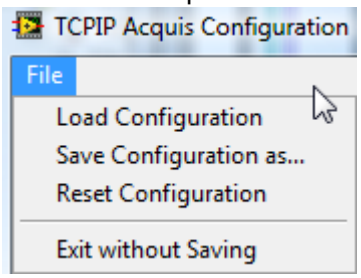
The configuration parameters concerning the transient recorders, certain global parameters, the program behavior in the case that no trigger is received, and the power meter integration and are set in a separate program dialog. This configuration dialog is accessed by pressing the *Configuration* button you find on the front panel of the on the right-hand side.



The configuration dialog will show the information found in the file `acquis.ini`. If a value is not found it will be set to a default value.
The configuration dialog is organized in terms of four tab pages each of them corresponding to one of the following sub-subsections.



1. The full path of the initialization file is displayed at the top.
2. Use *Save and Exit* to close the configuration dialog and to save all configuration data to the configuration file.
3. Use *Exit without Saving* to close the configuration dialog without saving the configuration. Please note that when you exit the program without saving, any unsaved data is lost! Thus if you have configured the data and wish to keep it, you need to choose *Save and Exit*.
4. The *File* menu provides the following entries:

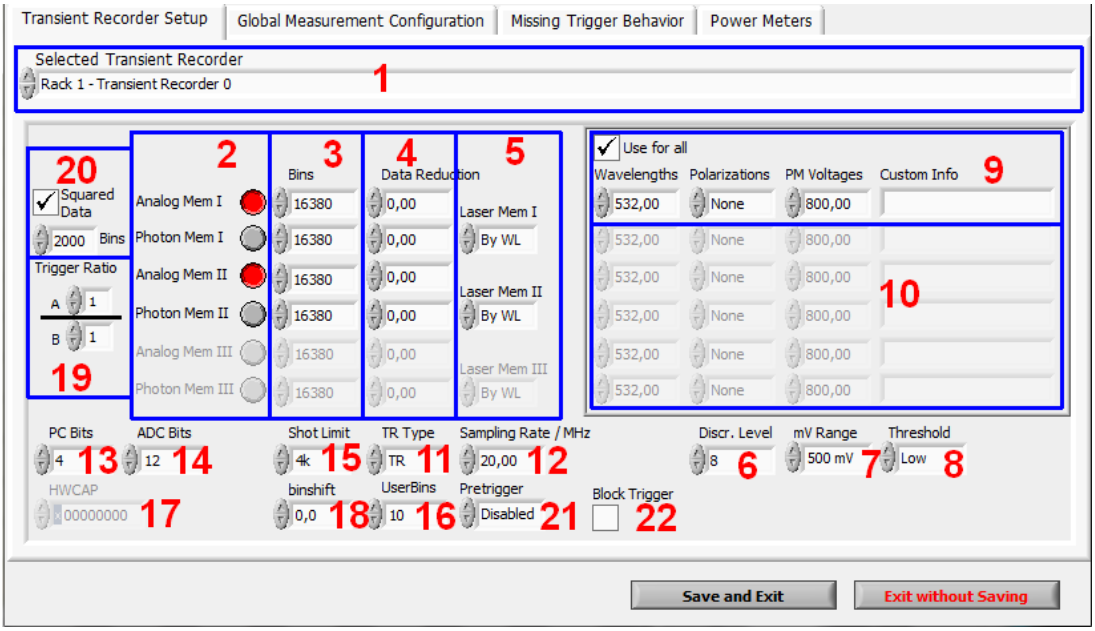


   - *Load Configuration* allows to select a configuration file anywhere in the file system to load the configuration parameters from there.
   - *Save Configuration as...* saves all configuration parameters to a selectable file in the file system.
   - *Reset Configuration* resets all changes since the configuration dialog has been opened.
   - Exit without Saving closes the configuration dialog without saving the configuration. Please note that when you exit the program without saving, any unsaved data is lost! Thus if you have configured the data and wish to keep it, you need to choose *Save and Exit* from above.

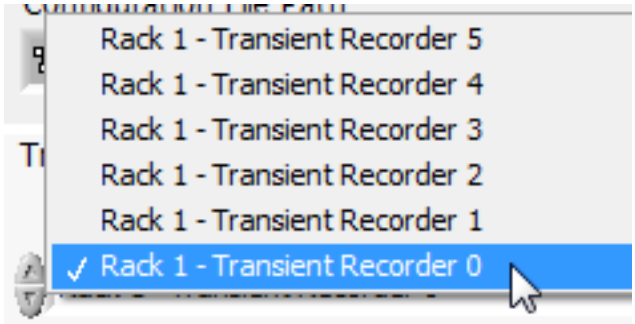
Please note: If you want the current configuration to be used as the default configuration for the next start of the program, you must save this data as `acquis.ini` in the directory where TCPIP-Acquis.llb or the Windows applications, respectively, are located.

Transient Recorder Setup

The *Transient Recorder Setup* is displayed on the corresponding tab page and contains all the information needed to configure the transient recorders. Several properties of the individual installed transient recorders can be set here. Some of the properties have fixed settings dependent on the capabilities of the transient recorder and the Licel Ethernet Controller. In such a case the corresponding input field will not be shown or is disabled and greyed out.



1. A selection list allows to select one of the installed transient recorder for configuration.



   Please completely configure a selected transient recorder before switching to the next one. If you call the configuration dialog from Licel's multi-rack acquisition software *M-Acquis* the list entries easily allow to identify the transient recorder's racks.

2. Select the memory banks for data transfer. The memory banks are Analog Memory A, Photon Memory A, Analog Memory B, and Photon Memory B. Analog Memory C and Photon Memory C are enabled if supported. For a PRxx-xx transient recorder (**TR Type**=*PR*, see position 11) use the photon counting switches, only.

3. Position 3 allows to specify the number of bins to read. The maximum number of bins is given by $6530/(2^{\text{data reduction}})$

4. Set the data reduction which allows for (software-) binning. A data reduction level of 0,1 and 2 corresponds to a height resolution of $1\times$, $2\times$7.5m, and $4\times$ the length corresponding to a primary bin. For a 20 MHz transient digitizer these values correspond to 7.5 m, 15 m, and 30 m, respectively. Each increasing in value reduces the height resolution by 1/2 and doubles the number of bins that are combined together to make a superbin. Thus the levels 0,1,2 correspond to 1 , 2, and 4 bins per data point, respectively.

5. Assignment of the used laser to a memory. *By WL* will assign the laser by using the wavelength specified in regions 9 and 10. The further selectable list entries correspond to the lasers in the global configuration.

6. Set the discriminator level for the transient recorder. There are 64 discriminator levels (values 0 – 63) which correspond to either a range of 0 – 24 mV without gain reduction or 0 – 96mV with gain reduction.

7. Set the range value of the transient recorder. Valid values are 0 – 20 mV, 0 – 100mV and 0 – 500mV.

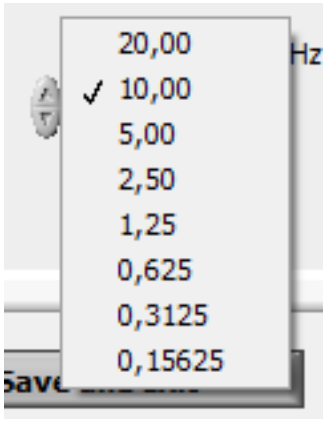
8. Setting the threshold mode is possible at position 8. In the *High* level the discriminator level is set four times as high.

9. The parameters in region 9 allow to enter further parameters which indicate the type of equipment that is used in the channel for memory A, analog aquisition. This information is stored as a header in the data files, so that the user (or whoever has to evaluate the data) can see what parameters were used to take the data. The fields are used for the laser **Wavelength**, the corresponding **Polarization**, and the photomultiplier voltage (**PM HV**). **Polarization** has the allowed values *None* (0), *Parallel* (1), *Crossed* (2), *R*ight *Circ*ular (3), and *L*eft *Circ*ular (4). **Custom Info** can be used to specify any further channel dependent paramters. The information entered in these fields has no effect whatsoever upon the data acquisition. It is used purely to store information about the experimental setup in the data files. The content of **Custom Info** will be saved enclosed by quotes as entered by the user. If the checkbox **Use for All** is checked the values for memory A, analog are used for all other acquisition channels. If not, the input fields in region 10 allow to enter indiviual values.

10. **Wavelengths**, the corresponding **Polarizations**, **PM Voltages**, and **Custom Info** for the memory A, photon counting, memory B, analog and photon counting, and memory C (if available), analog and photon counting channels. The values are set equal to those of region 8 if **Use for All** is checked. If not the input fields can be used to enter individual values.

11. The **TR Type** is the transient recorder type. The transient recorder is either a TRxx-xx (analog and photon counting transient recorder) or PRxx-xx (pure photon counting device), **TR Type** must be set to *TR* or *PR*, respectively. Align the memory switches (position 2) to the **TR Type**.

12. The **Sampling Rate** is the sampling rate of the transient recorder. This value defaults to 20 MHz. If a transient recorder does not support the *frequency divider* feature (`HWCap ∧ 0x40`, see below) the sampling rate is a fixed property: if your transient recorder has a different sampling rate, you will need to change the value to the corresponding sampling rate. For instance enter *40* for a TR-**40**-xxx or *10* for a TR-**10**-xx. New transient recorders will report their sampling rates, then the corresponding value is not changeable. For a transient recorder supporting the *frequency divider* feature the **Sampling Rate** is a list of selectable values.



13. The number of photon counting bits **PC Bits** must be specified here according to the transient recorder. This value is automatically set if supported by the controller. Then this field cannot be changed manually.

14. The number of ADC bits **PC Bits** must be specified here according to the transient recorder. This value is automatically set if supported by the controller. Then this field cannot be changed manually.

15. The **Shot Limit** (*4k* = 4094 shots or *64k* = 65534 shots) is set here. The shot limit must be the same for all transient recorders.

16. The input field to set the **UserBins** is visible only if the corresponding feature is supported by the transient recorder.

17. The non-changeable **HWCAP** field indicates individual transient recorder capabilities. **HWCAP** equals zero for older transient recorders. The following transient recorder capabilities are coded:

    `0x01` separate shot counter B
    `0x02` separate shot counter C
    `0x04` separate shot counter D
    `0x08` pretrigger
    `0x10` memory blocking
    `0x20` squared data support
    `0x40` frequency divider

18. If supported by the controller and the transient recorder the **binshift** is read from the active transient recorder and the value cannot be changed in the configuration dialog. Otherwise the value is changeable. The binshift is saved to the data file in the variable lines.

19. The **Trigger Ratio** is only available if acquisitions are enabled for both memories, A and B and no sparate shot counters are available. Then the ratio of the trigger frequencies A and B must be set according to the ratio of the trigger frequency inputs for memories A and B at the corresponding transient recorder.

20. The **Squared Data** with corresponding **Bins** is available if the individual hardware capability `0x20` is available and squared data handling is supported by the controller. If **Squared Data** is checked the summed squared counts will be read for active analog and photon counting memoryA-channels. The data will then be saved as $\sqrt{N\sum(x^2)-(\sum x)^2}$ (N: shots, x: counts in each bin). Reading this enables to calculate the standard deviation and standard error. In TCPIP Acquis the standard error is shown. Refer to the appendix for calculation details.

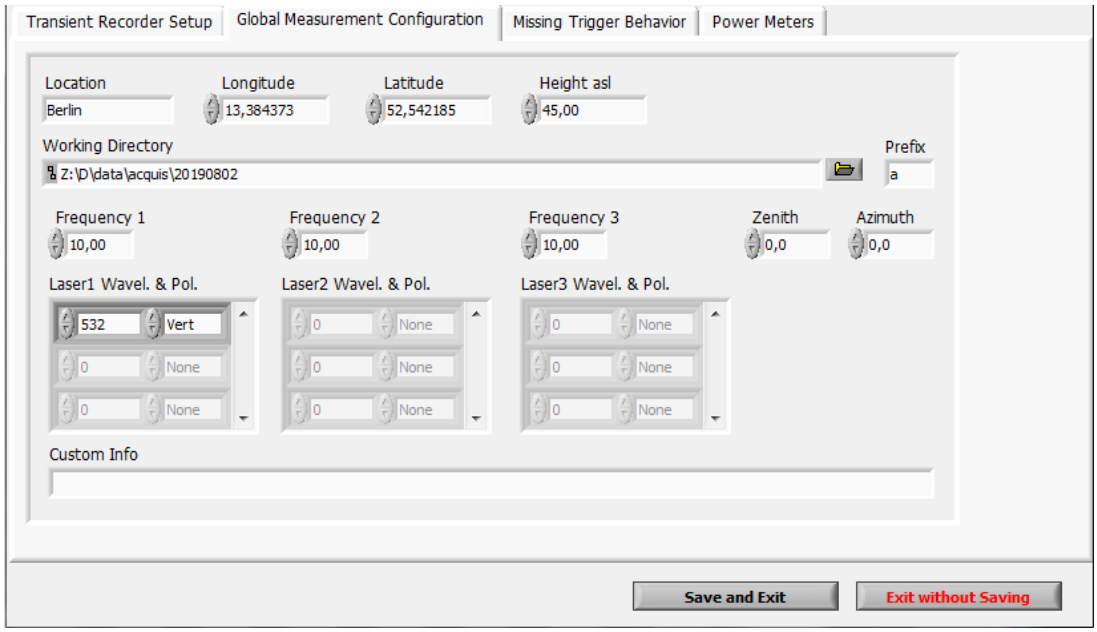
21. The **Pretrigger** is available if the individual hardware capability `0x08` is available and the pretrigger is supported by the controller. If this is checked, a trace where 1/16th of the transient recorder memory tracelength will be before the trigger and the remaining points will be after the trigger point. For TR with 16k memory configuration to store 15 km after the trigger with 40MHz, you will need 15km / 3.75 = 4000 bins + 1024 bins pretrigger = 5024 bins. For 16bit TR shipped before 2018 the pretrigger is 128 bins long. Newer units have the scaled length of the pretrigger.

22. The **BlockTrigger** is available if the individual hardware capability `0x10` is available and the memory blocking command is supported by the controller. When checked the on-used triggers are blocked during an acquisition.

If you have filled the control fields above with appropriate values, your transient recorders should now be configured for use with TCPIP Acquis.

Global Measurement Configuration

The global information allows you to set values that are stored in the data file headers which will tell you later about what sort of conditions were existent at the time the data was acquired. These are global values which usually do not vary from measurement to measurement and so it is named *Global Measurement Configuration*. The *Global Measurement Configuration* is displayed on the corresponding tab page.



1. The **Location**, **Longitude**, **Latitude**, and **Height asl** (above sea level) represent the location of your acqusition system and will be stored in the data file headers.

2. The **Working Directory** is the location where you want data files to be stored and the **Prefix** contains one or two letters that will be used as a prefix for the file names. Directly enter the path of the **Working Directory** into the control field or browse your file system using the browse button. The format of the file names is
   `?YYmddHH.MMSSuuu`
   where `?` is the **Prefix**, `YY` is the year of the century, where `m` is the month (hexadecimal, 0 – C), `dd` is the day of the month, `HH` is the current hour of the day, `MM` are the minutes, `SS` the seconds, and `uuu` the first 3 decimal places of the seconds.
   For example the filename

`a1981211.2816504`
is a file that would have been taken on August 12, 2019. The operator set `a` to be the first letter (as in the screenshot) and the time was 11:28:16.504.

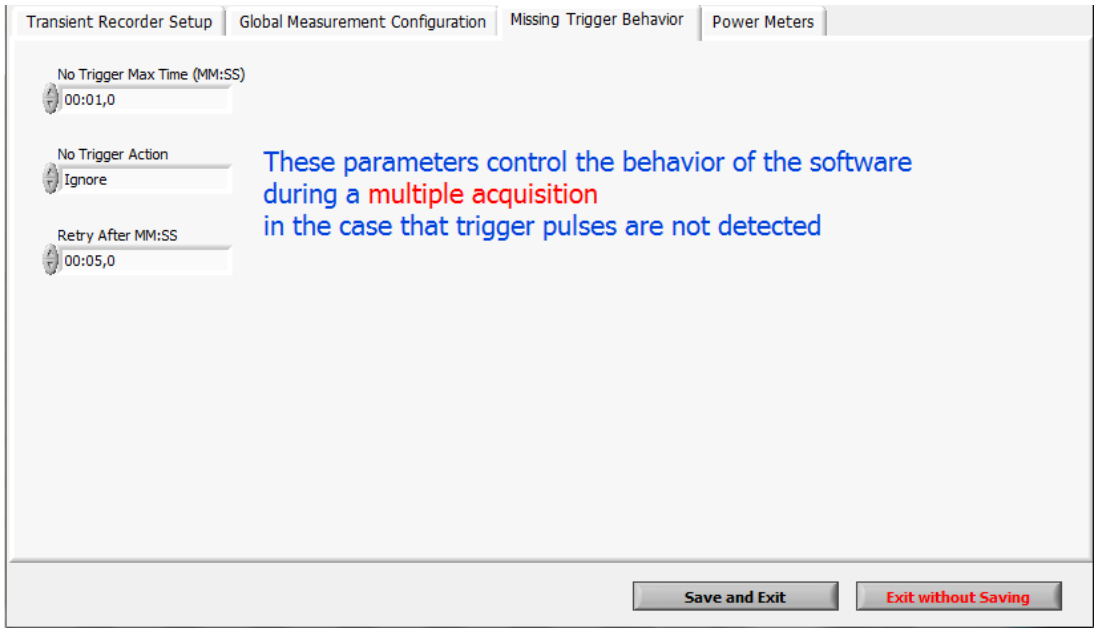
3. The **Zenith** and **Azimuth** angles are the angles of the used telescope/sending and receiving optics. They are stored as header information in the data files.

4. The next group of input fields contains the repetition rates and wavelengths with polarizations of the lasers (**Frequency1**, **Laser 1 wavel**ength, **Frequency2**, **Laser 2 wavel**ength), and **Frequency3**, **Laser 3 wavel**ength. At least one wavelength and polarization for laser 1 should be set. The allowed polarization values are *None* (0), *Vert*ical (1), *Hor*izontal (2), *R*ight *Circ*ular (3), and *L*eft *Circ*ular (4).

5. **Custom Info** is a free text field which is enquoted and attached to the second line of the data files.

Missing Trigger Behavior

The parameters to respond to missing trigger pulses during an acquisition are settable on the tab page *Missing Trigger Behavior*.



The following parameters may be set here:

1. **No Trigger Max Time** Maximum time the software accepts missing trigger pulses. When this time has elapsed during an acquisition (*single or multiple*) the **No Trigger** LED in the acquisition program will become visible. When this time has elapsed during a *multiple acquisition* the **No Trigger Action** will be processed.
   The value is set in the format `MM:SS.u` with the minutes `MM`, the seconds `SS`, and the one-digit fractional part of the seconds.

2. **No Trigger Action** Action to apply after the specified **No Trigger Max Time** has elapsed without having received a trigger during a multiple acquisition.

   (a) *Ignore* The missing trigger is ignored if the already acquired shots are less than (target **Shots** -32). Otherwise the running acquisition is stopped, data is saved with the currently acquired shots, and a new acquisition is started.

(b) *Stop – NoSafe* The running acquisition is stopped without saving data.

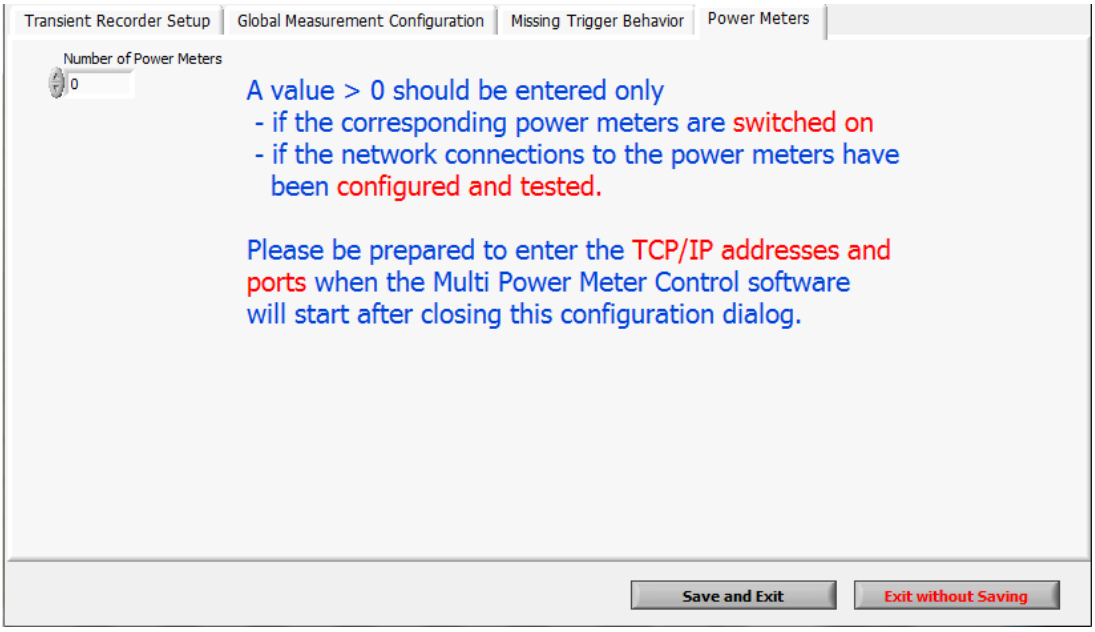
(c) *Stop – NoSafe – Retry* The running acquisition is stopped without saving data. After the **Retry After** time a new acquisition is started.

(d) *Exit – NoSafe* The running acquisition is stopped without saving data. The acquisition program is terminated.

3. **Retry After** Restart an acquisition when the specified time has elapsed after the previous acquisition has been stopped in the case that the **No Trigger Action** is set to *Stop- NoSafe - Retry*
   The value is set in the format `MM:SS.u` with the minutes `MM`, the seconds `SS`, and the ono-digit fractional part of the seconds.

Power Meters

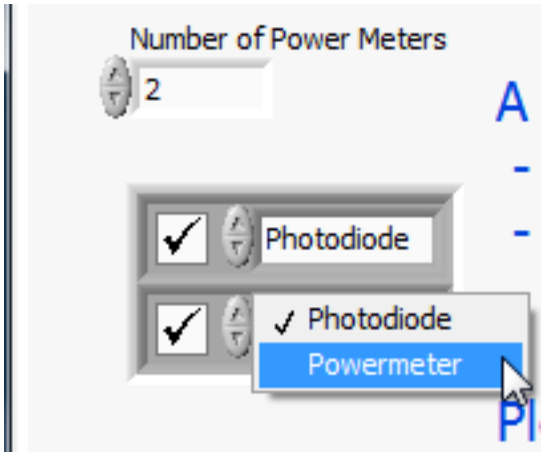
The Acquisition software is capable to start and communicate with 1 or more instances of Licel's `Power Meter Control` software and to save the power meter data together with the transient recorder data.



1. First of all it is required to configure the TCP/IP connection from the PC to the Power Meter Controller hardware. Please refer to the network setup and make sure that each IP address is uniquely used.

2. Please test the connection to each Power Meter Controller using the `Power Meter Control` software.

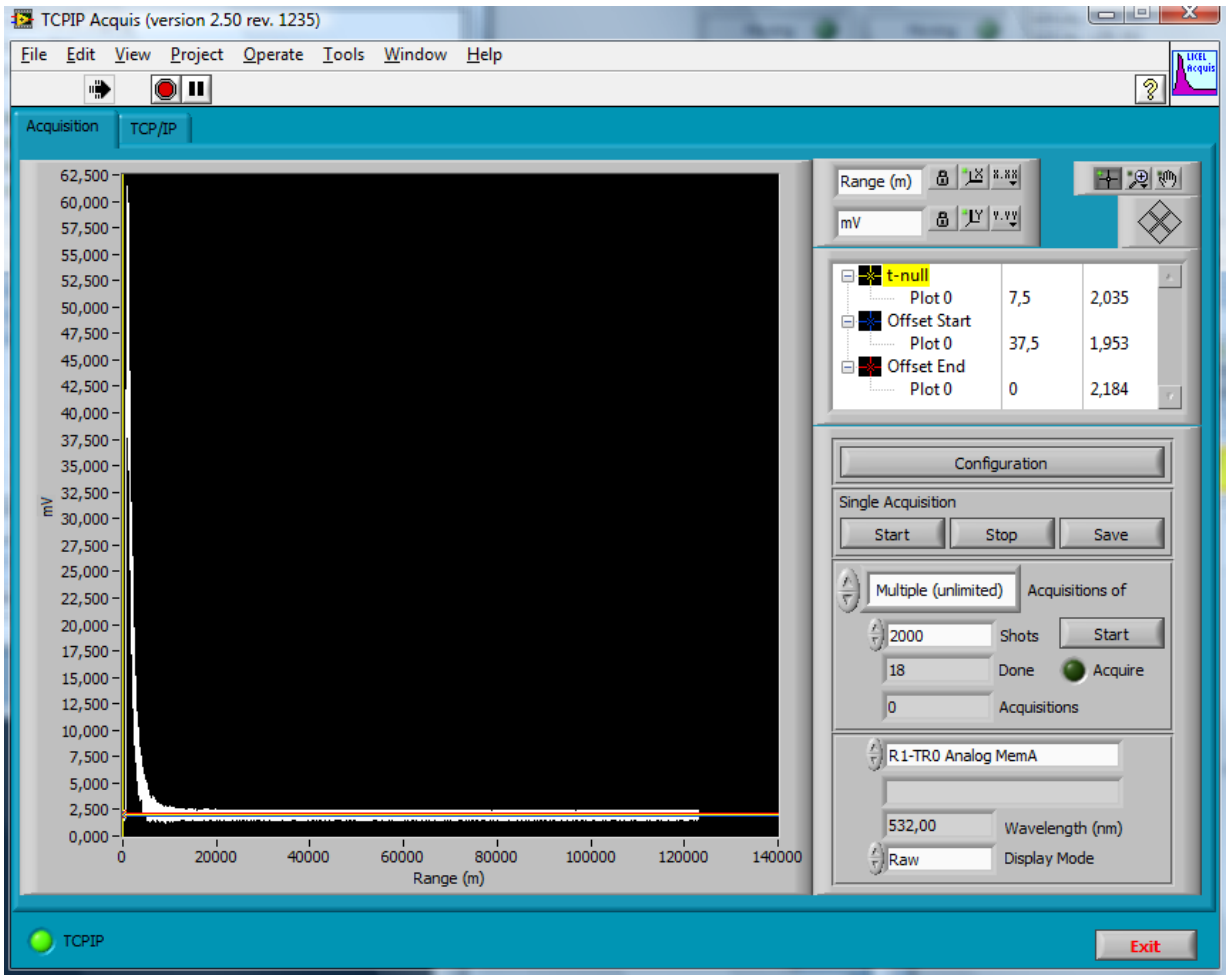
3. Top-right you may specify the **Number of Powers** you will use. After entering a number > 0 you may set further parameters for each power meter (**active** (checkbox) and **Sensor**).



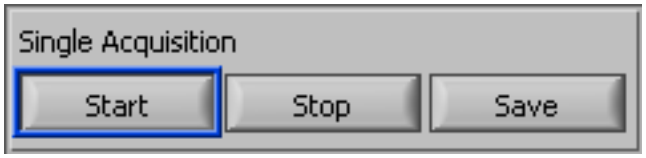
4. After leaving the configuration dialog the `TCPIP Acquis` will prepare an initialization file for the required software `Multi Power meter Control`, and the latter will be started. If it is started for the first time **you will have to enter the IP addresses and ports of the involved Power Meter Controllers**. Please refer to the subsection 7.3.3 for the details.

6.3.3 Acquisitions with TCPIP Acquis

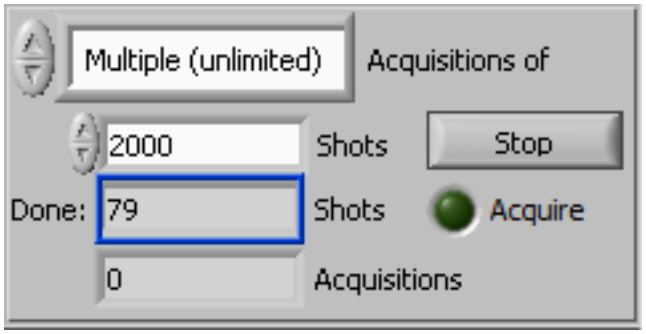
Now that the configuration is complete, it is time to take the first acquisitions with TCPIP Acquis. The Front Panel of TCPIP Acquis is shown again:



1. Press the **start** button in the **Single Acquisition** section of the right-hand side panel.



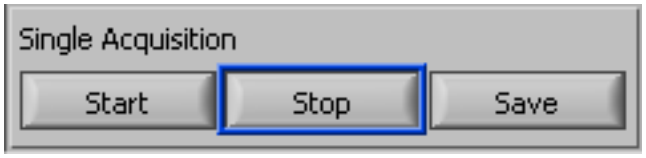
The transient recorders, in which you have activated data sets, should now acquire data if a sufficient trigger signal is connected to the input. If the acquisition has been started, the **number of shots done** in the **Multiple Acquisitions** sections of the right-hand side panel should start increasing. Please note that the **Acquire** Indicator is for multiple, automatically saved aquisitions, only.



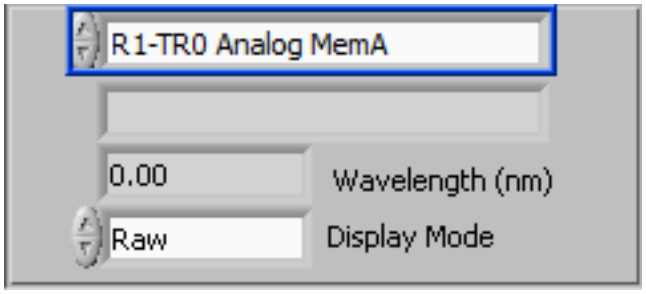
If the acquisition software does not receive an increasing shot number for a time larger than the **No Trigger Max Time** specified in the configuration an LED indicator becomes visible at the top right corner of the front panel window:

The data acquisition of the individual transient recorders can be checked by seeing if the Acquire LED of the specified transient recorder is brightly lit up. If not, the trigger is either insufficient, or the data sets are not activated as described in the section Changing the Transient Recorder Information.

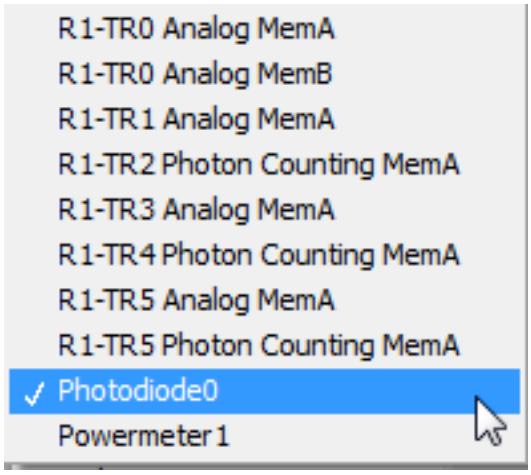
2. Stop the acquisition in the same group by pressing the corresponding button.



3. By changing the **data set selection**, you can now view the various data sets that were acquired. The entries in the selection list contain information about the used transient recorder (with it's rack), the acquisition mode (analog/photon counting) and the read memory. The wavelength is indicated in a seperate field.

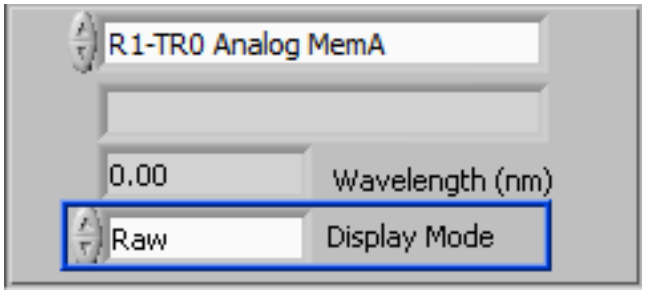


4. In the case that power meter data is acquired, as well, the list contains the corresponding entries at the end:

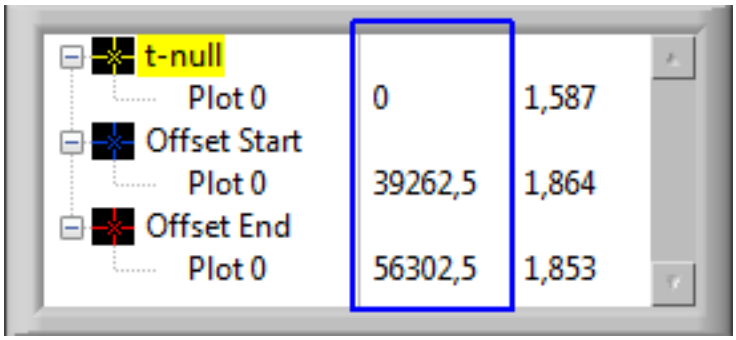


5. The horizontal scale of the data display is given in meters. The vertical scale is in mV for the analog and in MHz for the photon counting data. An incorrect scaling of the photon counting data might be the result of a wrong entry for the *Sampling Rate* in the transient recorder configuration (Configuring the Transient Recorders in section 6.3.2).

6. The **Display Mode** setting allows you together with the positions of the cursors to view the data in raw format (*Raw*), base line-corrected (*Offset Corr.*), or corrected to match the power loss due to the length of travel of the signal (*PR2*).

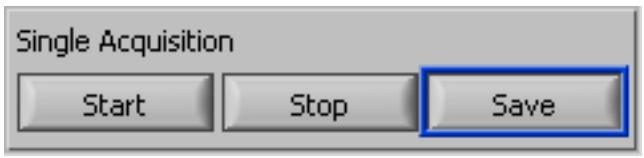


7. The background for the baseline correction is defined by the cursors *Offset Start* and *Offset End*, while the point-of-zero for the *PR2* mode corresponds to the cursor *t-null*. These values may either be set by moving the cursors in the graphic or by entering the appropriate values in the cursor control menu:

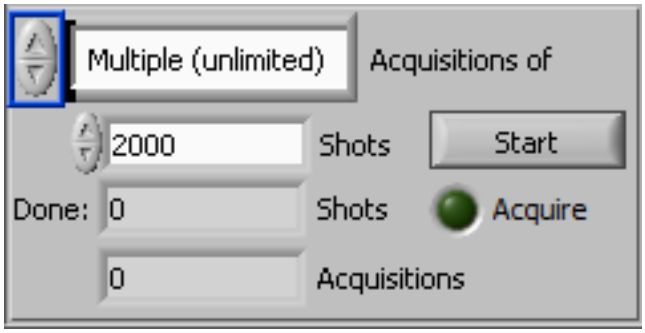


Please refer to the Advanced Viewer section for a descriptive example.

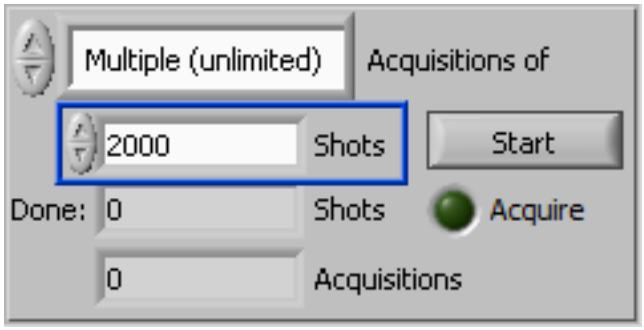
8. If you are satisfied with the data press **save**. The file is now renamed from temp.dat to a unique identifier according to Licel's data file format. The file contains the raw, uncorrected data as acquired by the transient recorders.



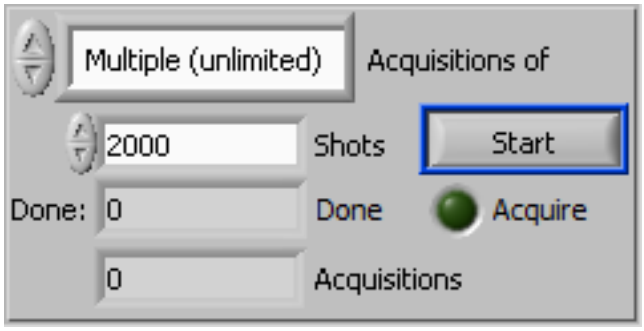
9. In order to automatically make multiple acquisitions, you must first decide whether to make acquisitions until the process is manually stopped or to enter the desired number of **Acquisitions**. Each acquisition will contain the specified number of **Shots** and will be saved to a separate file.



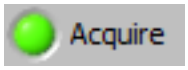
10. Then you will have to set the number of **Shots** which will be acquired for each file.



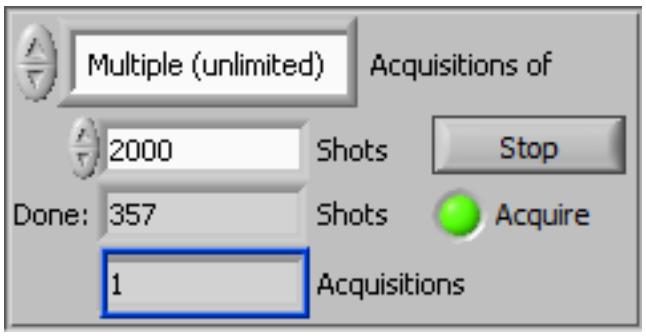
11. To start an automatic series of acquisitions, press the **Start** button in the multiple acquisition group. This button then turns into a **Stop** button.



The LED Indicator **Acquire** will change to light green. This indicator is for multiple acqusiitons, only.



The number of shots **done** will now start increasing and when the number *done* is equal to the number of **Shots**, the data sets will be written to a file. After this, the acquisition counter will increase and the program automatically starts acquiring the next data sets.



12. This process of automatically acquiring data sets of the defined number of **Shots** continues until you press the **Stop** button or — if specified — the number of acquisitions has been reached. In the case that acquiring data is terminated by pressing the **Stop** button the data acquired up to that time will be saved in a last file.

    If the acquisition software does not receive an increasing shot number for a time larger than the **No Trigger Max Time** specified in the configuration an LED indicator becomes visible at the top right corner of the front panel window:

    In this case the action specified in the configuration will be processed.

13. You can change the data set configuration or global information by pressing the **Configuration** button. The changes to the data set configuration are applied to any acquisitions that you make after changing the configuration.

14. If an overflow at the ADC has been detected an overflow LED appears red at the top of the front panel window.

15. If you are done taking data and want to leave the program, press the **Exit** button.

You should now have a first impression of the capabilities of the Licel data acquisition software and the capabilities of the transient recorders. You can use all vi's as raw material for your acquisition software.

6.3.4 Monitoring and Controlling TCPIP Acquis from Outside

TCP/IP Server

The basic functions of the TCPIP Acquis software can be accessed from third party applications via TCP/IP. For this TCPIP Acquis implements a TCP/IP Server listening on a defineable port (initialization file).
Please refer to the appendix Controlling TCPIP Acquis via TCPIP for the configuration of the TCP/IP server and the syntax of the supported commands.

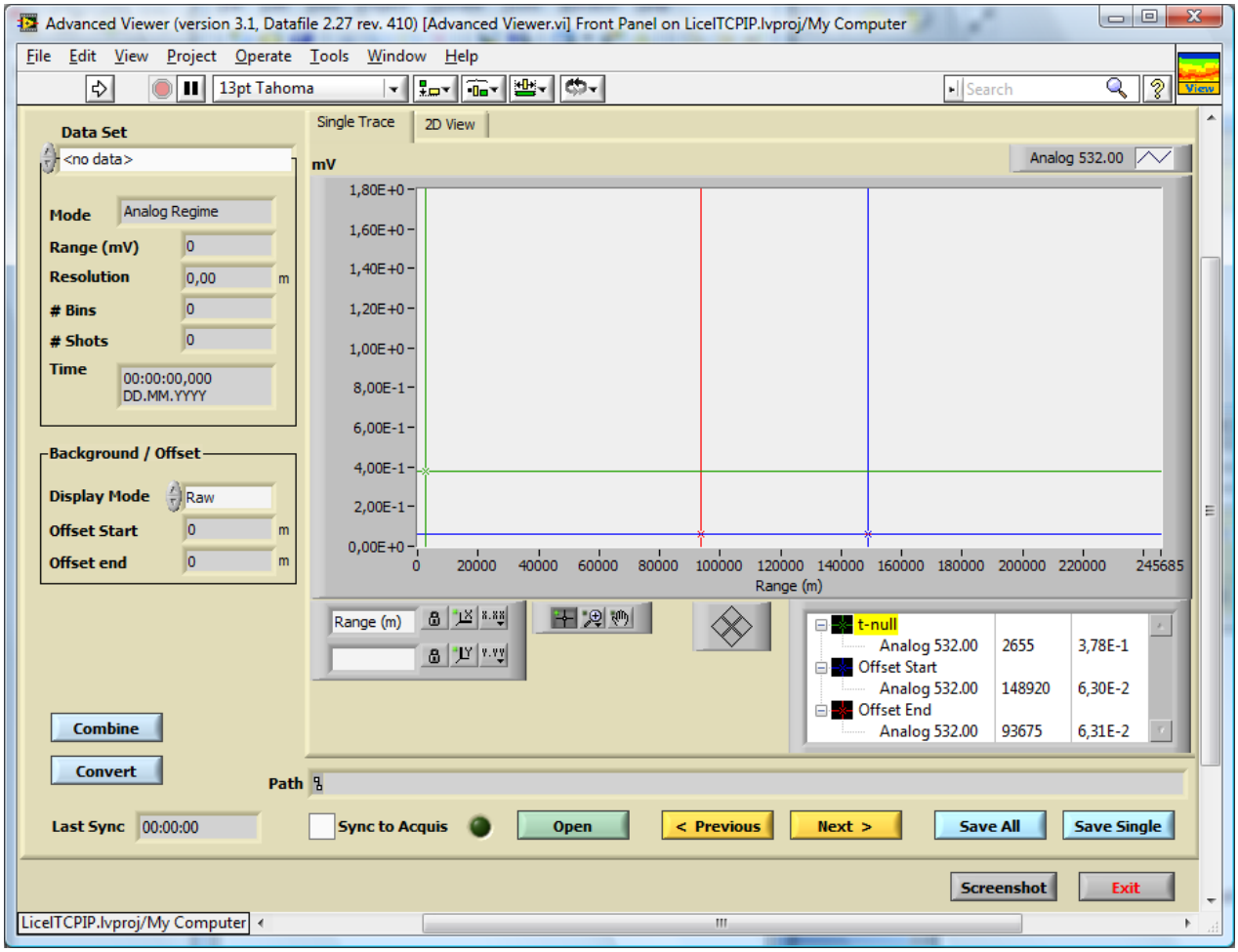
Queue Control

The basic functions of the LabVIEW version of the TCPIP Acquis software can be accessed from third party LabVIEW VIs using LabVIEW's named queue mechanism. Since version 2.31 TCPIP Acquis uses a listening queue named `ACQUIS_LISTEN` to accept commands, and a reply queue

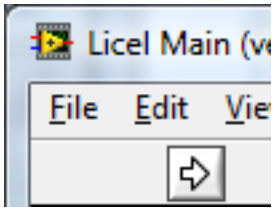
ACQUIS_REPLY to send answers to the commands received via the listening queue. If your TCPIP Acquis is controlled by this queue mechanism please remember to wait for the reply to the command you sent.
Please refer to the appendix Controlling TCPIP Acquis from Outside for the syntax of the queue commands.

6.4 Advanced Viewer

The Advanced Viewer is designed for reading data files created by Licel acquisition software like TCPIP Acquis. In order to demonstrate the capabilities of this program, you must already have acquired at least one dataset and saved it. For more information about acquiring data, please refer to the manuals for Acquis, Track and Live Display. After opening the Advanced Viewer.vi located in the Advanced Viewer.llb, you should see the following screen:

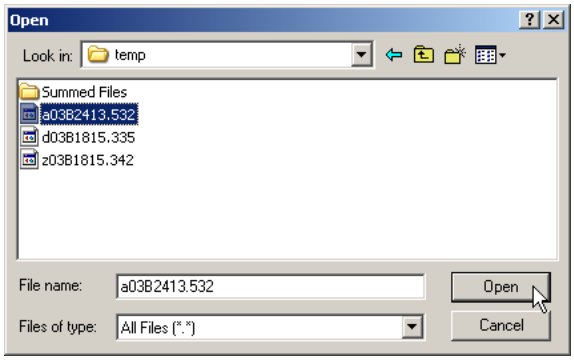


To start the program press the **Run** button at the top left of the screen.

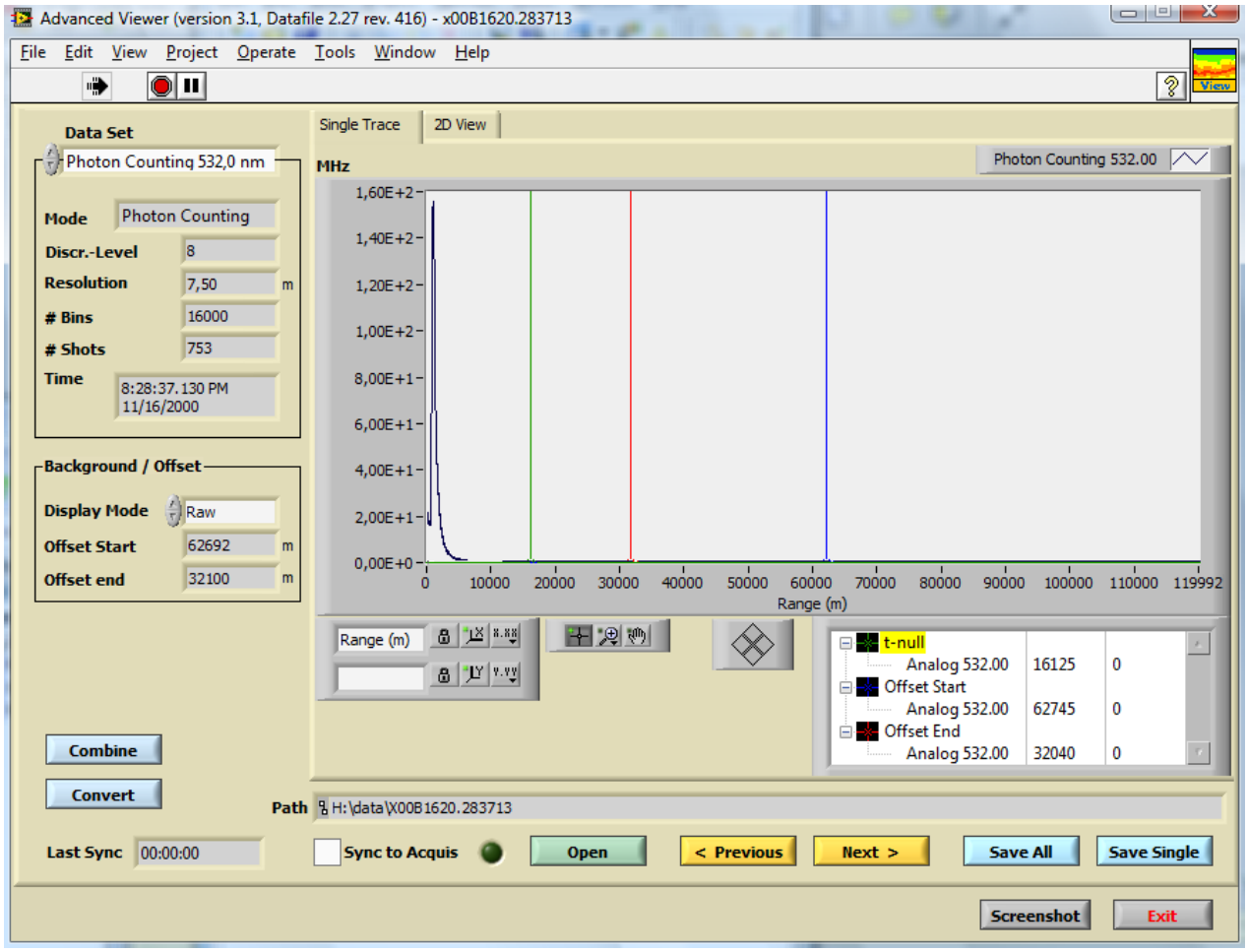


The Windows application will start automatically when called for the first time.
A desired data file may be loaded by pressing the **Open** button.

A file selection dialog will appear. At the program start this dialog will come up without pressing the button.



Select a valid data file, press open and the dataset number specified by the **dataset** control will appear in the graph indicator.



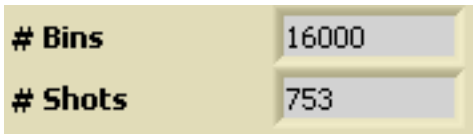
The signal type and wavelength are displayed in the graph legend

and the units used for the y-axis, which can be either megahertz or millivolts, are displayed in the upper left hand corner of the graph

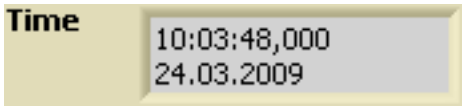
For photon counting data the discriminator level used during the acquisition is displayed

Furthermore the bin resolution is given in meters

and the number bins and the number of acquired shots are displayed:

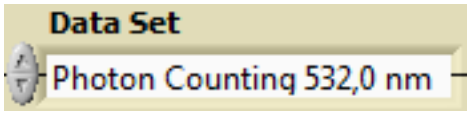


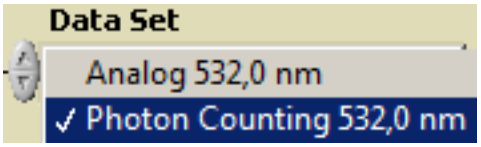
Additionally the acquisition's start time is shown.



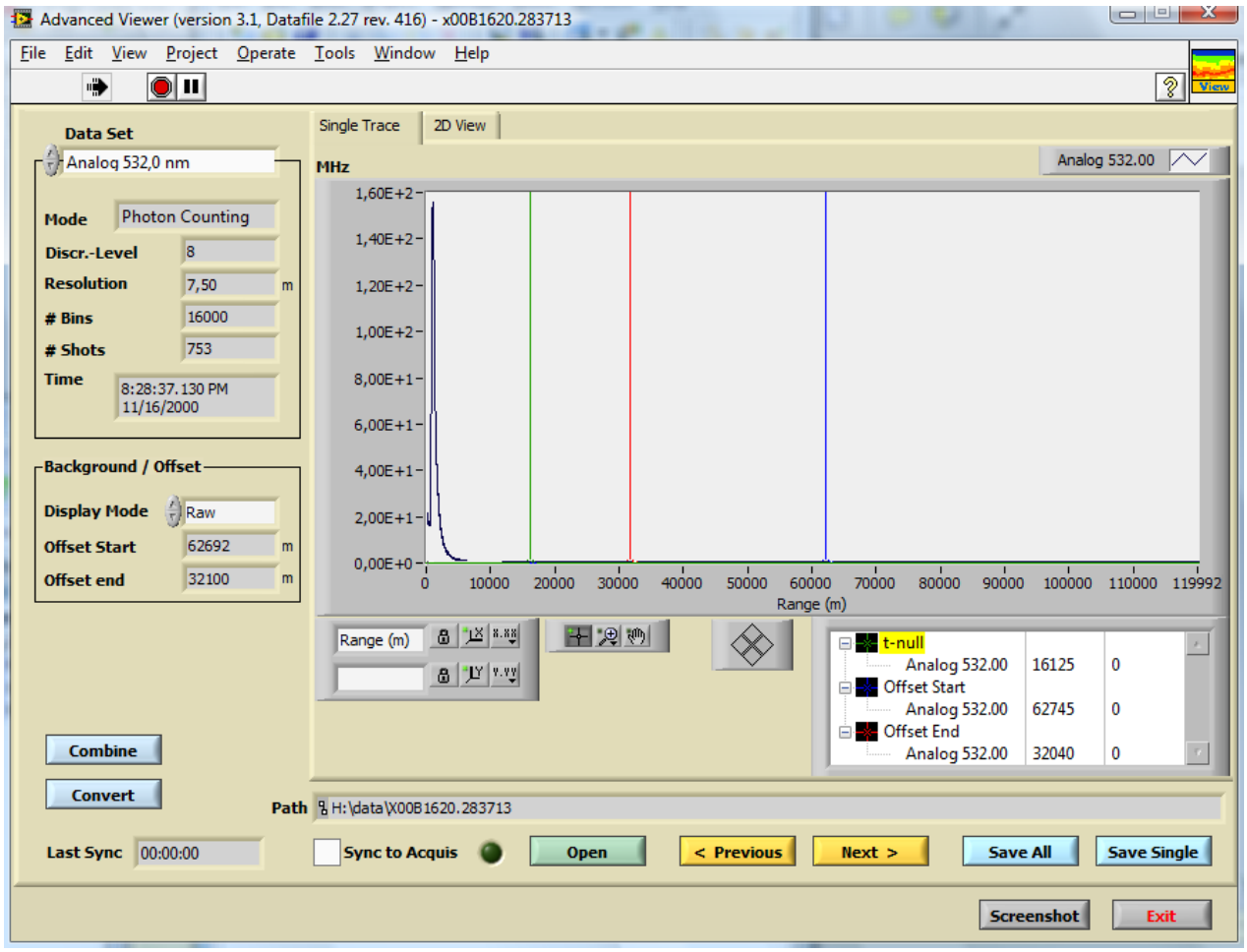
The full path to the current data file is shown in the **path** indicator.

If you would like to see a different dataset from the file, use the **Data Set** control to choose it.

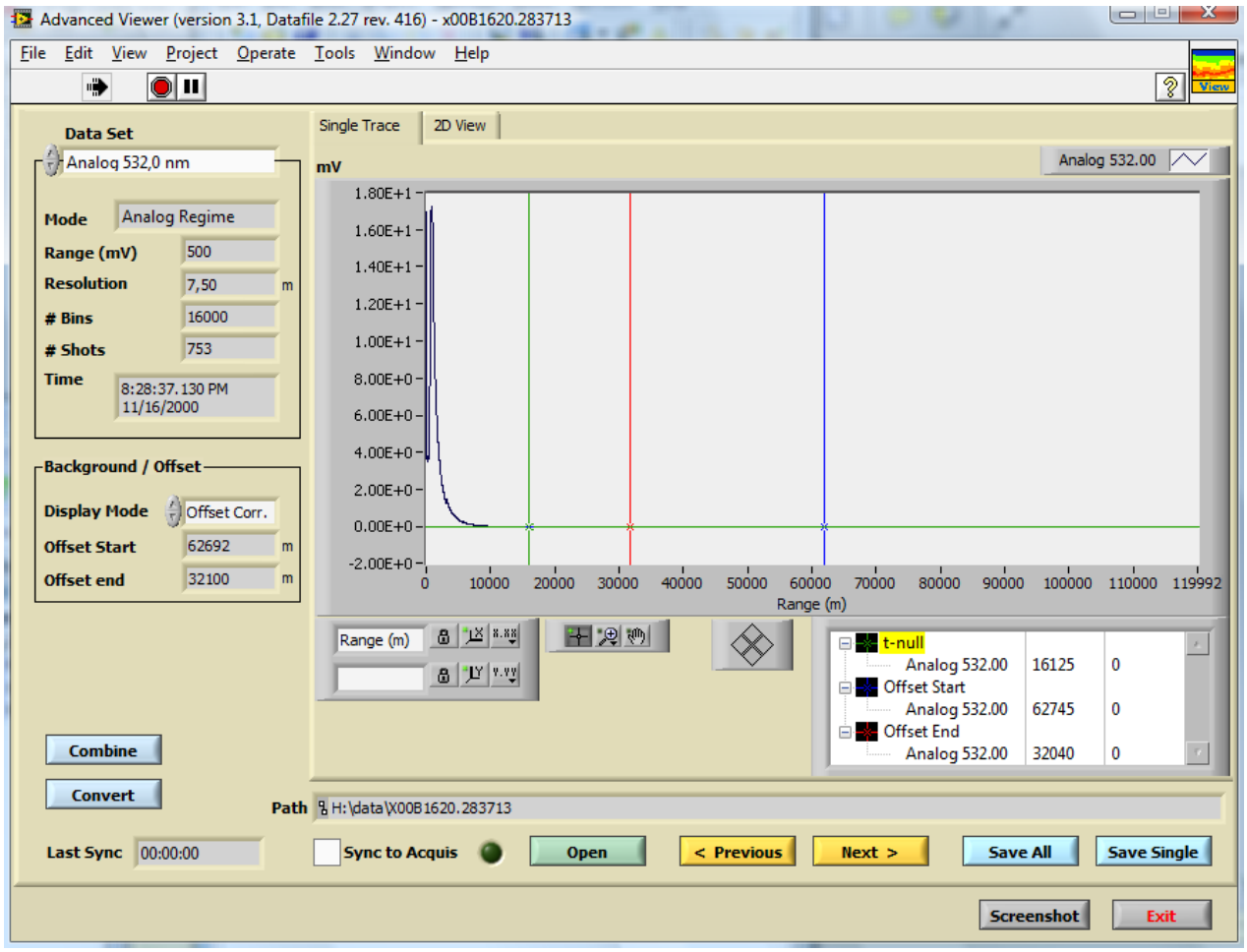




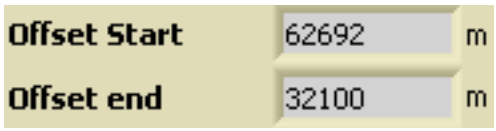
In this example, by switching to data set to *Analog 532 nm*, the analog 532 nm channel is displayed. Now, instead of the discriminator level the range used during the acquisition is displayed

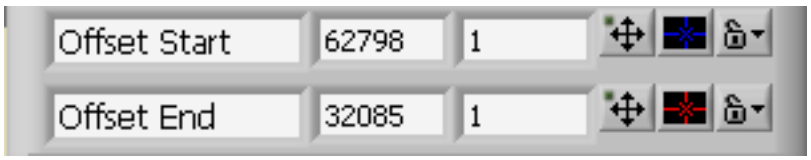


Note that the legend has changed to Analog 532 and the units have changed to millivolts. Currently the data is being shown in raw mode and we see that in the example above, that the baseline appears to have a value of about 1000. The **Offset Start** and **Offset End** cursors can be used to correct for the baseline offset. If the **display mode** *Offset Corr.* is used, then the mean value of the signal between these two cursors will be subtracted from the signal in order to create a baseline corrected signal. Use the cursor controls to move the blue and red cursors (Offset Start and Offset End) to a region which will be evaluated to generate the new baseline. Change the **Display Mode** to *Offset Corr.* and observe that the baseline of the signal changes.

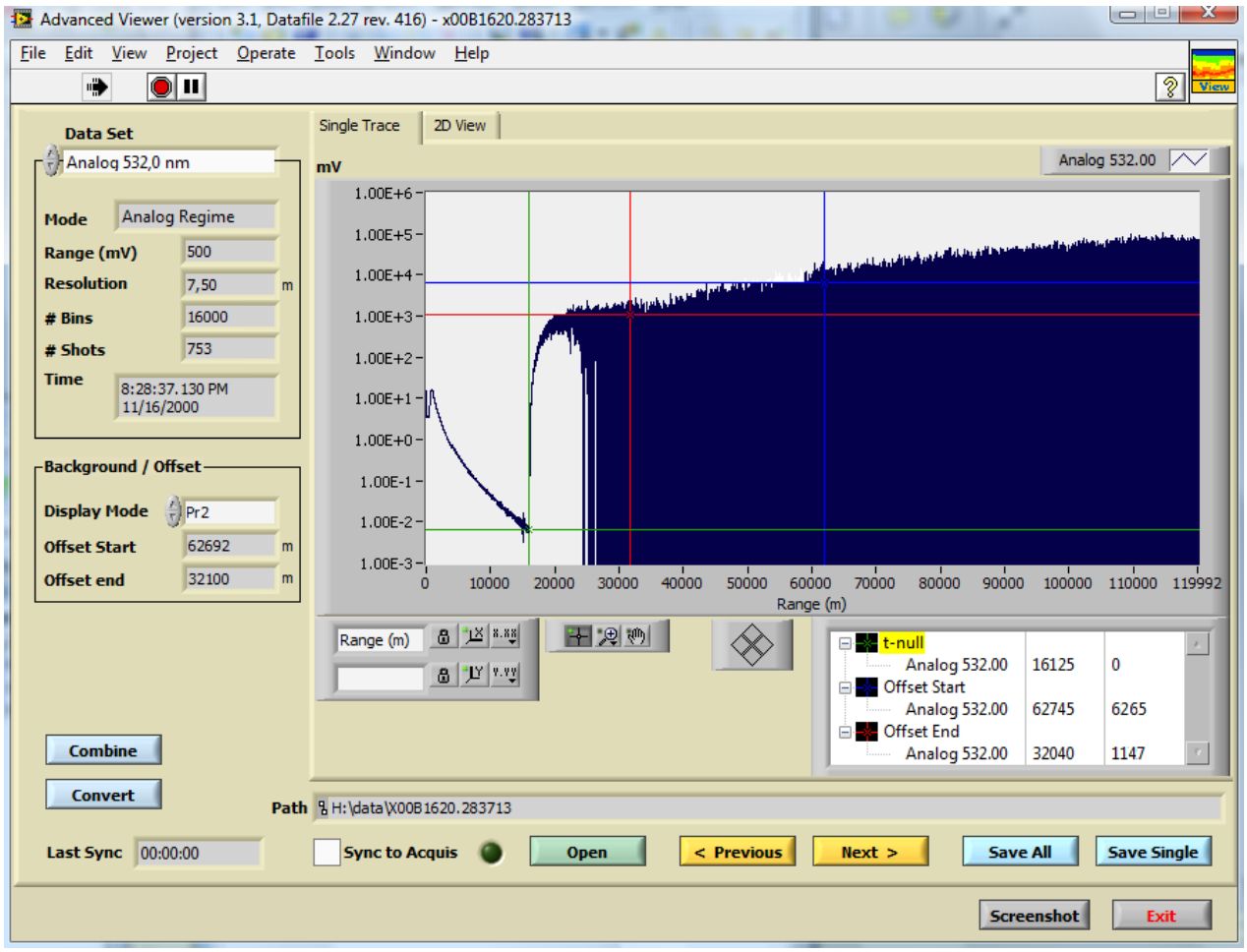


In the image above, you can see that the baseline is now close to zero. The end values of the region to be used to evaluate the baseline are shown in the **offset 1** and **offset 2** indicators as well as in the cursor controls.

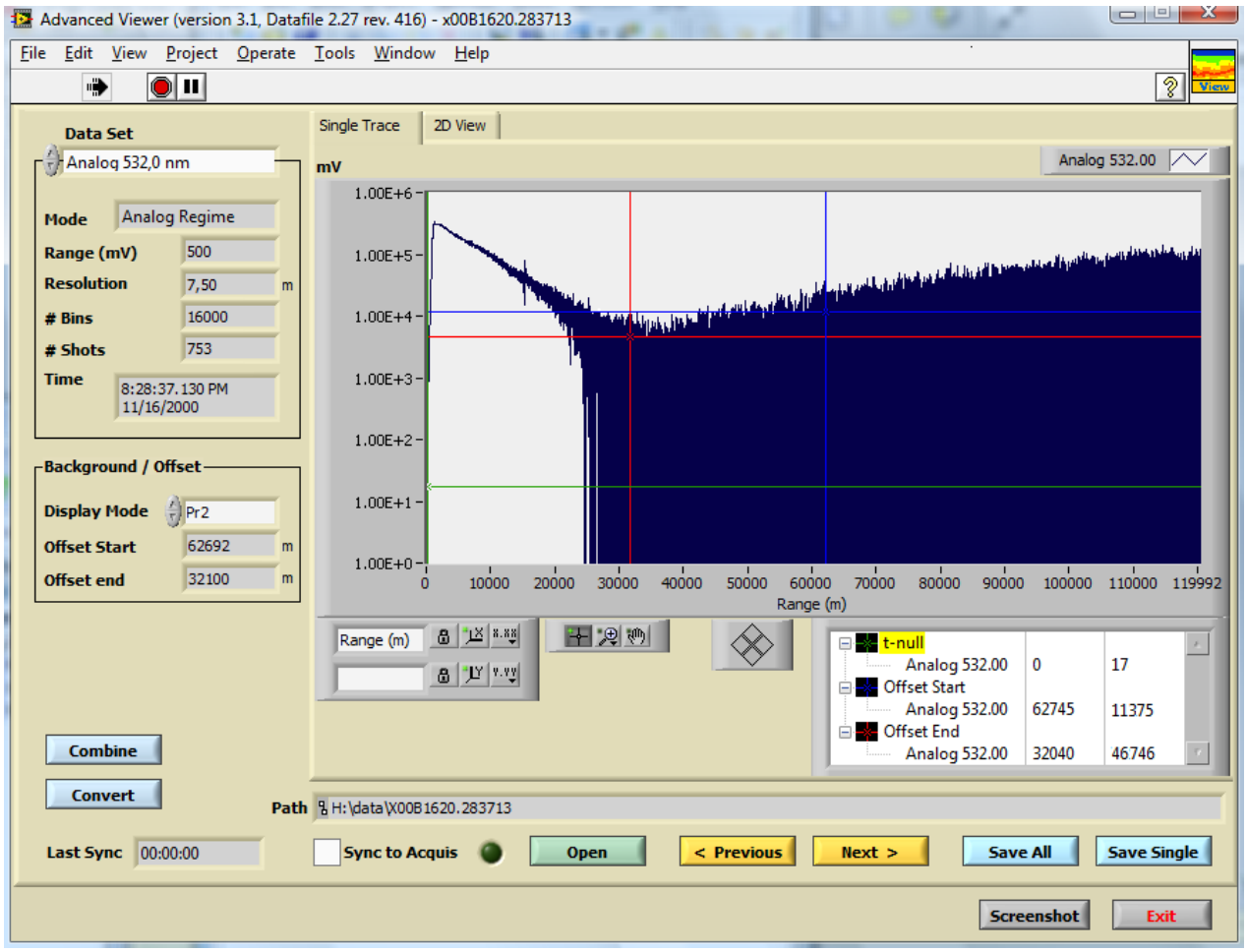




The data can also be displayed in the *Pr2* mode which corrects for the power loss due to the length of travel of the signal. The key parameter for the *Pr2* mode is **t-null** which defines the starting point of the signal. When switching to the *Pr2* mode, the data will look similar to the following.

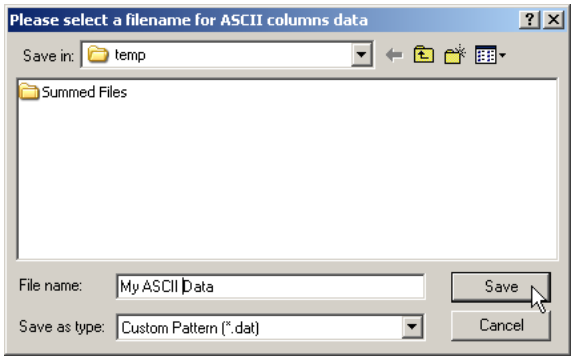


Note that the signal is only corrected for power loss after the green cursor, which is the t-null point. All values before **t-null** are left unchanged and those after **t-null** are corrected for the power loss due to distance. The difference in the display is due to the fact that the scaling has been changed to a logarithmic scale for easier viewing. After moving the cursor to the new t-null point, the signal changes to represent the new start time of the signal.



Please remember that the *Pr2* signal is offset corrected, as well.
Once you have adjusted your signal and would like to save it to a file in ASCII format as it is displayed, then press the **Save Single** button.

A file dialog appears asking you to name the ASCII file.



Enter the desired file name press **Save**. The file extension `.dat` will automatically be added to the file name unless you choose a different extension. Afterwards the data is saved in ASCII format as a column and can be imported into other programs for further evaluation.

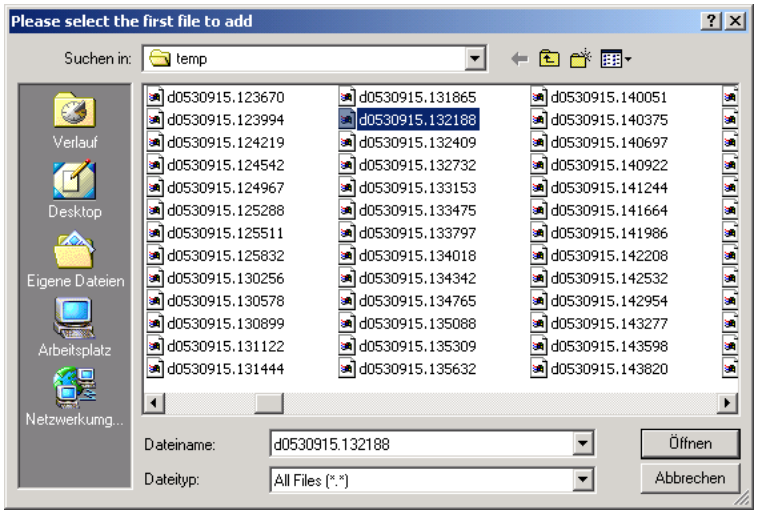
converts all datasets contained in the actual data file to an ASCII format file and appends the extension `.dat` to the end of the actual file's name. The whole file can then be imported into other programs.

If you would like to load the next file or previous file in a time series, this can be done by pressing the <**Previous** or **Next**> buttons.

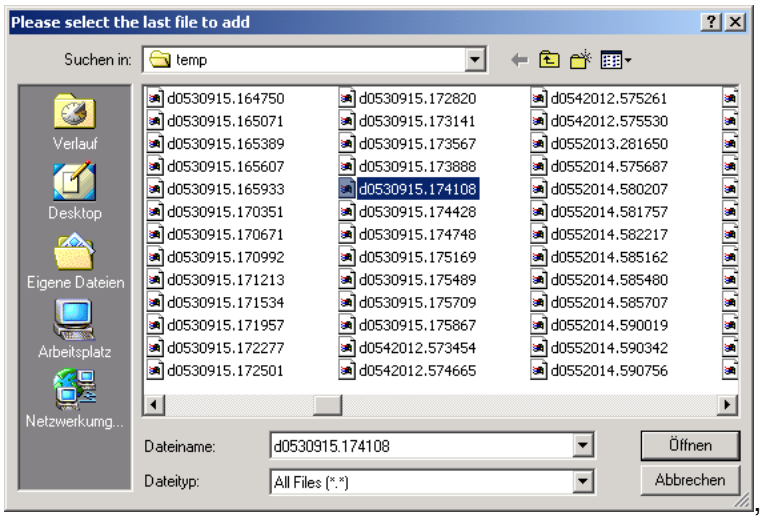
By pressing one of these buttons, either the file acquired before or after the current file will be displayed if it exists in the same directory. One can then manipulate the data using the aforementioned capabilities of `Advanced Viewer` and save the data from the new datasets to ASCII if desired.
Two data file utilities may be called from the `Advanced Viewer`, one to sum the data values of several files to one single file (`Datafile Addfiles Interface.vi`), and another to convert the data from several files to corresponding ASCII files (`Datafile Batch Converter.vi`).
By pressing **Combine** `Datafile Addfiles Interface.vi` is interactively called to sum the data contained in a set of subsequently recorded data files.

You will have to specify:

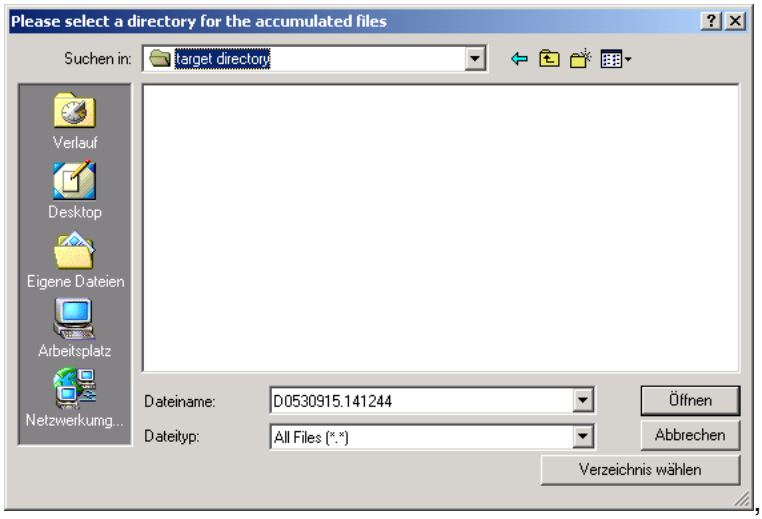
1. the name of the first file to add



2. the name of the last file to add

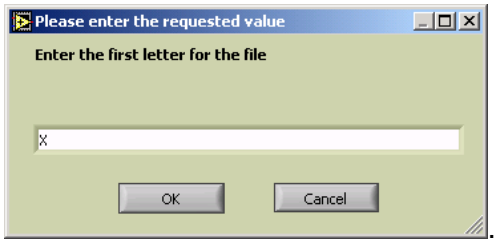


3. the name of the target directory for the file containing the summed data.



, and

4. the first letter of the file name



.

Both the first and the last file must reside in the same directory. The data from the files with acquisition dates/times lying between the first and the last files (including them) are summed and written to a target file into the target directory. The target file's name begins with the first letter, and the rest of the name is taken from the first selected file.
By pressing **Convert** `Datafile Batch Converter.vi` is interactively called to convert the data contained in a set of subsequently recorded data files to ASCII files.

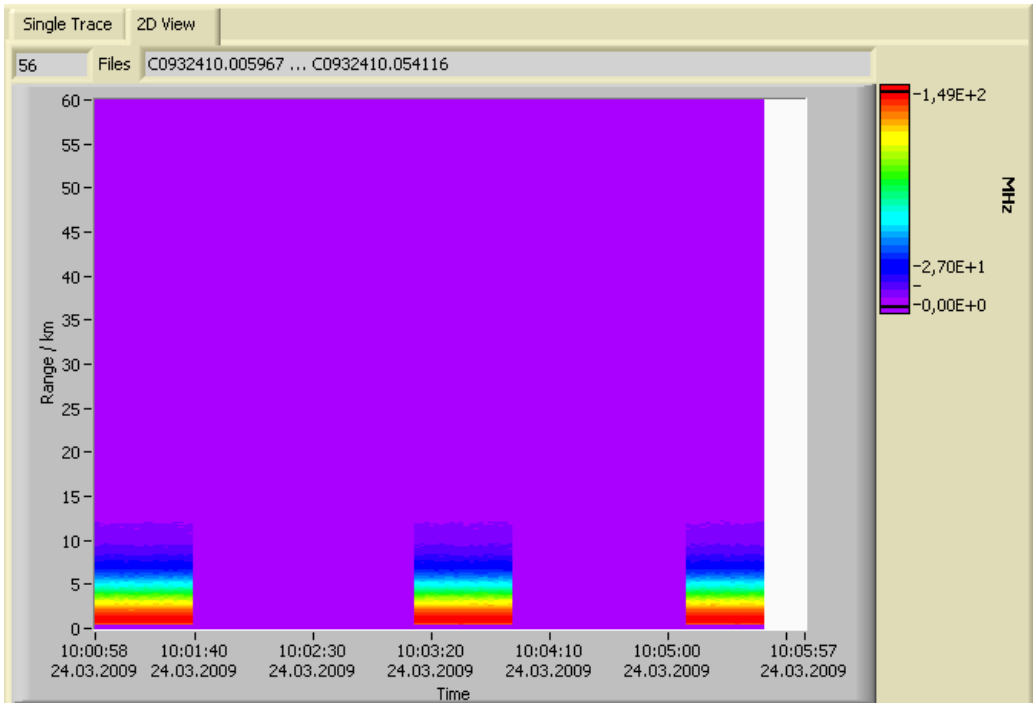
The selection mechanism to select the first file, the last file, and the target directory is the same as for the sum operation.
Both the first and the last file must reside in the same directory. The data from the files with acquisition dates/times lying between the first and the last files (including them) are converted as described above for the **Save All** operation. Each data file will result in an ASCII file named by the original file name with the additional extension `.txt`.
A screenshot of the current Viewer window can be saved to a portable network graphics (`png`) file by clicking the **Screenshot** button and selecting a file name in the subsequent file dialog.

The `Advanced Viewer` is terminated by pressing the **Exit Program** button.

2D Viewing of Multiple Data Files

The Advanced Viewer supports a 2D display for several data files. To switch to the 2D View please use the *2D View* page of the tab page selector.

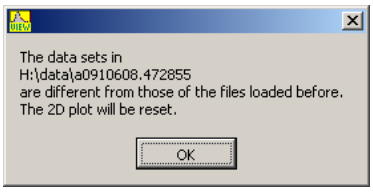
In the *2D View* the signal is displayed in a color map as a function of time and range.



Data is added to the 2D plot with the following buttons:

clears the 2D plot and adds the selected **Data Set** of the new loaded file according to the **Display Mode** setting.

adds the selected **Data Set** from the next file in the directory to the 2D plot and displays it according to the **Display Mode** setting. If the data is not compatible to the previously loaded files the 2D plot will be cleared before displaying the new file's data. A warning message will be displayed:



adds the selected **Data Set** from the previous file in the directory to the 2D plot and displays it according to the **Display Mode** setting. If the data is not compatible to the previously loaded files the 2D plot will be cleared before displaying the new file's data. A warning message will be displayed as above.

The number of loaded **Files** and the file name range is shown on the top of the *2D View* page. The data from the last file loaded by these buttons is always shown in the *Single Trace* plot according to the **Data Set** selection and **Display Mode** setting. If a file is already plotted in the 2D plot no data will be added. The `temp.dat` file created by the Acquis program will not be displayed in the 2D plot.

6.5 Further Data Analysis

The analysis of the acquired data depends strongly on the individual application of Licel detection systems. Therefore, we do not provide a complete data analysis package.

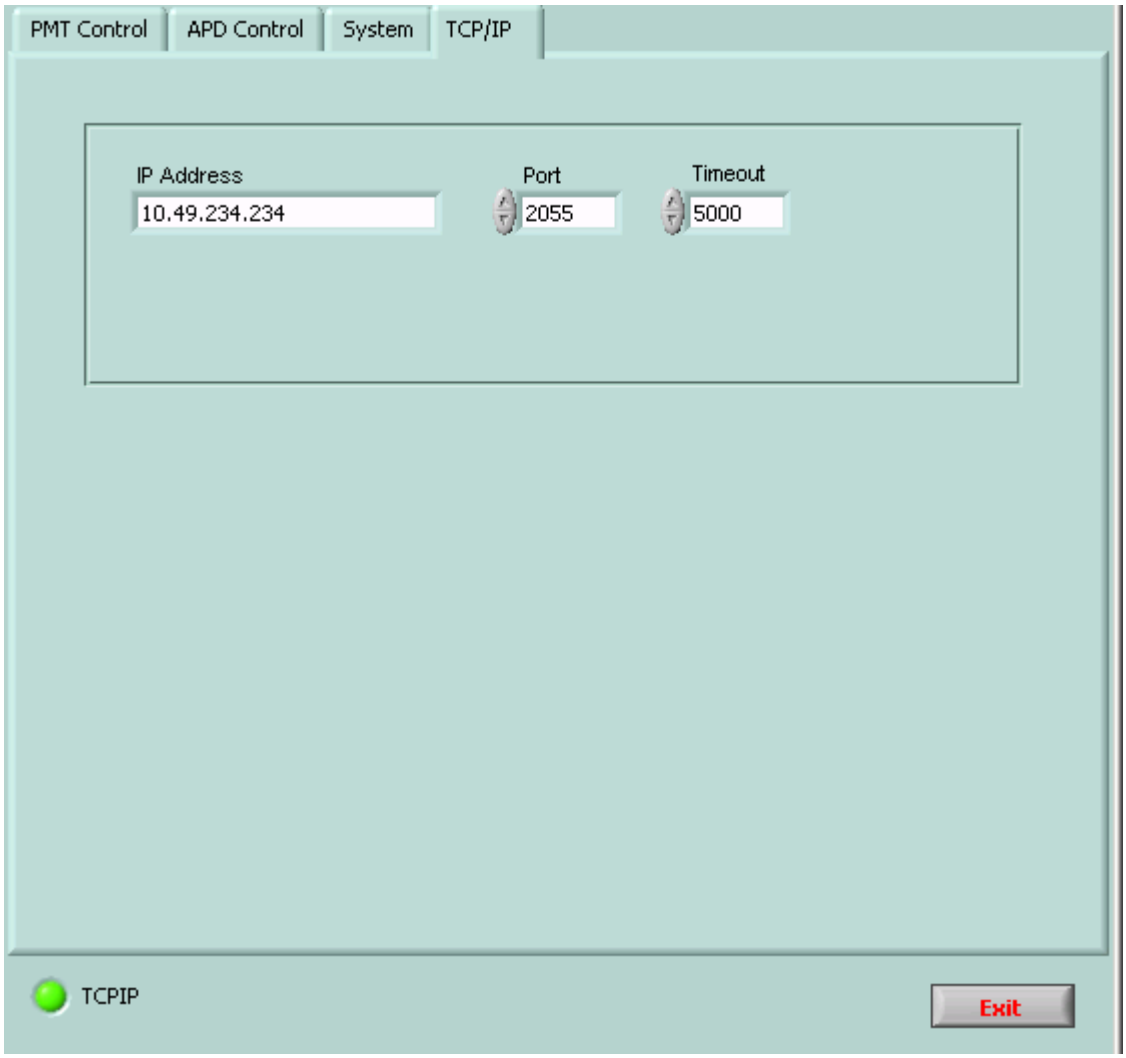
In an appendix we provide a strategy and an example to combine analog and photon counting data acquired with Licel transient recorders. Then a dynamic range of more than 5 orders of magnitude can be achieved.
The data file format is described in an appendix.

Chapter 7

Detector and Timing Control Utilities

7.1 The Combined APD and PMT Control Panel

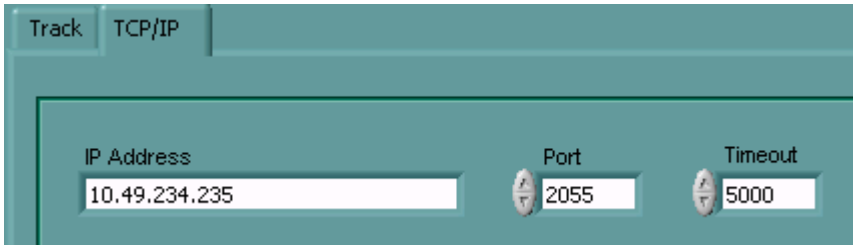
A sample application `ControlAPD-PMT.llb/Control_APD-PMT_Panel.vi` is provided to demonstrates the use of the driver VI's. The corresponding Windows application is started directly from the start menu in the subfolder `Licel \Detector Control`. After opening it you should see a screen like the following:



7.1.1 Starting the Application

1. First of all the **IP Address** and **Port** have to be set. You should already have set these values for the Licel Ethernet Controller following the network setup section above.

   - Using the LabVIEW vi, just enter the required values on the *TCP/IP* page and save them as defaults.

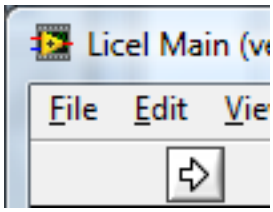
   - If you use the Windows application you must directly enter the correct values into the corresponding control fields on the *TCP/IP* page. On exit the values will be saved to the initialization file in the case that the TCP/IP connection is established.
   You may also set the values in the initialization file `Licel Main.ini`. You will see the full path of the file in a file path indicator on the *TCP/IP* page.

   - If you run `APD-PMT Control` within a sub panel on a page from `Licel Main` the latter is responsible for managing the TCP/IP connection.
   - Enter the number of available PMTs and APDs to the initialization file `Control APD-PMT.ini` in the same directory where the LabVIEW llb or the executable program resides:

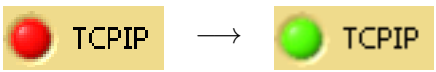
   ```
   [Number_Of_PMTs]
   Number_Of_PMTs=5
   [Number_Of_APDs]
   Number_Of_APDs=1
   ```

2. To start the program press the **Run** button at the top left of the screen.

   The Windows application will start automatically when called for the first time.

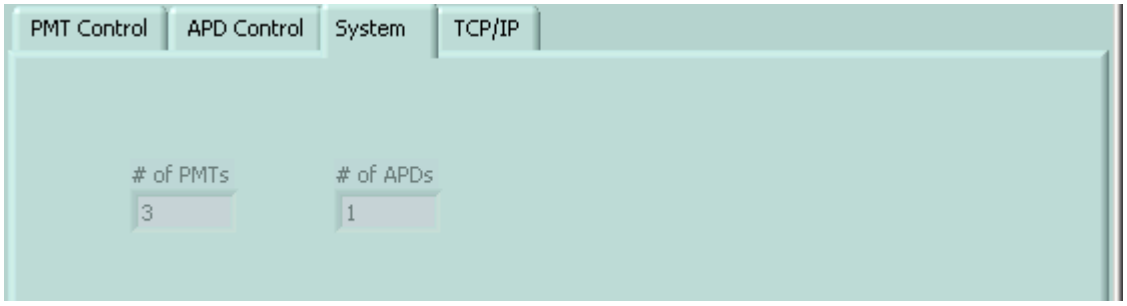
3. After a short time the **TCPIP** indicator should change its color from red to green indicating a successful connection with the Licel Ethernet Controller. If the indicator remains red and/or an error is indicated, please check the values for **address** and **Port**, change them (on the program's panel or in the initialization file) if necessary. Check if the Licel Ethernet Controller is running and that all network connections are correct.

7.1.2 Operation

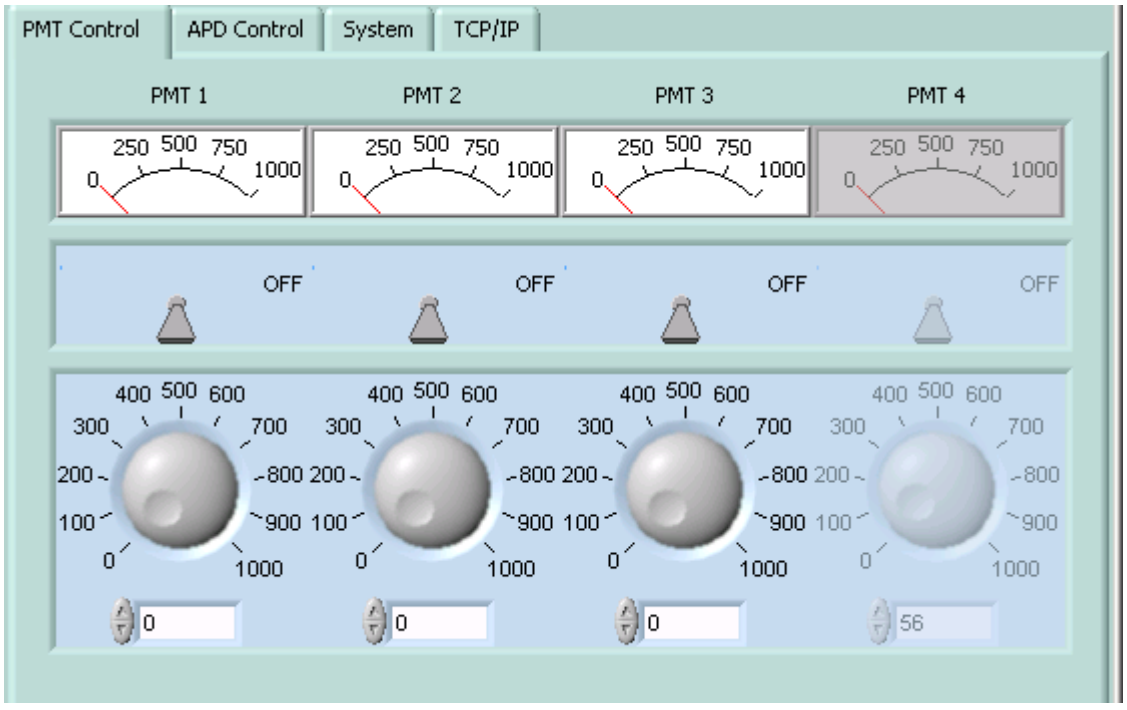
Dependent on the Licel Ethernet controller capabilities only those tab pages will be shown with the corresponding functionality. If a controller is capable to control both, PMTs and APDs, the pages *PMT Control* and *APD Control* will be present.

System Information



On the page *System* you may inspect the number of PMTs and APDs.

PMT Control



On the page *PMT Control* the voltage of the PMTs can be set via the knobs at the bottom. The displays at the top show the actual voltages. Turning the switches on, will set the desired voltages at the PMT.

APD Control

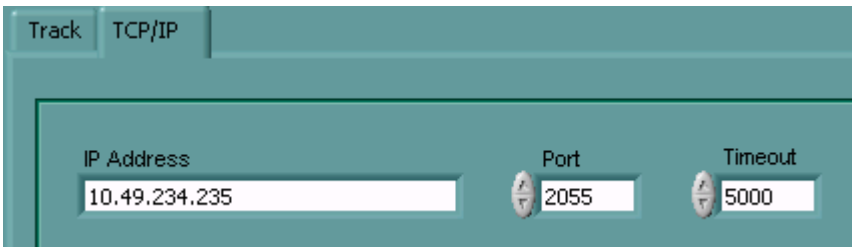
On the page *APD Control* The voltages can be set via the knobs at the bottom. The displays at the top show the actual voltages. Turning the switches on, will set the desired voltages at the APD. After switching from the passive to the active cooling mode the cooling status indicator will first turn red indicating that the APD temperature is not in range later will become green when the temperature is stabilized.

7.2 The Trigger Module Control Panel

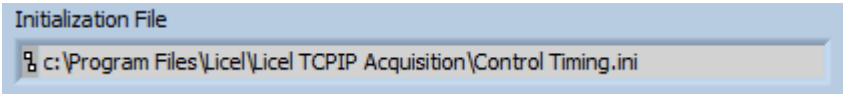
A sample application `ControlTiming.llb/Control Timing.vi` is provided to control the timing parameters of the trigger module.

7.2.1 Starting the Application

1. First of all the **IP Address** and **Port** have to be set. You should already have set these values for the Licel Ethernet Controller following the network setup section above.
   - Using the LabVIEW vi, just enter the required values on the *TCP/IP* page and save them as defaults.

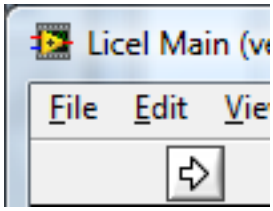


- If you use the Windows application you must directly enter the correct values into the corresponding control fields on the *TCP/IP* page. On exit the values will be saved to the initialization file in the case that the TCP/IP connection is established.
  You may also set the values in the initialization file `Licel Main.ini`. You will see the full path of the file in a file path indicator on the *TCP/IP* page.



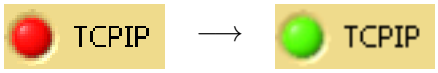
- If you run `Control Timing` within a sub panel on a page from `Licel Main` the latter is responsible for managing the TCP/IP connection.

2. To start the program press the **Run** button at the top left of the screen.



   The Windows application will start automatically when called for the first time.

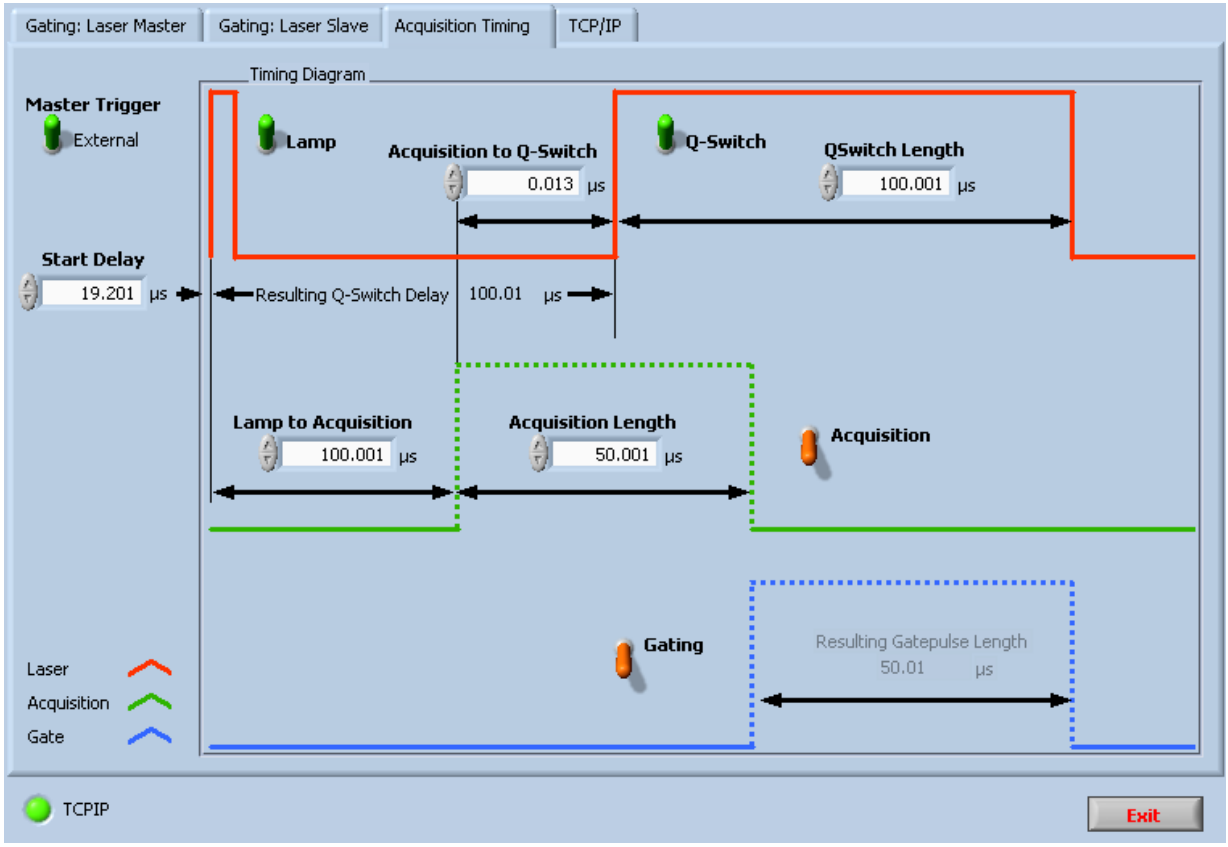
3. After a short time the **TCPIP** indicator should change its color from red to green indicating a successful connection with the Licel Ethernet Controller. If the indicator remains red and/or an error is indicated, please check the values for **address** and **Port**, change them (on the program's panel or in the initialization file) if necessary. Check if the Licel Ethernet Controller is running and that all network connections are correct.



Note that in each of the following scenarios the pretrigger pulses are only generated if the corresponding switches are in the *On* position.

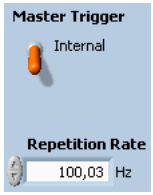
7.2.2 Direct Control of the Timing Parameters

In this scenario the user has to directly set the same timing parameters as the Licel Trigger Module is using.



The following parameters may be changed or set:

- **Master Trigger**: Determines whether the triggers are internally (*Internal*) generated or an external trigger is supplied (*External*). In the *External* mode the **Start Delay** may be set, in the *Internal* mode the **Repetition Rate** can be set.
- **Repetition Rate**: Frequency in Hz of the internally generated trigger pulses, i.e. of the laser **Lamp**, **Acquisition**, **Q-Switch**, and **Gating** pulses. The **Repetition Rate** is available in the *Internal* mode.

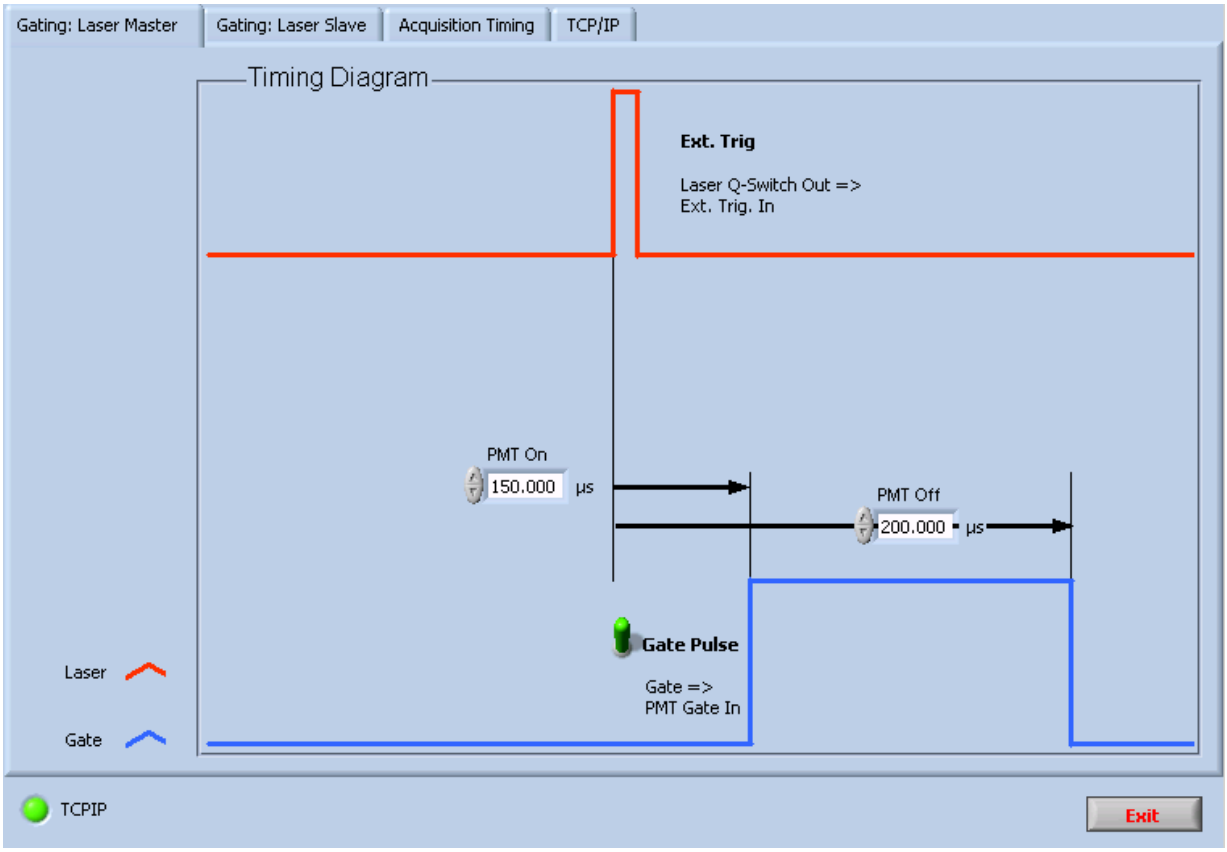


- **Start Delay**: Initial delay time at the start of the time chain. The **Start Delay** is available in the *External* mode. For trigger controllers shipped before june 2007 a firmware update is required to use the **Start Delay**. Otherwise the delay time set is ignored (start delay 0 µs).
- **Lamp to Acquisition**: Delay time between the laser lamp pulse and the acquisition pretrigger pulse (µs)
- **Acquisition Length**: Length of the acquisition pretrigger pulse (µs)
- **Acquisition to Q-Switch**: Delay time between the acquisition pretrigger pulse and the Q-switch trigger pulse (µs)
- **Q-Switch length**: Length of the Q-switch trigger pulse (µs).

Lamp, **Acquisition**, **Q-Switch**, and **Gating** have to be set *On* to generate the corresponding trigger pulses.

7.2.3 Gating: Laser Master

Use this scenario if the laser's Q-switch output is used as the trigger source. Then the gate pulse is characterized by its start and end times with respect to the external trigger.



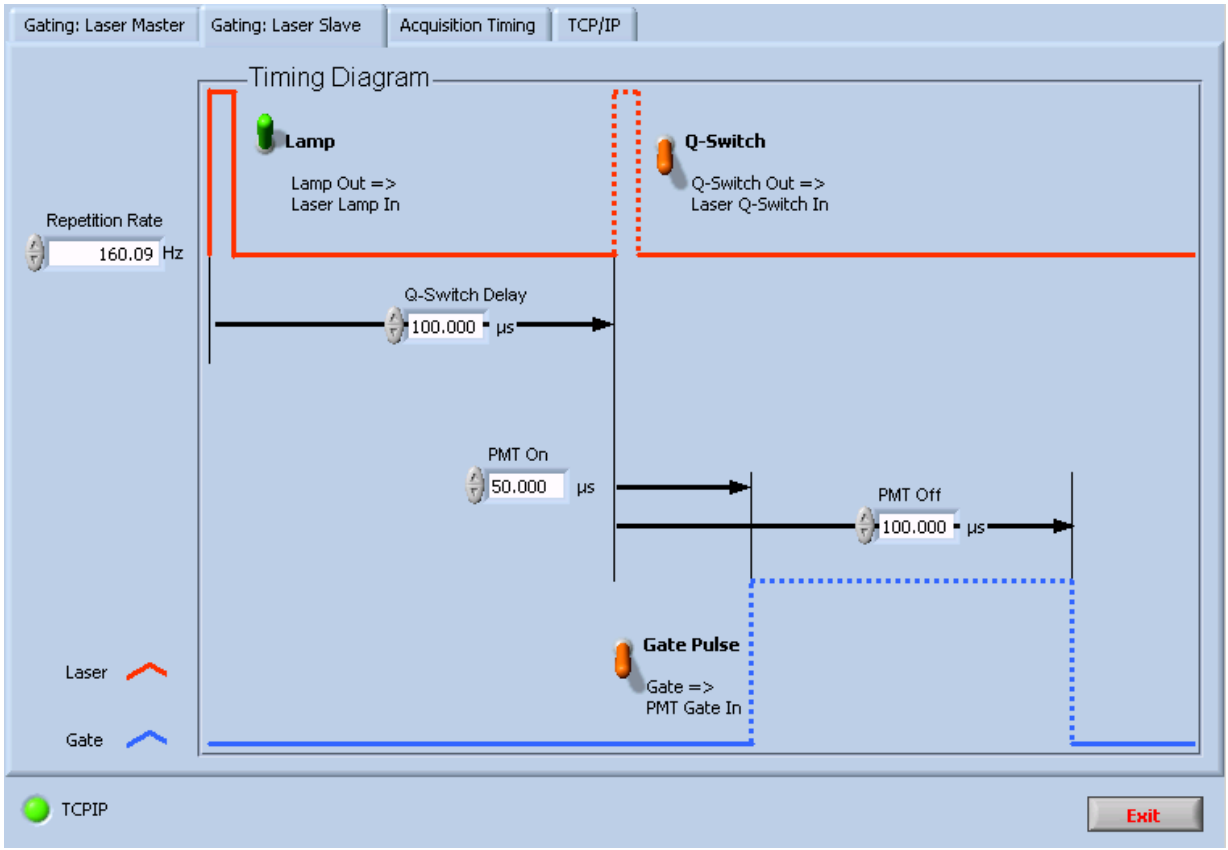
The following parameters may be set (µs):

- **PMT On**: Gate pulse start time with respect to the external trigger (laser Q-switch out)
- **PMT Off**: Gate pulse stop time with respect to the external trigger (laser Q-switch out).

Gate Pulse needs to be set *On* to enable the gate pulse output.

7.2.4 Gating: Laser Slave

Use this scenario to trigger the laser lamp and the Q-switch from the Licel Trigger Module. Then the gate pulse is characterized by its start and end times with respect to the Q-switch pulse.



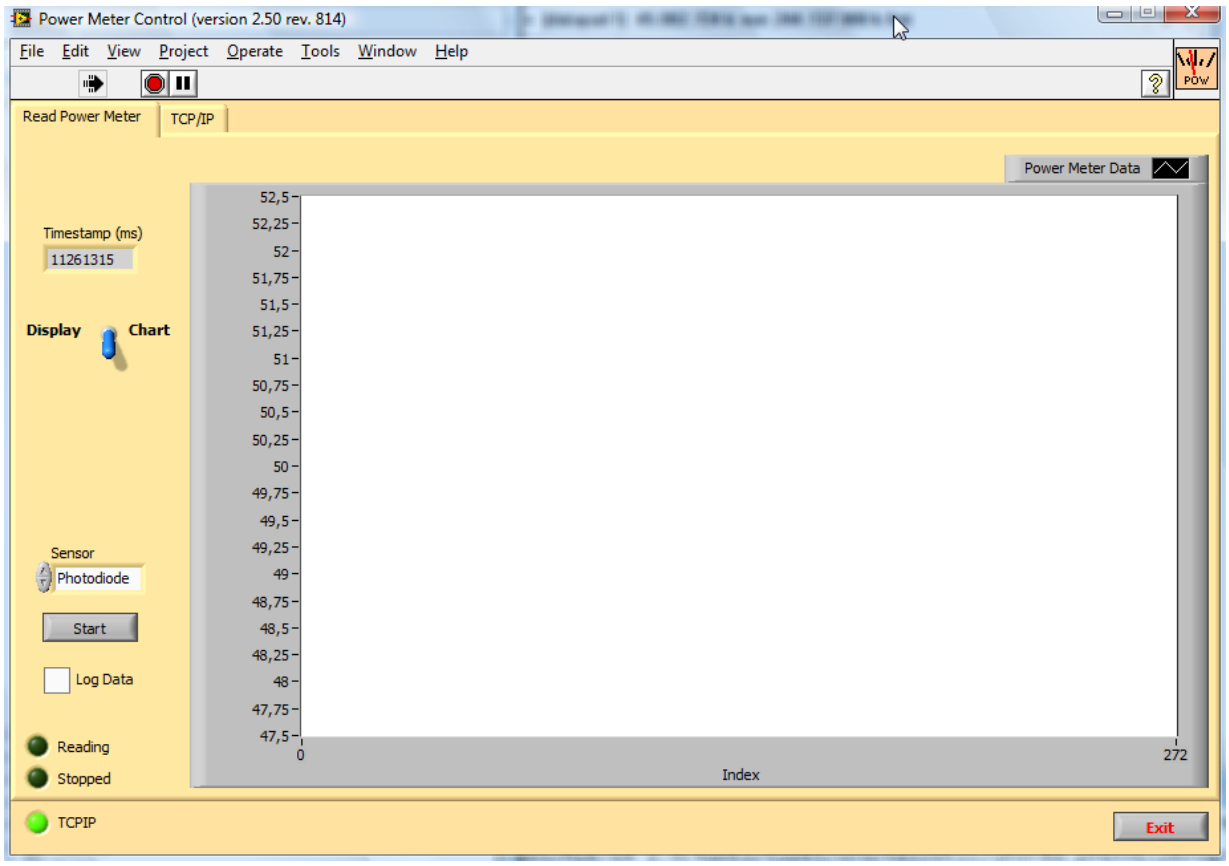
The following parameters may be changed or set:

- **Repetition Rate**: Frequency in Hz of the internally generated trigger pulses, i.e. of the laser **Lamp**, **Q-Switch**, and **Gating** pulses.
- **Q-Switch Delay**: Time between the lamp trigger output and the Q-Switch output (µs)
- **Gate On**: Gate pulse start time with respect to the Q-switch out (µs)
- **Gate Off**: Gate pulse stop time with respect to the Q-switch out (µs).

Lamp, **Q-Switch**, and **Gate Pulse** have to be set *On* to generate the corresponding trigger pulses.

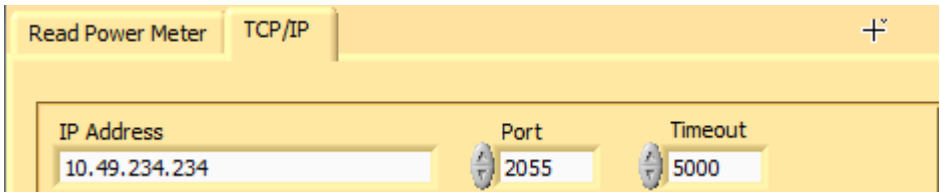
7.3 Power Meter Control

To load the power meter control program, either double click on the `TCPIP-Power Meter.llb` or open the file `Power Meter Control.vi` in the `Power Meter.llb`. If you installed the Windows applications please start the program by selecting the corresponding entry in the Licel section of the Windows Start menu. The Power Meter Control software is capable to control one Power Meter controller.
The front panel of the Power Meter Control software is seen in the next picture:

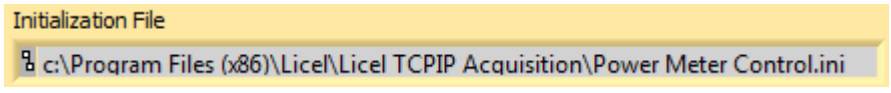


1. First of all the **IP Address** and **Port** have to be set. You should already have set these values for the Licel Ethernet Controller following the network setup section above.

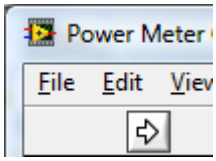
   - Using the LabVIEW vi, just enter the required values on the *TCP/IP* page and save them as defaults.



   - If you use the Windows application you must directly enter the correct values into the corresponding control fields on the *TCP/IP* page. On exit the values will be saved to the initialization file in the case that the TCP/IP connection is established.
   You may also set the values in the initialization file `Power Meter Control.ini`. You will see the full path of the file in a file path indicator on the *TCP/IP* page.

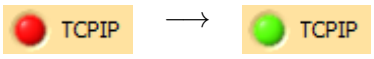


2. To start the program press the **Run** button at the top left of the screen.

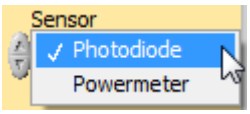


   The Windows application will start automatically when called for the first time.

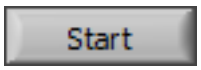
3. After a short time the **TCPIP** indicator should change its color from red to green indicating a successful connection with the Licel Ethernet Controller. If the indicator remains red and/or an error is indicated, please check the values for **address** and **Port**, change them (on the program's panel or in the initialization file) if necessary. Check if the Licel Ethernet Controller is running and that all network connections are correct. The LED of the transient recorder should be lit up.



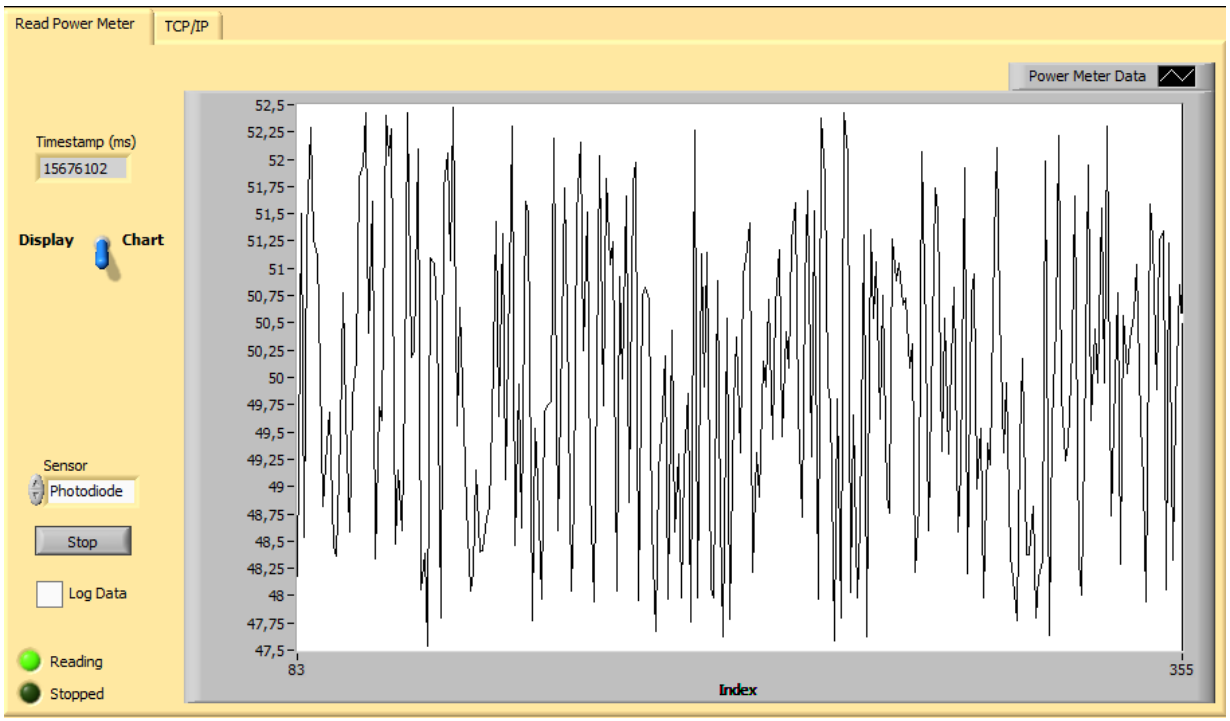
Once the TCP/IP connection is established the **Sensor** needs to be selected so that it corresponds to the external detector (*Photodiode* or *Power Meter*).



Press the start button to start acquiring data from the power meter controller.

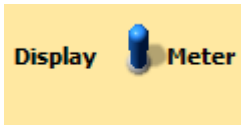


The button text will change to *Stop*. The **Reading** LED will turn to light green. The acquired data is displayed in the graphic indicator.



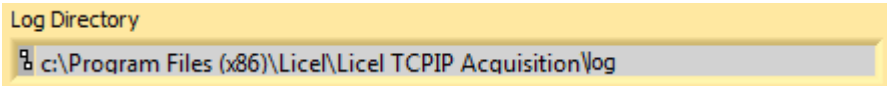
The display of the data can be in two modes, either history chart that shows the last 1024 readings or a power scale. The button text will change to *Stop*. Use the **Display** selection switch to change the display mode.

or



Check **Log Data** to write the acquired data to a file. Two tab-seperated ASCII columns are written, the first column contains the millisecond timer value returned by the controller, and the second contains

the sensor value. The files are written to the sub directory `log`. The file name is YYYYmmdd_HH-MM-SS_uu.log where YYYYmmdd is the date, and HH-MM-SS the 24 hour time. uu stands for the first decimal places of the seconds. The complete path of the log file directory can be inspected on the *TCPIP* tab page:



An acquisition is stopped using the *Stop* button. After stopping the **Stopped** LED may turn to light green until all data already sent by the controller has been received.

7.3.1 Initialization File

The following settings are used in the initialization file `Power Meter Control.ini`.

```
[TCPIP]
UseValues = TRUE
Port = 2055
IPAddress = "10.49.234.234"

[Power Meter Control]
HideRootWindow = True

[Data]
CalibrationFactor = 1
UnitLabel = ""
Offset = 0
```

`UseValues`	Use TCPIP values from ini file (Only for Windows applications)
`Port`	TCPIP Port
`IPAddress`	TCPIP address
`CalibrationFactor`	Conversion from binary controller data to physical unit
`UnitLabel`	physical unit
`Offset`	Offset in physical units

7.3.2 LabVIEW TCPIP Power Meter VIs

The low level TCP/IP commands supported by the Licel Power Meter Controller is described in the corresponding appendix.

7.3.3 Data Acquisition with TCPIP Acquis

The acquisition program `TCPIP Acquis` is capable to communicate with 1 or more instances of the Power Meter Control software to configure, start, and stop power meter data acquisitions and to read the acquired data to store it in the `TCPIP Acquis` data files together with the transient recorder data. For the configuration please refer to the TCPIP Acquis configuration.
The needed instances of the Power Meter Control software are then combined in the container application `Multi Power Meter Control` which is described in the next subsection.

7.3.4 Multi Power Meter Control

`Multi Power Meter Control` is a variant of `Lisel Main`. `Multi Power Meter Control` is intended to load several instances of `Power Meter Control.vi`. This mechanism is defined in an initialization file.

Initialization File

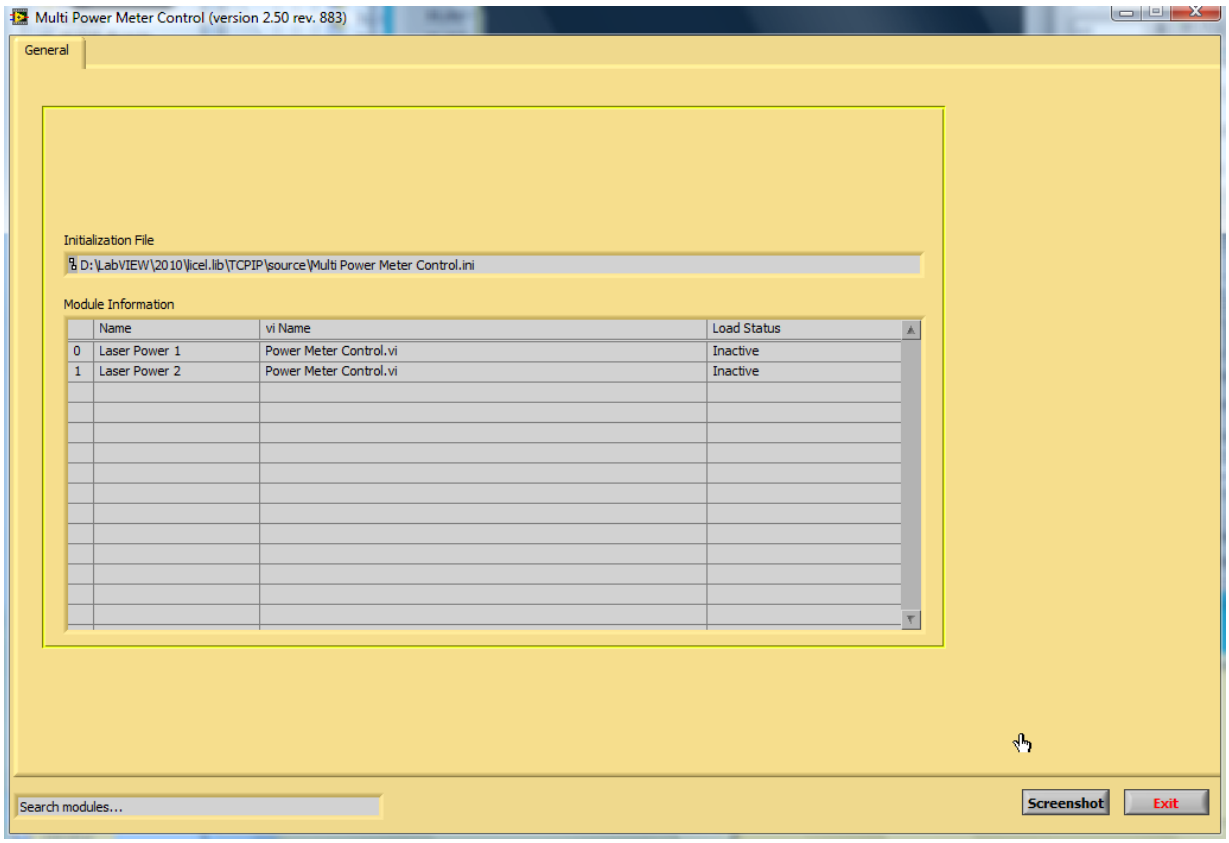
The syntax of the initialization file `Multi Power Meter Control.ini` is equal to that used by `Licel Main`. Each section `[Module<n>]` defines how to load an individual instance of `Power Meter Control.vi`.

```
; you may duplicate the sections named [Module<x>]
; containing the key 'Path = "Power Meter Control.vi"'
; to run more than 1 Power meter
; the key value of 'Name = ..' should always end up with
; a number TCPIP Acquis will automatically handle this file
[Module0]
Active = "TRUE"
Path = "Power Meter Control.vi"
Name = "Laser Power 1"
Controllers = "0"
CAPs = "POW"
Push = "FALSE"
IndependentTCPIP = "TRUE"
```

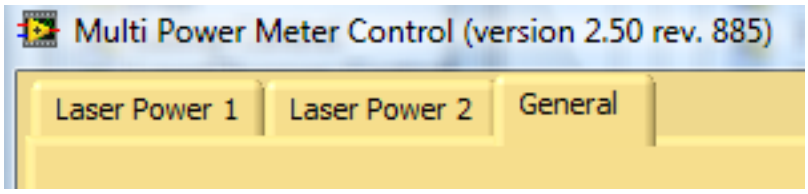
Each section must contain the initialization file key `IndependentTCPIP = "TRUE"` as each `Multi Power Meter Control` will communicate with a different Power Meter Controller.
In the case that `TCPIP Acquis` starts `Multi Power Meter Control` the initialization file is prepared by the acquisition program — in the configuration dialog the number of power meters can be specified.

Starting the Multi Power Meter Control

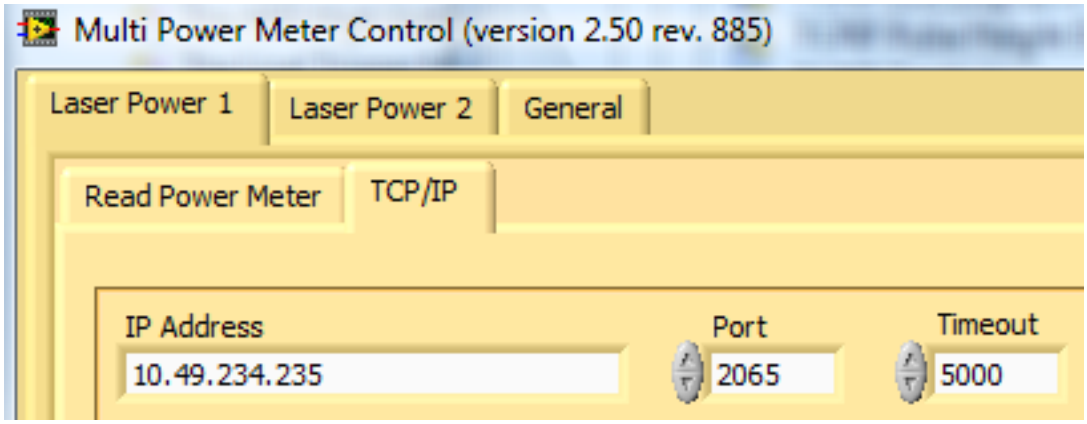
The Program can be started in the same way as `Lisel Main`. **In the case that the power data acquisition is integrated into `TCPIP Acquis`** the latter will starting it it after preparing the initialization file.
The front panel contains a tab page with a table containing the information about the modules to load.



After starting the program will load and initialize the requested instances of the `Power Meter Control` software. For each instance a tab page will be added according to the settings in the initialization file.



When starting `Multi Power Meter Control` for the 1st time the user must set the IP addresses and ports for each of the loaded Power meter Control Instances. These values will be saved in the initialization files `Power Meter Control<`*n*`>.ini` for the *n*th instance and reloaded at the next start of the program. This mechanism known from the behavior of most Licel TCPIP Windows applications is used here in the case that the LabVIEW development environment is in use, as well.



Each instance of `Power Meter Control` will continue to run even if the tab page of the container application `Multi Power Meter Control` changes.
When controlled by `TCPIP Acquis` the Sensor (*Photodiode* or *Powermeter*) will automatically be set according to the setting in the TCPIP Acquis configuration. Then the start and stop of the power meters will automatically be controlled, too.

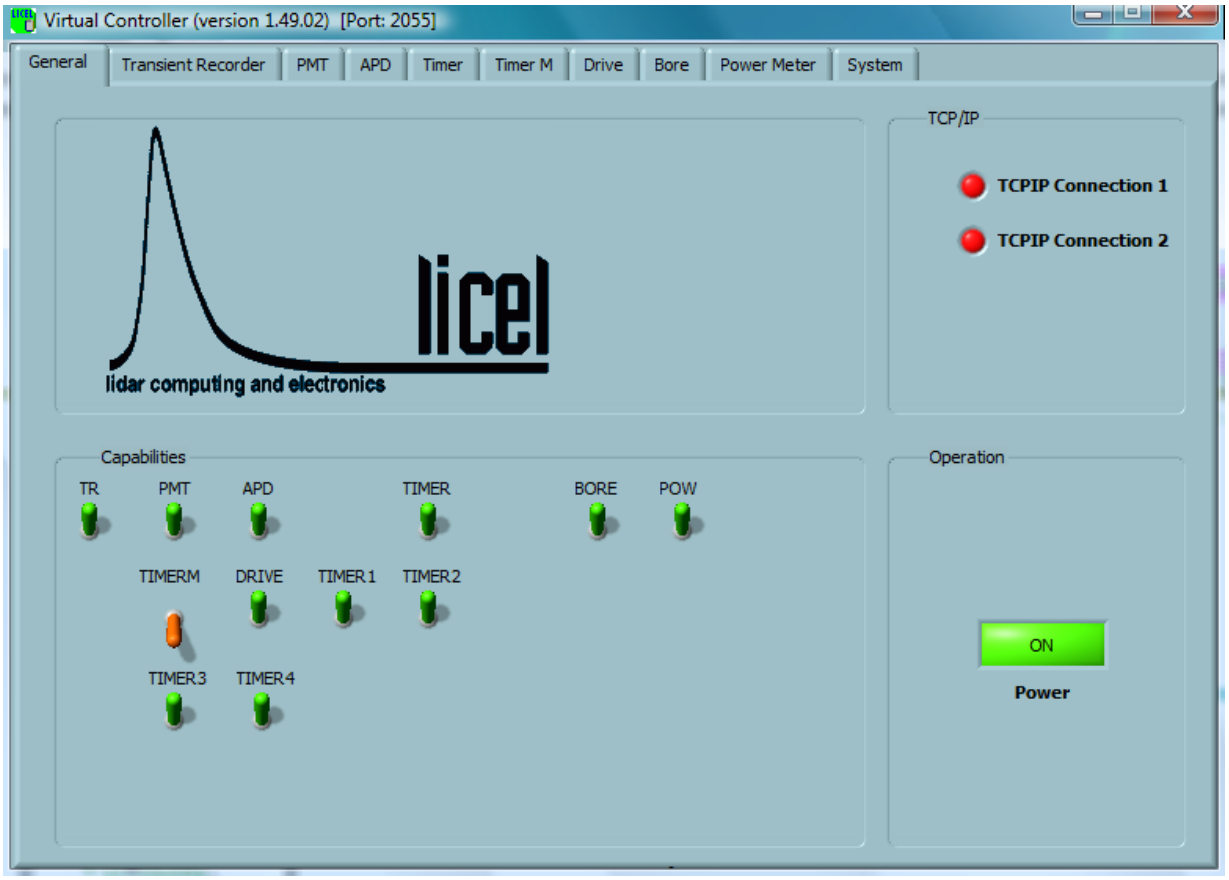
Chapter 8

The Licel Virtual Controller

The Licel Virtual Controller simulates the behavior of a Licel Ethernet Controller. This will help a user to develop and test his own software as debugging is possible without having the hardware installed. The software can substitute hardware while testing the other LabVIEW modules provided by Licel or while developing custom software. The Licel Virtual Controller supports the TCPIP commands related with the standard functionality of a Licel Ethernet Controller. Furthermore it is able to simulate realistic backscatter signals to test LIDAR acquisition software.

8.1 Starting the Application

The Licel Virtual Controller is started directly from the Windows start menu and opens with the following screen.



The Licel Virtual Controller is waiting for applications to connect to it via TCP/IP. These applications must use the IP address of the computer where the Licel Virtual Controller is running. If the appli-

cation you would like to run against the Virtual Controller is located on the same machine, just use `127.0.0.1` (localhost) as the IP address. The Licel Virtual Controller is listening on port 2055 which equals the standard port of the Licel hardware. The port may be changed on the *System* tab page. Note that the choosen port must not be used by other programs and not be blocked by any firewall.
The Virtual Controller provides tab pages for each capability and furthermore the pages *General* and *System*.

8.2 Initialization File

On start the Virtual Controller searches for an initialization file named like the executable application (default: `Virtual Controller.ini`) with the following entries:

```
[TCPIP]
Port=2055
[UDP]
Port=2000
[CAPS]
File="standard.cps"
```

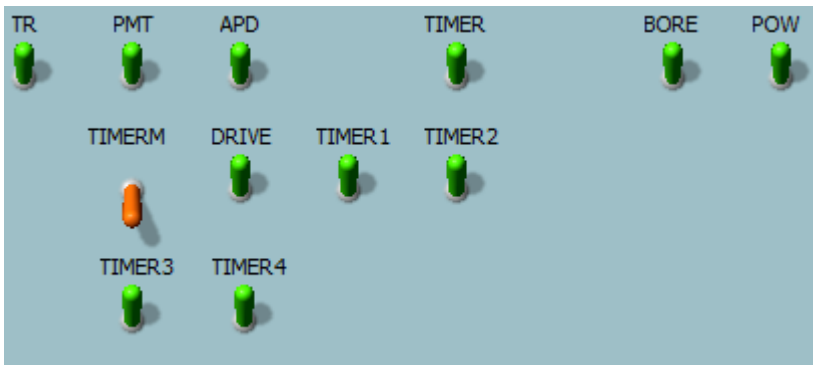
`Port` in the `[TCPIP]` section determines the port for the TCP/IP command socket. The Virtual Controller uses the ports `Port`...`Port+2`. `Port` in the `[UDP]` section determines the UDP port. The entry `File` in the `[CAPS]` section may only be used if Licel provided you with such a file. `cps` encode information about the virtual capabilities of the Virtual Controller. If no file is specified the Virtual Controller has the default capabilities.
If no initialization file is present the Virtual Controller uses default values.
Initialization files must be used to run more than one Virtual Controller. Follow the steps below to configure a system with 2 Virtual Controllers:

1. Create a file `Virtual Controller.ini` and enter the values for the `[TCPIP]` and `[UDP]` sections
2. Copy the application `Virtual Controller.exe` to `Virtual Controller1.exe`
3. Copy the initialization file `Virtual Controller.ini` to `Virtual Controller1.ini`
4. Change the `port` values in `Virtual Controller1.ini` to 2065 for `[TCPIP]`, and 2001 for `[UDP]`
5. Start both Virtual Controllers, one is now accessable by your software at port 2055, the second at 2065.

8.3 General

On the tab page *General* you may inspect and set the following parameters:

Capabilities Each switch here represents a capability of the Virtual Controller. The standard capabilities are **TR**, **PMT**, **APD**, **TIMER**, **BORE**, **POW**, **TIMERM**, **DRIVE**, **TIMER1**, **TIMER2**, **TIMER3**, and **TIMER4**. Licel may provide more capabilities for customized systems. The active capabilities will be part of the answer to the CAP? command. If you switch a capability off, corresponding commands will no longer be accepted. This is useful for debugging purposes to implement an appropriate error handling at the calling application.

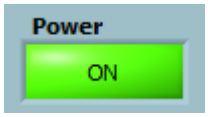


TCP/IP Indicators **TCPIP Connection 1** and **TCPIP Connection 2** indicate wether applications are connected via TCP/IP with the Virtual Controller.

If both indicators are red, no client is connected. **TCPIP Connection 1** become green when an application uses the command socket on the specified port.

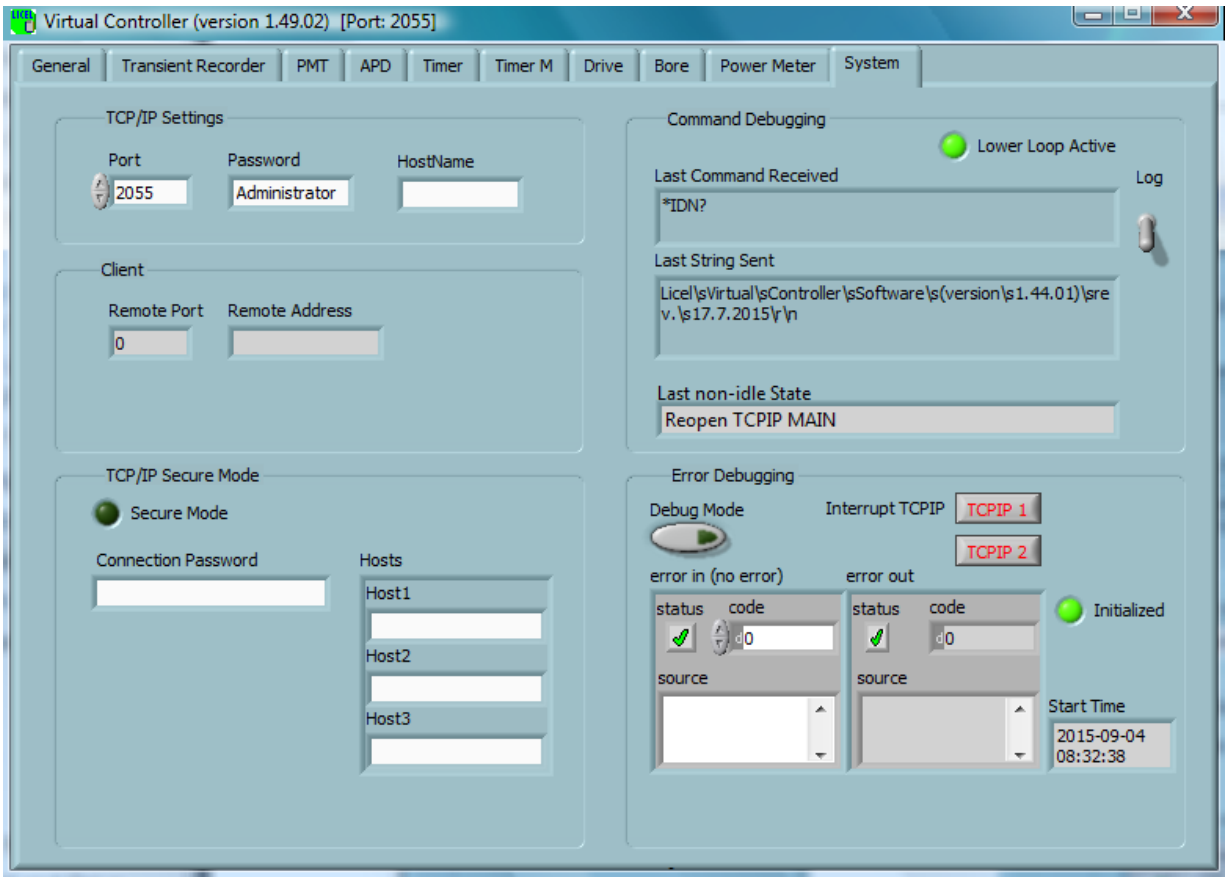
If an application used the second TCP/IP socket on a port number advanced by 1 (push socket), as well, the second indicator would become green. You may test this with the example applications `TCPIP Live Display` with an **Update #** smaller than 15 for 12 bit TR and 1 Shot for 16bit unit or `TCPIP MPush Acquis` from the Licel TCPIP Acquisition Software package without any shot restriction.

Power By clicking the **Power** button the Virtual Controller will be stopped.



8.4 System

On the *System* tab page system and debugging options are available.



TCP/IP Settings In this section the **Port** and the **Password** for the Virtual Controller may be changed. Note that **Port** is used directly for the command socket (TCPIP Connection 1), and **Port**$+1$ is the port for the push socket (TCPIP Connection 2). The **Password** is the administrator password of the Virtual Controller which is needed to be sent with certain commands like the command to change the IP address. Note that a change of the IP address of the Virtual Controller is not possible, as the IP address is set in the computer's system setup. The **Hostname** is used to simulate Ethernet Controllers with the corresponding capability.

Client When a client is connected to the Virtual Controller it's **Remote Port** and **Remote Address** are displayed here.

TCP/IP Secure Mode This section is related to the secure mode and the related TCP/IP commands WHITELIST and ACCESS.

- **Secure Mode** Controls whether or not the secure mode is active. Normally the secure mode is enabled with the ACCESS command, for debugging it can be activated by clicking.
- **Connection Password** The connection password to be used to login.
- **Hosts** The allowed host or address ranges set by the WHITELIST command.

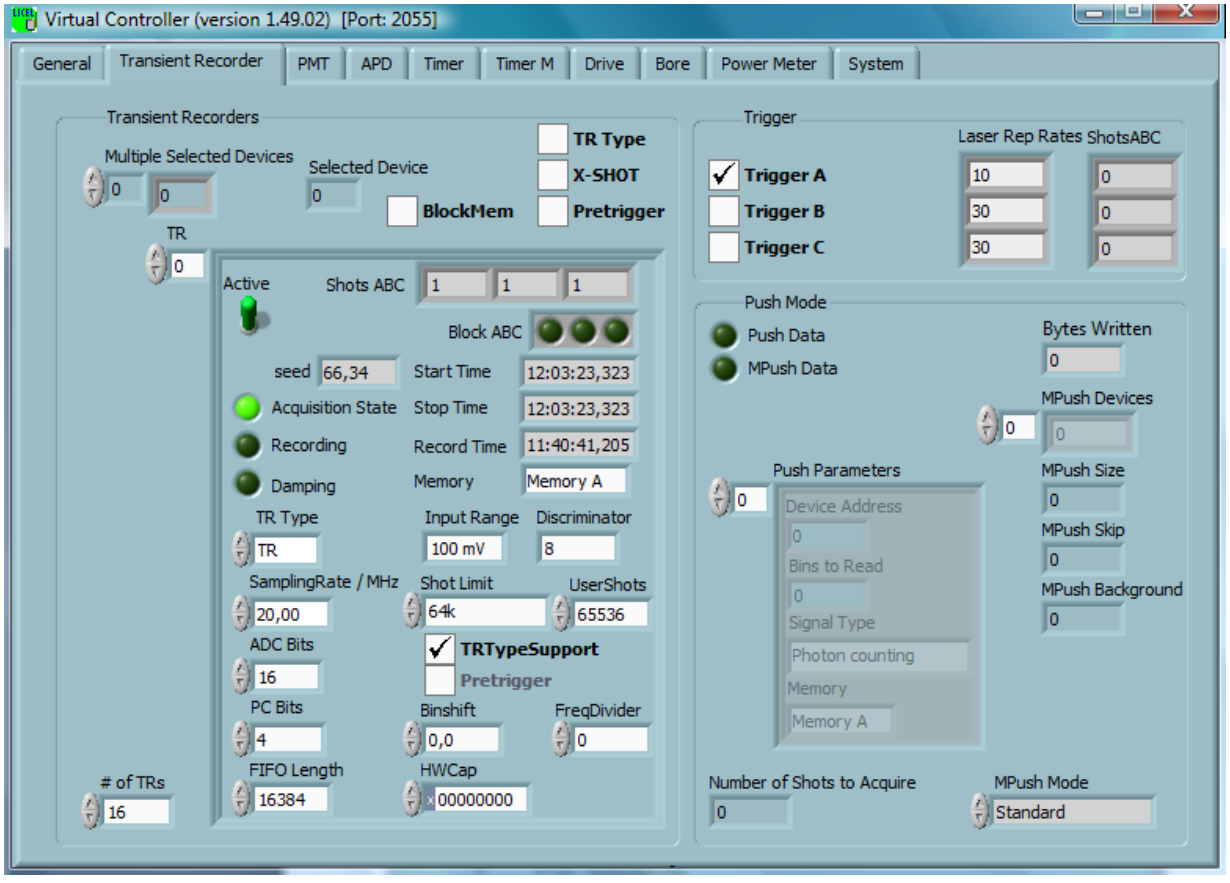
Command Debugging This part of the front panel contains information about the TCP/IP communication flow (**Last Command Received** and **Last String Sent**), and about the loop activities of the Virtual Controller (**Last non-idle State** and **Lower**

Loop Active). **Last non-idle State** describes the internal state of the command socket loop of the program while **Lower Loop Active** indicates that the push socket loop to handle requests on **Port**+1 is alive. This indicator should always light in a green color. Switching the **Log** switch to the up position will make the `Virtual Controller` write all received commands to a log file named `Virtual Controller.exe.log`.

Error Debugging This section contains the standard LabVIEW error clusters. If **Debug Mode** is set errors in the Virtual Controller are shown. Timeout errors (code=56) are normal during operation. The buttons **Interrupt TCPIP** (**TCPIP 1** and **TCPIP 2**) may be used to interrupt the TCP/IP connections to inspect the behavior of an application to such an event. This is especially useful if a mechanism to reconnect has been implemented to the application when TCP/IP relevant errors occur. The LED indicator **Initialized** shows whether or not the program has been initialized.

8.5 Transient Recorder

The controls and information located on this tab page correspond to the TR capability. When a transient recorder setting is changed or while an acquisition is running you will find the corresponding information here.



The following controls and indicators are available:

Left: Transient Recorders

Selected Device The currently `SELECT`ed device number for single functions like `START` and `STOP`.

Multiple Device Numbers An array containing the device IDs of the active transient recorders if any have been selected using `SELECT`.

# of TRs Number of transient recorders at the Virtual Controller.

TR Type Control whether or not the Virtual Controller understands the `TRTYPE?` command.

X-Shot Control whether or not the Virtual Controller understands the extended shot commands `SHOTAB?` and `MSHOTSAB?`.

Pretrigger Control whether or not the Virtual Controller understands the `PRETRIG` command.

BlockMem Control whether or not memory blocking is supported.

TR Contains the current status of the transient recorders. You may switch between the different TRs by changing the device number located top-left of this control. **Tr** contains in detail:

- **Active | OFF** Set the transient recorder at the current index (TR address) active or not (= not installed). This setting is stored in initialization file for future usage.
- **Shots ABC** Number of shots acquired at the memories A,B, and C still stored in the transient recorder that are to be added to the new acquisition.
- **Start Time** Time of the last start.
- **Stop Time** Time of the last stop. If the **Start Time** is larger than the **Stop Time** the transient recorder is acquiring data, otherwise the transient recorder is inactive.
- **Record Time** The Record Time is used to control whether or not a new seed for simulated data generation should be created or not. In this way, the seed for generating the data is not changed if the acquisition is continued instead of a new acquisition being started.
- **Seed** Used for generating random white noise for the simulated lidar signal.
- **Acquisition State** TRUE when the transient recorder returns from the armed state, FALSE, when an aquisition is running.
- **Recording** TRUE during acquisition-time, e.g. the ADC or the photon counting is aquiring data. Recording is FALSE during summation and when the device is waiting for a new trigger.
- **Damping** Indicates the threshhold range in photon counting mode. *Off*: the current discriminator level is used, *On* the current discriminator level multiplied by 4 is used.
- **Block ABC** Indicate whether or not the memory blocking is active for the meories A, B, or C.

Input Range Shows the analog input range of the preamplifier. The signal start at 0 and reaches -20, -100, -500 mV. (0=-500mV,1=-100mV,2=-20mV).

Discriminator For a device with a photon counting unit, the discriminator threshold can be set in 64 steps between 0 and 63.

Memory Shows to which summation memory (Memory A or B) the last acquisition was added.

TR Type Type of the transient recorder: 0 (TR) = analog and photon counting device, 1 (PR) = pure photon counting device. This value is changeable here. This setting is stored in initialization file for future usage.

Sampling Rate Sampling rate in MHz of the transient recorder. This value is changeable here if this information is not stored in the transient recorder. The setting is always stored in initialization file.

ADC Bits Number of ADC Bits. If supported this information will be submitted with the the `TRTYPE?` command.

PC Bits Number of PC Bits. If supported this information will be submitted with the the `TRTYPE?` command.

Shot Limit

- *Not Supported* Setting the shot limit is not supported i.e. the `LIMIT` command returns an error.
- *4k* The shot limit is 4k shots.
- *64k* The shot limit is 64k shots.

UserShots Limit of the `SETMAXSHOTS` command, range 2k . . . 64k.

TRTypeSupport Check this box if the individual simulated transient recorder should supports the `TRTYPE?` command.

Pretrigger Check this box if the individual simulated transient recorder should supports the `PRETRIG` feature.

Bin Shift Bin shift of ADC data bins with respect to photon counting bins. If supported this information will be submitted with the the `TRTYPE?` command.

FreqDivider Freqeuncy divider - control the hardware binning of the transient recorder. Supported only if the 7th bit(HWCap) is set (HWCap AND 0x40).

FIFO Length FIFO length. If supported this information will be submitted with the the `TRTYPE?` command. Currently the Virtual Controller does not use this value when generating and returning data.

HWCap Hardware capabilities of the transient recorder.

- `0x01` shot counter B support
- `0x02` shot counter C support
- `0x04` shot counter D support (not yet used)
- `0x08` pretrigger support (not used in simulations)

- `0x10` block memory support
- `0x20` squared data support
- `0x40` frequency divider

Right Top: Trigger

Checkboxes Here the triggers A, B, and C can be enabled (checked) or disabled

Laser Rep Rates Enter the laser repetition rates (Hz) here

ShotsABC Acquired shots at for memories A, B, and C

Right Bottom: Push Mode

Push Data Shows whether or not the PUSH mode is active.

MPush Data Shows whether or not the multiple push mode MPUSH is active.

Push Parameters Push parameters used for push and mpush modes:

- **Device Address** Device address of the active transient recorder.
- **Bins to Read** The number of bins to read from the appropriate memory.
- **Signal Type** Specifies which part of the raw information should be transfered from the device to the computer.
- **Memory** Shows the memory to read from.

Number of Shots to Acquire The number of shots to acquire in push mode.

MPush Devices An array containing the device IDs of the transient recorders addressed in the MPush mode.

MPush Size Total number of bytes to send in MPush mode.

MPush Skip Number of bins to skip after **MPush Size** in MPush mode as defined by the `MPUSHBACK` command. Defines the background start = MPush Size + MPush Skip.

MPush Background Number of background bins to append in MPush mode as defined by the `MPUSHBACK` command.

MPush Mode The MPush Mode determines whether or not a timestamp is sent with the transmitted data and how the data is organized.

- *No Timestamp* The MPush data does not contain a timestamp (for older controllers).
- *Standard* A timestamp is send in MPush mode. This format is expected by `TCPIP MPush Acquis`
- *MPush AB* The MPush AB mode is available only in customized controllers.

8.5.1 LIDAR Signal Simulation

The Licel Virtual Controller simulates backscatter signals and sends the data via TCPIP to the calling client. Therefore, the Virtual Controller functions in exactly the same way as a real detection system. As a result the software acquisition programs will observe no difference between a simulated and a real signal.
The simulation starts with molecular back scattering for 550 nm at the standard height read from the files `standard_back.txt` and `standard_height.txt`, respectively. A backscatter signal is simulated and the overlap with the optical detection system is calculated. Finally some random white noise is added and the data is prepared to be sent via TCPIP. For the simulation a default laser repetition rate of 10 Hz is assumed, the rate is changeable on the front panel.
The shape of the overlap function is controlled by the coefficients read from the initialization file `Overlap_Globals.ini`:

```
[Overlap]
Beam_Radius=2.000000
Beam_Divergence=0.002000
Beam_Inclination_Parallel=0.000000
Beam_Inclination_Perpendicular=0.000000
Telescope_Diameter=0.600000
Detector_Radius=1.000000
Focal_Length=1800.000000
Axial_Separation=800.000000
Noise_Amplitude=1.000000
Sqrt_Signal_Noise_Amplitude=0.120000
```

These parameters are read when the Licel Virtual Controller is started. They may be changed in the initialization file according to your own optical system to simulate a realistic backscatter signal you will measure with your hardware.

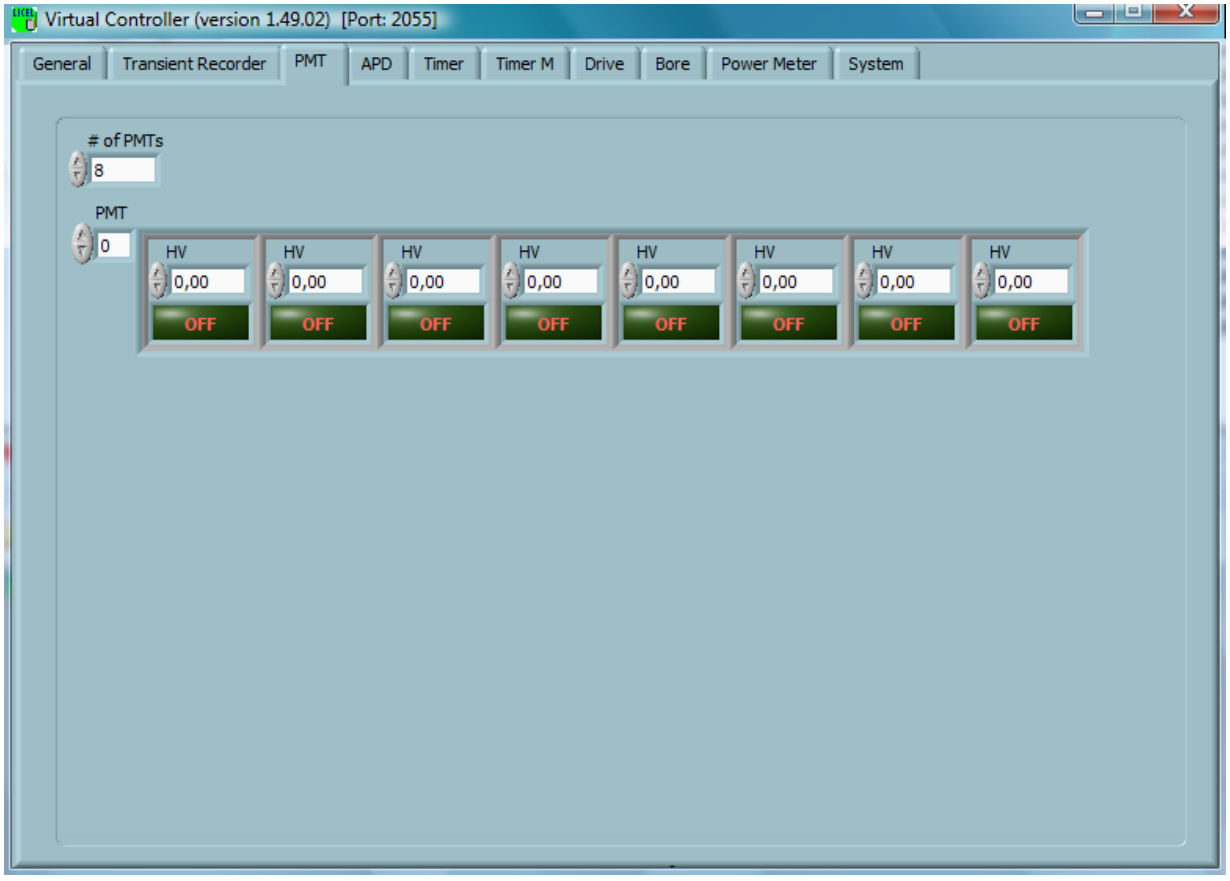
`Beam_Radius`	Radius of the assumed beam in mm
`Beam_Divergence`	Divergence of the assumed laser beam in rad
`Beam_Inclination_Parallel`	Assumed parallel beam inclination in rad
`Beam_Inclination_Perpendicular`	Assumed perpendicular beam inclination in rad
`Telescope_Diameter`	Telescope diameter in m
`Detector_Radius`	Assumed detector radius in mm
`Focal_Length`	Focal length of the receiving telescope in mm
`Axial_Separation`	Axial separation between the receiving telescope and outgoing laser beam
`Noise_Amplitude`	The noise amplitude allows for control of the amplitude of the random white noise added to the Lidar Data. The Lidar Data has a range of 0-2000, 0-10000, or 0-50000 depending upon which input range is taken
`Sqrt_Signal_Noise_Amplitude`	Square root of the signal noise amplitude for the simulation

LabVIEW Source Options

If you purchased the LabVIEW source code for the Licel Virtual Controller you are able to change the optical alignment while an acquisition is active. After having read the initialization file the values are written to global variables. These global variables are accessed at the time the simulated signal is generated. You may change them in `Virtual Globals Overlap Globals.glb.vi` in the library `Virtual Globals.llb`. The simulated signal will immediately follow this ”change” of the optical alignment.
The variable `Virtual Globals Laser Rep Rate.glb.vi` controls the laser repetition rate (located in `Virtual Globals.llb`, as well).
Furthermore you will be able to replace the model used for the simulated backscatter signal. You can substitute the code to generate the backscatter signal with your own model in the vis `Lidar-Sim Simulated Lidar with Noise.vi` and `Lidar-Sim interpolate sigma test.vi`. Both vis are located in the library `lidar-sim.llb`.

8.6 PMT

The controls and information located on this tab page correspond to the PMT capability. When the photomultilpier high voltage is changed you will find the corresponding information here.



The following information is available:

# of PMTs number of simulated PMTs.

PMT An array containing the status information of the PMTs. Each element represents a PMT with the following detailed information:

HV Shows the value of the high voltage power supply for the PMT.

ON/OFF switch Shows whether or not the HV is being applied to the PMT.

8.7 APD

The controls and information located on this tab page correspond to the APD capability. When the APD parameters are changed you will find the corresponding information here.

The following information is available:

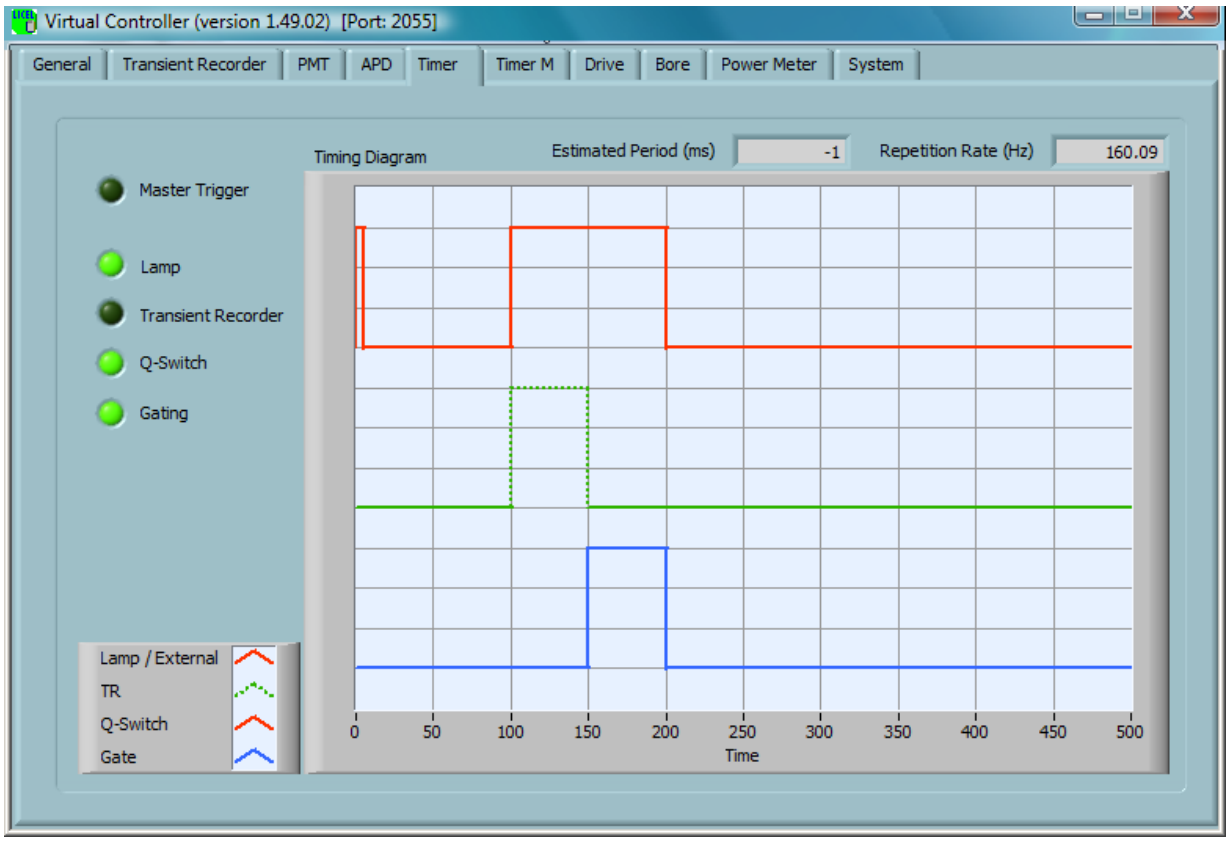
# of APDs Number of simulated APDs.

APD An array containing the status information of the APDs. Each element represents an APD with the following detailed information:

HV	Shows the value of the high voltage power supply for the APD.
ON/OFF switch	Shows whether or not the HV is being applied to the APD.
Temp Regulation	Shows whether or not the APD is being passively or actively cooled.
Temp in Range	Controls whether or not the temperature of the apd is in or out of range. This value may be changed here to test the reaction of a client application.

8.8 Timer

The controls and information located on this tab page correspond to the TIMER, TIMER1,and TIMER2 capabilities. They are changed in response to the `TRIGGERMODE` and `TRIGGERTIME` commands.



The following information is available:

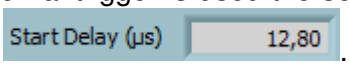
Master Trigger This switch indicates whether the internal trigger of the Licel Trigger Module or an external trigger will be used as master trigger.

Lamp Indicates whether or not the trigger output for the laser lamp is enabled.

Transient Recorder Indicates whether or not the transient recorder (acquisition) will be triggered.

Q-Switch Indicates whether or not the Q-switch output will be triggered.

Gating Indicates whether or not a gate pulse will be generated.

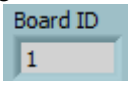
Repetition Rate Frequency in Hz of the internally generated trigger pulses. Here, the repetition rate is displayed in the case that the internal trigger is used as the master trigger. If the external trigger is used the set start delay is shown instead at the same position: 

.

Estimated Period (ms) Estimated periodin milliseconds.

Timing Diagram Timing Diagram displaying the trigger pulses. Please refer to the explanations in the TCP/IP Command List.

Board ID This value is displayed whenever an access is simulated for timing sub-boards corresponding to the TIMER1 or TIMER2 capablilities.

The example 

indicates an access corresponding to TIMER1.

In this example the controller internally generates a trigger (**Master Trigger** dark) with the repetition rate 160.09 Hz. The controller generates lamp, q-switch, and gating pulses.

8.9 Bore

The controls and information located on the tab page *Bore* correspond to the BORE capability to simulate Licel's Bore Site Aligment Detector. This is described in the corresponding manual available at http://www.licel.com/manuals/BoreManual.pdf.

8.10 Power Meter

The controls and information located on this tab page correspond to the POW capability. They are changed in response to the commands to control a Licel Power Meter. Please refer to the Power Meter Control section 7.3.

8.11 TimerM and Drive

The controls and information located on these tab pages correspond to the TIMERM and DRIVE capabilities. They are changed in response to the commands to control a Licel Polarotor. Please refer to the corresponding manual for the Licel Polarotor found at http://www.licel.com/manuals/polarotor.pdf.

Chapter 9

Appendices

9.1 TCP/IP Command List and Syntax

This section lists and describes the TCP/IP command syntax for Licel TCP/IP Ethernet Controllers. Most commands can be sent either in a short form or a long form. In this description the abbreviations TR, PMT, and APD are used to denote a Licel transient recorder, a Licel photomultiplier module, or a Licel avalanche photodiode, respectively. <CRLF> is carriage return line feed. All commands sent to the TR should end with <CRLF>, and all replies from the Licel TCP/IP controller end with <CRLF> which will not explicitly be shown in this document.
If the controller detects an unknown command it will return the string

```
<command> unknown command
```

back to the caller where <command> is the command originally sent.
The following commands are available dependent on the Licel Ethernet Controller you ordered.

Short	Long
	ACCESS
	ALIGNDATA
	ALIGNSIGN
	ALIGNTIME
APD?	APDSTAT?
APDT	APDTEMPERATURE
APDG	APDGAIN
	BLOCK
CAP?	CAP?
CLE	CLEAR
CONT	CONTINUE
DATA	DATA
DISC	DISCRIMINATOR
	DRIVEMODE
	DRIVERESET
	DRIVESPEED
	DRIVESPEED?
	DRIVESTATUS?
	FREQDIV
	FREQDIV?
HOST	
	HOSTNAME?
*IDN?	IDENTIFICAT?

	INTERNALTRIGA
	KILL
	LIMIT
	LOGON
MCL	MCLEAR
MCON	MCONTINUE
	MILLISEC?
MPUS	MPUSH
	MPUSHAB
	MPUSHBACK
	MSHOTS?
	MSHOTSAB?
MSTA	MSTART
MSTO	MSTOP
MWA	MWAIT
PASS	PASS
PMT?	PMTSTAT?
PMTG	PMTGAIN
	POW
	PRETRIG
PUSH	PUSH
RANG	RANGE
SEL	SELECT
	SETMAXBINS
	SETMAXSHOTS
SING	SINGLE
	SHOTAB?
SLAV	SLAVE
STAR	START
STAT?	STATUS
STOP	STOP
TCPIP	TCPIP
THR	TRESHOLD
	TRIGGERMODE
	TRIGGERTIME
	TRIGGERTIMEM
	TRTYPE?
	WHITELIST

ACCESS <LIMIT "Password" "Connection Password" | FREE "Password">

Switches the secure mode on or off.
If used with the keyword `LIMIT` the secure mode is switched on. The administrator password (`"Password"`) and the password for client connections (`"Connection Password"`) have to be transmitted together with the `LIMIT` keyword. Access to the controller is limited to clients operating from hosts specified with the `WHITELIST` command. After establishing his TCPIP connection a client must use the `LOGON` command to login in secure mode. The example

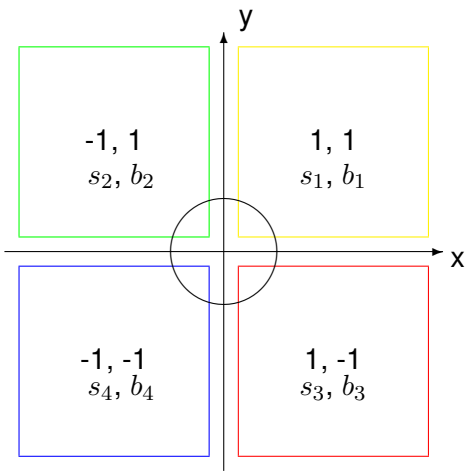
```
ACCESS LIMIT "Administrator" "Connected"
```

will start the secure mode with the `Connection Password Connected` (if the current controller password equals `Administrator`). In case of a non-correct controller password or bad command syntax the controller will return

`ACCESS not accepted`,
other wise the return value is
`ACCESS Limited`.
If the `ACCESS` command is used with the keyword `FREE` the secure mode is switched off. The administrator password (`"Password"`) has to be transmitted together with the `LIMIT` keyword. The response of the controller after a successful `ACCESS` command is
`ACCESS Unlimited`.
Note that one has to establish a secure mode connection using the `LOGON` command (i.e. one has to know the `Connection Password`) before switching the secure mode off with the `ACCESS` command. The only other way to disable the secure mode is a hardware reset.

ALIGNDATA <START EVERY #shot SHOTS #cycle CYCLES>| <STOP>

Starts or stops the controller to send bore site alignment data via the data push socket. After `ALIGNDATA START` The controller will average the involved channels over `#shot` shots and write the data to the data push socket in the following form:
`Align Info: id` s_1 s_2 s_3 s_4 b_1 b_2 b_3 b_4 .
The reply ends with a <CRLF>.
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. $s_{1..4}$ and $b_{1..4}$ are the averaged signal and background values for the involved channels and corresponding to the background and signal regions defined by the ALIGNTIME command, i.e. by `Background Start` and `Background Stop`, and `Signal Start` and `Signal Stop`. This step is repeated for `#cycle` times. If `#cycle` equals -1, the controller must explicitly be stopped by sending `ALIGNDATA STOP` to the controller. The indices of $s_{1..4}$ and $b_{1..4}$ correspond to the Licel Bore Sight Detector as seen in the figure below.
The controller replies
`ALIGNDATA START executed` or `ALIGNDATA STOP executed`.



Sketch of the quadrants of the Licel Bore Sight Detector (cathode)

ALIGNSIGN

Sending this command toggles the counter sign in the data that is send when the bore system is acquiring data. The main purpose is to synchronize the bore data with the alignment moves. Once a movement is finished one could send ALIGNSIGN command and wait that the counter sign in the received data toggles to make sure that the data has been recorded after the last alignment move.
If the command is successful the controller replies:
`ALIGNSIGN executed`.

ALIGNTIME <Background Start><Background Stop><Signal Start><Signal Stop>

Sets the timing parameters for bore site alignment. The parameters correspond to the bins of the acquired data. The region between `Background Start` and `Background Stop` is assumed to correspond to the background, while `Signal Start` and `Signal Stop` define the signal region. The following restrictions apply:

$$
\begin{aligned}
0 < &\ \texttt{Background Start} &&< 1024 \\
\texttt{Background Start} < &\ \texttt{Background Stop} &&< 1024 + \texttt{Background Start} \\
\texttt{Background Stop} < &\ \texttt{Signal Start} &&< 1024 + \texttt{Background Stop} \\
\texttt{Signal Start} < &\ \texttt{Signal Stop} &&< 1024 + \texttt{Signal Start}
\end{aligned}
$$

where `Background Start` and `Background Stop` may pairwise be interchanged with `Signal Start` and `Signal Stop`.
If the command is successful the controller replies:

```
ALIGNTIME executed.
```

An example for the command is

```
ALIGNTIME 400 850 1200 1600.
```

APDSTAT? <Device Number>

APD? <Device Number>

Returns the current status of the APD with the given `Device Number`. For example to get the status of APD number 3 send

```
APD? 3
```

to the controller. The reply is of the following form:

```
APD <Voltage> <HV control state> <temperature regulation>
<T in range?> <T control state>
```

with the values

```
Voltage                 HV voltage
HV control state        HV_local | HV_remote
temperature regulation  T_on | T_off
T in range?             T_in_range | T_out_of_range
T control state         T_local | T_remote.
```

`Voltage` is the gain voltage and indicates whether the power supply of the APD is switched on or off. The `HV control state` indicates whether the APD HV is being controlled locally (`HV_local`) or remotely (`HV_remote`). Valid answers for the `temperature regulation` are `T_on` and `T_off`. If the temperature is in range, then the `T in range?` value is `T_in_range`, otherwise `T_out_of_range` is returned. The `T control state` returns `T_local` or `T_remote`. An example of a reply is

```
APD 750.0 HV_local T_on T_in_range T_remote.
```

In this case the gain voltage is 750.0 volts, the APD HV is controlled locally, and the temperature is being regulated, is in range and remotely controlled. If the APD with the specified device number is not installed the reply is

```
APD 3 is not available
```

where the number 3 is the device number of the non-existent APD. Valid values for the device number are 0-3. `APDSTAT?` works in both remote and local control modes.

APDT <Device Number> <on|off>

APDTEMPERATURE <Device Number> <on|off>

Turns the temperature regulation for the APD specified by `<device number>` either `on` or `off`. For example to turn on the temperature regulation on the APD with device number 3, send

```
APDT 3 on
```

to the controller. The reply is

```
APDT executed
```

If the APD with the specified device number is not installed the reply is

```
APD 3 is not available
```

where the number 3 is the device number of the non-existent APD. Valid values for the device number are 0-3. The long form breaks the SCPI convention since it is longer than 12 characters.

APDG <Device Number> <HV Voltage>

APDGAIN <Device Number> <HV Voltage>

Sets the gain voltage for the specified APD to the given `<HV Voltage >` value. For example to set the gain of APD with device number 3 to 300 Volts, send

```
APDG 3 300
```

to the controller. A successful execution is indicated by the reply

```
APDG executed.
```

If the APD with the specified device number is not installed the reply is

```
APD 3 is not available.
```

where the number 3 is the device number of the non-existent APD. Valid values for the device number are 0-3.

CAP?

Requests the control capabilities of the controller.

The controller's response is

```
CAP: [List of Capabilities],
```

where `List of Capabilities` is a space-separated list with on or more of the following items:

`TR`	for controlling transient recorder
`APD`	for APD remote control
`PMT`	for PMT remote control
`TIMER`	for the trigger timing controller
`CLOUD`	for transient recorder controller cloud mode
`BORE`	Boresight alignment system .

A response could be

```
CAP: TR
```

for a controller which is able to control transient recorders, only, while

```
CAP: APD PMT TIMER
```

indicates a controller capable of controlling APDs PMTs and the timing generator.

CLEAR

CLE

Clears both memories (A and B) of the previously selected transient recorder, if the TR is in SLAVE mode. After sending this command, the controller replies with the string

```
CLEAR executed.
```

If this command is sent while PUSH or MPUSH mode is active, the reply is

```
CLEAR ignored due to active PUSH mode.
```

If the selected TR does not answer, the response will be:

```
CLE failed for TR <TR#>, Can't write.
CLE failed for TR <device number >, <Can't clear Memory >
```

indicates a memory access error to `Memory` (A or B).

CONTINUE

CONT

Continues data acquisition without clearing the memory of the selected transient recorder if the TR is in SLAVE mode. After sending this command the controller replies with the string

```
CONTINUE executed.
```

If this command is sent while PUSH or MPUSH mode is active, the reply is

```
CONTINUE ignored due to active PUSH mode.
```

The error message

```
CONTINUE failed for TR <Device Number>, Can't write
```

is sent if the transient recorder identified by `Device Number` is not responding.

DATA? <Device Number> <Number to Read> <Signal Type> <Memory>

Requests data from the transient recorder with the corresponding `Device Number` if the TR is in SLAVE mode. The `Number to Read` determines the number of bins to be read. The `Signal Type` can be either be `PC`, `MSW`, or `LSW` for photon counting, analog MSW, or analog LSW, respectively. Analog LSW is the default value. If the TR supports squared data `P2L` and `P2M` give the photon counting squared data where `P2L` stands for the lower word and `P2M` for the upper Word. The analog squared data needs to be read in three pieces with the `A2L`, `A2M` and `A2H` argument. Here `A2L` stands for the lower word, `A2M` for the middle and `A2H` for the highest word. The `Memory` can be either `A` or `B`, for memory A or memory B, respectively. As an example, we could have

```
DATA? 6 8000 PC B
```

which would return the first 8000 bins of the photon counting Memory B of transient recorder #6. The controller replies to the DATA? request by returning the data. As the transient recorder's data is an array of 16-bit numbers the returned number of bytes equals twice the number of requested bins. If this command is sent while PUSH or MPUSH mode is active, the reply is

```
DATA? ignored due to active PUSH mode.
```

If `Device Number` is not in range the reply is

```
Device ID <Device Number> is currently not supported.
```

The error message

```
DATA failed for TR <Device Number>, Can't write
```

is sent if the transient recorder #`Device Number` is not responding.

DISCRIMINATOR <Integer>

DISC <Integer>

Sets the discriminator level. Valid values for the discriminator are 0–63. To set the discriminator level to 16, send

```
DISCRIMINATOR 16
```

to the controller. The reply is

```
DISCRIMINATOR set to 16.
```

If the `Integer` value is out of range the reply is

```
DISCRIMINATOR value is out of range.
```

The error message
`DISCRIMINATOR failed for TR <Device Number>, Can't write`
is sent if the transient recorder #`Device Number` is not responding.

DRIVEMODE <DriveMode><Count>

Sets the drive mode for a polarotor controller. `DriveMode` is a number with the following settings encoded into the five least significant bits:

`00000`	0	`STOP`	no motion, stops motion
`01000`	8	`START`	free run, starts motion with the programmed `DRIVESPEED` setting
`11001`	25	`WHILE 1234 HIGH`	run the stepper motor with the programmed `DRIVESPEED` *as long as* the angular position detector is active
`11010`	26	`WHILE 1234 LOW`	run the stepper motor with the programmed `DRIVESPEED` *until* the angular position detector is active
`11011`	27	`WHILE SYNC HIGH`	run the stepper motor with the programmed `DRIVESPEED` *as long as* the synchronization detector is active
`11100`	28	`WHILE SYNC LOW`	run the stepper motor with the programmed `DRIVESPEED` *until* the synchronization detector is active
`11101`	29	`X STEPS`	move the stepper motor `Count` number of steps.

`Count` is used only for the mode `11101`. For the modes with bit 4 set, a small velocity (recommended: `320`) must be set using the `DRIVESPEED` command.
If the command is successfully executed the controller replies
`DRIVEMODE executed.`
The current drive mode can be read with the DRIVESTATUS command.

DRIVERESET

Reset the polarotor controller. This command should be used after the stepper motor of the polarotor has been stopped by `DRIVEMODE 00000` while it was free running.
If the command is successfully executed the controller replies
`DRIVERESET executed.`

DRIVESPEED <Speed>

Sets the velocity for the polarotor's stepper motor in counts per second.
If the command is successfully executed the controller replies
`DRIVESPEED executed.`
The current stepper motor velocity can be read with the `DRIVESPEED?` command.

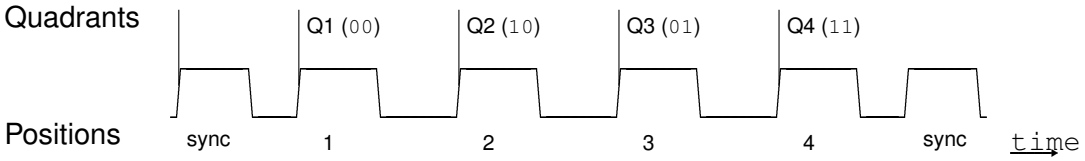
DRIVESPEED?

Reads the programmed stepper motor velocity at the polarotor.
If the command is successfully executed the controller replies
`DRIVESPEED Speed` where `Speed` is the velocity of the stepper motor in counts per second.
The stepper motor velocity can be set with the `DRIVESPEED` command.

DRIVESTATUS?

Returns the status of the polarotor.
If the command is successfully executed the controller replies
`DRIVESTATUS Quadrant SyncLevel 1234Level DriveMode.`

The 2 least significant bits of `Quadrant` contain the quadrant information. `00` is the quadrant beginning with the first angular position following the synchronization position. The sequence of quadrants is encoded by `00` (1) – `10` (2) – `01` (3) – `(`11) (4).
`SyncLevel` indicates whether the synchronization position is active (1) or not (0).
`1234Level` indicates weather an angular position is active (1) or not (0).



`DriveMode` is the current drive mode, a number with the following settings encoded into the five least significant bits:

`00000`	0	`STOP`	stopped
`01000`	8	`START`	free running
`11001`	25	`WHILE 1234 HIGH`	run the stepper motor *as long as* the angular position detector is active
`11010`	26	`WHILE 1234 LOW`	run the stepper motor *until* the angular position detector is active
`11011`	27	`WHILE SYNC HIGH`	run the stepper motor *as long as* the synchronization detector is active
`11100`	28	`WHILE SYNC LOW`	run the stepper motor *until* the synchronization detector is active
`11101`	29	`X STEPS`	move the stepper motor `Count` number of steps.

HOST <"NewHostName"> <"Password">

Set the host name of the Licel Ethernet Controller to `NewHostName`. `Password` is the administrator password (default: *Administrator*, changeable using the PASSWORD command).
Example:

```
HOST "MyNewHostName" "Administrator"
```

gives the reply

```
HOSTNAME changed to MyNewHostName.
```

If a command parameter is not correct the reply is

```
HOST failed.
```

HOSTNAME?

Returns the current host name set by the HOST command:

```
HOSTNAME?
```

gives the reply

```
LICEL
```

if the host name has been set to `LICEL`.

IDENTIFICAT?

*IDN?

Asks the controller to send its identity and firmware revision. The reply from the controller is e.g.

KILL <SOCKETS> <Password>

Causes the controller to close all TCP/IP connections. `Password` is the internal password of the controller. This command can be used only at a TCP/IP connection with the controller on the 3rd supported Ethernet port, i.e. on `Port` + 2 when `Port` is he Ethernet port used for the bidirectional communication. The default is $2055 + 2 = 2057$. If required, the base port can be changed using the TCPIP command, the internal password (default: *Administrator*) can be changed with the PASSWORD command. `KILL SOCKETS` must be sent before reopening the TCP/IP communication with the controller.
Usage:

1. Open a TCP/IP connection to the controller at the 3rd Ethernet port, i.e. `Port` + 2 (default $2055 + 2 = 2057$).
2. Immediately send `KILL SOCKETS` (terminated by <CRLF>).
3. Ignore all communication errors, the controller will close the connection on `Port` + 2, as well.

LOGON <"Encrypted Hexcode">

Is used to log in while the secure mode is active. Directly after establishing the TCP/IP connection with the controller the latter will send two 4 byte unsigned integer numbers in a hex-encoded string. The client has to decode these numbers from the hexadecimal string and use them to encrypt the connection password set by the `ACCESS` command using the Blowfish encryption algorithm. The resulting two 4 byte unsigned integer numbers have to be converted to a hexadecimal string and sent to the controller with the `LOGON` command. While secure mode is active the controller will close the TCP/IP connection without any comment if it does not receive the correct code within 20 seconds.

MCLEAR

MCL

Clears all memories of the SELECTed transient recorders, if the TR is in SLAVE mode. The answer is

```
MCLEAR executed.
```

If this command is sent while PUSH or MPUSH mode is active, the reply is

```
MCLEAR ignored due to active PUSH mode.
```

If a selected TR does not answer, the response will be:

```
MCLEAR failed for TR <TR#>, Can't write.
MCLEAR failed for TR <device number >, <Can't clear Memory >
```

indicates a memory access error to `Memory` (A or B).

MCONTINUE

MCON

Restarts the SELECTed transient recorders without clearing the memories, if the TRs are in SLAVE mode. The reply is

```
MCONTINUE executed.
```

If this command is sent while PUSH or MPUSH mode is active, the reply is

```
MCONTINUE ignored due to active PUSH mode.
```

The error message

```
MCONTINUE failed for TR <Device Number>, Can't write
```

is sent if the transient recorder #`Device Number` is not responding.

MILLISEC?

Requests the millisecond timer value of the controller. The reply is

```
MILLISEC: time
```

where `time` is a number with the milliseconds since the start of the controller.

MPUSH <Shots>
<Device Number> <Number to Read> <Signal Type> <Memory>
[<Device Number> <Number to Read> <Signal Type> <Memory>[...]]

MPUS <Shots>
<Device Number> <Number to Read> <Signal Type> <Memory>
[<Device Number> <Number to Read> <Signal Type> <Memory>[...]]

Causes the controller to enter a state where data of `Signal Type Memory` is directly pushed from the transient recorder(s) `Device Number` to the computer. The `Signal Type` can be either `PC`, `MSW`, or `LSW` for photon counting, analog MSW, or analog LSW, respectively. The `Memory` can be either `A` or `B`, for memory A or memory B, respectively. The transient recorders acquire n shots, n is given by `Shots` and in opposition to the PUSH Mode there is only the internal shot limit of the TR (4094 by default).

After having acquired the requested number of `Shots` the controller reads `Number to Read` bins from the corresponding `Memory` and `Signal Type` from each transient recorder with the given `Device Number`. Additionally, the controller reads the `Background Bins` defined by the MPUSHBACK command. Signal and Background are combined and sent to the computer. As the transient recorder's data is an array of 16-bit numbers the returned number of bytes equals twice the number of requested bins. The data have a header consisting of 2 marker bytes `0xFF` and a 4 byte integer with a timestamp defined as the milliseconds since the start of the controller. The data sets for each transient recorder are preceded by the number of shots as a 16-bit number. Note that the number of shots has an offset of 2 caused by the clear shots. The length of each device-specific data set has to be known by the acquiring computer.

Then, the transient recorders automatically continue to collect data sets for pushing them to the computer. The SLAVE command stops the MPUSH command. The example

```
MPUSH 5 1 8000 PC B 4 6000 LSW A
```

would cause the data from the transient recorders 1 and 4 to be pushed to the data acquisition computer after recording 5 shots. From device 1, 8000 bins of the photon counting Memory B will be sent. From device 4, 6000 bins of analog LSW memory A will be sent. Having sent the data the TRs will automatically be restarted by the controller and the next set of data will be acquired and sent. The reply is

```
MPUSH executed.
```

If the command syntax is not correct the controller replies

```
MPUSH syntax is wrong,
```

if the PUSH mode is active the controller will return

```
MPUSH ignored due to active PUSH mode.
```

If the number of shots is not in range the controller returns

```
Illegal Push shot number.
```

MPUSHBACK <Skip> <Background Bins>

Defines the background for the data in the MPUSH mode. `Skip` is the number of bins to skip between the signal bins (`Number to Read`) and the first background bin. `Background Bins` is the number of background bins the controller will read and attach to the `Number to Read` signal bins in the MPUSH mode.

if the command is correctly executed the controller replies

```
MPUSHBACK executed.
```

MSHOTS? [<numMemories >]

Returns the acquired shots and summation memories of the last acquisitions of all SELECTed transient recorders.
The reply is

```
MSHOTS [shots(0) memory(0) ...  shots(n-1) memory(n-1)]
```

where n transient recorders have been SELECTed before. shotsi contains the decimal number of shots of the i^{th} selected transient recorder and memory(i) is either 0 (memory A) or 1 (memory B).

The optional argument numMemories will require the information about the the shots that have been acquired into the different memories. If 2 is choosen for numMemories the response will be

```
MSHOTS [shotsA(0) shotsB(0) ...  shotsA(n-1) shotsB(n-1)]
```

MSTART

MSTA

starts the SELECTed multiple TRs, if the TRs are in SLAVE mode. As an example

```
MSTART
```

would start selected Devices. The reply is

```
MSTART executed.
```

If this command is sent while PUSH or MPUSH mode is active, the reply is

```
MSTART ignored due to active PUSH mode.
```

The error message

```
MSTART failed for TR <Device Number>, Can't write
```

is sent if the transient recorder #Device Number is not responding,

```
MSTART failed for TR <device number >, <Can't clear Memory >
```

indicates a memory access error to Memory (A or B).

MSTOP

MSTO

Stops the SELECTed multiple TRs, if the TRs are in SLAVE mode.

```
MSTOP
```

will stop the currently selected devices. The reply is

```
MSTOP executed.
```

If this command is sent while PUSH or MPUSH mode is active, the reply is

```
MSTOP ignored due to active PUSH mode.
```

The error message

```
MSTOP failed for TR <Device Number>, Can't write
```

is sent if the transient recorder #Device Number is not responding.

MWAIT <Timeout in ms>

MWA <Timeout in ms>

Waits until all SELECTed TRs stop or until the timeout time is exceeded. The range for Timeout in ms is between 0 and 400. If all TRs are ready within the timeout limit, the reply is

```
MWAIT executed.
```

If Timeout in ms is not in range the controller replies

```
MWAIT failed delay:  <Timeout in ms> should be between 0 and 400ms.
```

If a selected TR does not answer, the response will be:

```
MWAIT failed for TR <TR#>, Can't write.
```

PASSWORD <"Old Password"> <"New Password"> <"New Password">

PASS <"Old Password"> <"New Password"> <"New Password">

Changes the password for the controller. The actual password is required to change the IP configuration of the transient recorder. The user needs to enter the old password and then the new password twice. The default password is "'*Administrator*"'. The password will be reset to this if a hardware reset is executed on the controller. For example

```
PASS "Administrator" "MyPassword" "MyPassword"
```

will change the password to MyPassword. The controller replies with

```
PASSWORD set to "MyPassword",
```

if an error occurs (wrong `Old Password`, nonequal `New Password` entries, or empty `New Password`s) the reply is

```
PASSWORD not set.
```

PMT? <Device Number>

PMTSTAT? <Device Number>

Returns the status of the PMT with the specified device number. The reply parameters are <HV value in Volts> <HV on/off> <local/remote>. For example for requesting the status of the PMT with the device number 5 send

```
PMT? 5
```

to the controller. An example of a reply is

```
PMT 970 on remote
```

which indicates that the PMT is in remote mode, the HV power supply is on and is set to 970 Volts. Another example would be

```
PMT 30 off local:
```

here, the PMT is being controlled locally, the HV power supply is off and it is set to 30 volts, which is the default return value when the PMT is off. If the PMT with the specified device number is not installed the reply is

```
PMT 5 is not available
```

where the number 5 is the device number of the non-existent PMT. Valid values for the device number are 0-7.

PMTG <Device Number> <Voltage>

PMTGAIN <Device Number> <Voltage>

This command sets the gain voltage applied to the dynodes of the PMT with the specified device number. For example

```
PMTG 3 980
```

will set the gain voltage to 980 volts. The reply is

```
PMTG executed.
```

If the PMT with the specified device number is not installed the reply is

```
PMT 3 is not available
```

where the number 3 is the device number of the non-existent PMT. Valid values for the device number are 0-7.

POW <Command> [Number]

Submit a command to a Power Meter Controller. The usage of `Number` depends on the `Command`. The reply is

`POW <Command> executed` for all values of `Command` excepted `TRACE` (see below).

The following values for `Command` are supported:

`CHANNEL` Selects the ADC channel for the data acquisition. The ADC channel `Number` can be either be 0 (photodiode) or 2 (power meter head).
example: `POW CHANNEL 0` selects the ADC channel 0. The controller response is:
`POW CHANNEL executed`.

`START` (`Number` is not used) Activates the data acquisition and data transmission over the previously opened push socket. The controller response is:
`POW START executed`.
For every received trigger one ASCII line will be sent with the following format
`<Milliseconds since controller start> <Pulse amplitude><CRLF>`

`STOP` (`Number` is not used) Deactivates the data acquisition and stops the data transmission over the push socket. The controller response is:
`POW STOP executed`.

`TRACE` (`Number` is not used) Starts a single pulse acquisition and returns one pulse in the following ASCII format:
`<Number of points:N>` $<Y_0>$ $<Y_1>$ ... $<Y_{N-1}>$`<CRLF>`.

PUSH <Shots> <Number to Read> <Signal Type> <Memory>

Causes the controller to enter a state where data from `Signal Type Memory` is directly pushed from the SELECTed transient recorder to the computer. The `Signal Type` can either be `PC`, `MSW`, or `LSW`, for photon counting, analog MSW (default), or analog LSW, respectively. The `Memory` can either be `A` (memory A) or `B` (memory B). The example

`PUSH 3 8000 PC B`

would return the first 8000 bins of the photon counting Memory B after 3 shots have been acquired. The controller will start an acquisition of n `Shots`. n is limited to a maximum value of 14. After having acquired the requested number of `Shots` the controller reads `Number to Read` bins from the corresponding `Memory` and `Signal Type` from the transient recorder and sends them to the computer. As the transient recorder's data is an array of 16-bit numbers the returned number of bytes equals twice the number of requested bins. These data have a header consisting of 2 marker bytes `0xFF` followed by the number of shots as a 16-bit number. Note that the number of shots has an offset of 2 caused by the clear shots. Then, the controller forces the TR to collect the next data for pushing it to the computer. The SLAVE command stops the PUSH command. The reply is

`PUSH executed.`

if the MPUSH mode is active the controller will return

`PUSH ignored due to active MPUSH mode.`

If data from more than one transient recorder or more than the LSW should be pushed to the acquisition computer the MPUSH command should be used.
If the number of shots is not in range the controller returns

`Illegal Push shot number.`

RANGE <0|1|2>

RANG <0|1|2>

Sets the input range to either -500mV (0), -100mV (1), or -20mV (2). The command

`RANGE 0`

sets the input range to -500mV. The TR replies with

```
RANGE set to -500mV.
```

If an illegal value for the range is submitted to the controller the reply is

```
Illegal Range Value.
```

The error message

```
RANGE failed for TR <Device Number>, Can't write
```

is sent if the transient recorder #`Device Number` is not responding.

SELECT <Device Number List>

SEL <Device Number List>

Selects or unselects the active transient recorders. The parameter `<Device Number List>` is a comma-separated list of transient recorder numbers or -1 to unselect all selected transient recorders. For example to activate transient recorder #8, send

```
SELECT 8
```

If a TR with the given device number is available the answer by the controller is

```
SELECT 8 executed
```

To select more than one TR, separate the transient recorder numbers with a comma. For example, to select the transient recorders 1, 3, 8, and 12, send

```
SELECT 1, 3, 8, 12
```

Note that the separator is a comma and the empty spaces between the TRs will be ignored. Thus,

```
SELECT 1,3,8,12
```

is equivalent to the previous command. The answer by the controller is

```
SELECT 1, 3, 8, 12 executed
```

if any device number is out of range, the controller does not execute the command while replying

```
Device ID %d is currently not supported,
```

where `%d` is the first illegal device number. To unselect the active transient recorders send

```
SELECT -1
```

to the controller, the reply is

```
SELECT executed.
```

SINGLE

SING

Clears the TR memory and causes the transient recorder to take a single shot. The reply is

```
SINGLE executed
```

If this command is sent while PUSH or MPUSH mode is active, the reply is

```
SINGLE ignored due to active PUSH mode.
```

An access error at a transient recorder with the device number `<device number>` is indicated by

```
SINGLE failed for TR <device number>, Can't write.
```

SLAVE

SLAV

Ends the PUSH or MPUSH mode. The reply is

```
SLAVE executed.
```

START

STAR

Clears the memories and starts the data acquisition of the selected transient recorder, if the TR is in Slave mode. After sending this command, the controller replies with

```
START executed.
```

If this command is sent while PUSH or MPUSH mode is active, the reply is

```
START ignored due to active PUSH mode.
```

The error message

```
START failed for TR <Device Number>, Can't write
```

is sent if the transient recorder #`Device Number` is not responding,

```
START failed for TR <device number >, <Can't clear Memory >
```

indicates a memory access error to `Memory` (A or B).

STATUS?

STAT?

Returns the current status of the `SELECT`ed transient recorder. If more than one TR is selected, the Status of the selected TR with the lowest device number is returned (e.g. when 5, 7, 12, 14 are selected, then `STAT?` returns the status of TR #5). The values returned are the `shotnumber`, the `acquisition state`, the `recording` state, and the transient recorder's summation memory of the last acquisition. The `shotnumber` is returned as an integer. The acquisition state can be either armed or disabled, hence the string `Armed` is returned if the TR is armed, otherwise an empty string is returned. Whether or not the TR recorder is collecting data is shown by the recording state. If the TR is storing data in its memory the string `Acquiring` is returned or an empty string. The summation memory can either be Memory A or Memory B. If Memory B has been used for the last acquisition `MemB`, otherwise an empty string is returned. An example reply would be

```
Shots 8032 Armed Acquiring
```

indicating that the TR has acquired 8032 shots, is armed and currently accumulating data. Another example is

```
Shots 8032 Armed,
```

here, TR has acquired 8032 shots of data, is armed and is not storing data.
The error message

```
STAT? failed for TR <Device Number>, Can't write
```

is sent if the transient recorder #`Device Number` is not responding,

STOP

Stops the data acquisition of the selected transient recorder, if the TR is in Slave mode. After sending this command, the controller replies with the string

```
STOP executed.
```

If this command is sent while PUSH or MPUSH mode is active, the reply is

```
STOP ignored due to active PUSH mode.
```

The error message

```
STOP failed for TR <Device Number>, Can't write
```

is sent if the transient recorder #`Device Number` is not responding.

TCPIP <"ip#"> <"subnet mask"> <"Gateway"> <"Port"> <"Password">

TCP <"ip#"> <"subnet mask"> <"Gateway"> <"Port"> <"Password">

Sets the IP address, subnet mask, gateway and Ports that are used for TCP connections. Please note that the port numbers `Port`, `Port` + 1 and `Port` + 2 are used by the controller. This command will only be executed if the password corresponds with the controller's internally stored password. The defaults are

IP Address	10.49.234.234
Subnet Mask	255.255.255.0
Gateway	empty
Port	2055 .

In this case port 2055, port 2056, and port 2057 are used by default. Port 2055 is used for the bidirectional communication with the controller. The communication on port 2056 is monodirectional and contains the data that is pushed to the acquisition computer when it is in PUSH or MPUSH mode. Furthermore, port 2057 is used to enforce the controller to close all TCP/IP connections on the other ports (KILL SOCKETS). In order to restore the default values, the reset button needs to be pressed when powering up the controller (hardware reset). The default password is "`Administrator`." To change the password, see the PASS command. For example

```
TCPIP "197.13.17.23" "250.250.250.29" " " "2013" "Administrator"
```

will change the IP Address to 197.13.17.23, the Subnet mask to 250.250.250.39, the gateway would be empty and the ports 2013 and 2014 would be used. The controller replies

```
IP "197.13.17.23" Subnet "250.250.250.39" Gateway " " Port "2013"
executed.
```

If the password is incorrect, then the reply is

```
TCPIP failed due to invalid password.
```

TCPIP "DHCP" <"Port"> <"Password">

TCP "DHCP" <"Port"> <"Password">

Enable DHCP mode on the network controller. The controller will listen at the specified port and at Port+1. This command will only be executed if the password corresponds with the controller's internal password. If not

```
TCPIP failed due to invalid password
```

will be returned. If the command is successfully executed the controller replies

```
DHCP activated.
```

The controller comes with the defaults described for the TCPIP IP command. A hardware reset will disable the DHCP mode.

THRESHOLD <0|1>

THR <0|1>

Sets the damping state to either on or off. If a value of 1 is sent then damping is turned on. If a value of 0 is sent, the damping is turned off. To turn Damping on, send

```
THRESHOLD 1,
```

to turn the damping back off, send

```
THRESHOLD 0.
```

The controller replies with either

```
THRESHOLD executed :  Damping on
```

or

```
THRESHOLD executed :  Damping off.
```

The error message

`THRESHOLD failed for TR <Device Number>, Can't write`
is sent if the transient recorder #`Device Number` is not responding.

TRIGGERMODE[BoardID] <mode >

Enable/Disable the trigger in and outputs at Trigger and Polarotor controllers. `BoardID` is a board identifier which is used only if the Licel Trigger Module is equipped with more than 1 timing unit. Then `BoardID` is in the range 1 . . . number of timing sub-boards. `BoardID` is not used for the polarotor timing (but used for additional timing sub boards) in a Polarotor Controller.
Mode is a bitfield where for every set bit the corresponding output is enabled:

	Trigger Controller	Polarotor Controller
0x01	`Laser Lamp trigger`	`Laser Lamp trigger`
0x02	`Pretrigger (Acquisition)`	`Acquisition +`
0x04	`Q-Switch`	`Q-Switch`
0x08	`Gating`	`Acquisition ||`
0x10	`Master Trigger`	`Master Trigger`

.
If the Master trigger bit is set an external trigger will be accepted, if not the internal trigger will be used. The internal trigger will be controlled via the `RepetitionRate` in the `TRIGGERTIME` command. If successful the controller will return:
`TRIGGERMODE executed`
otherwise the returned string is
`TRIGGERMODE failed.`
If the parameter is out of the range (not a byte) the reply is:
`TRIGGERMODE: invalid parameter.`

TRIGGERTIME[BoardID] <RepetitionRate|StartDelay> <Pretrigger> <PretriggerLength> <QSwitch> <QswitchLength> <TriggerPeriod>

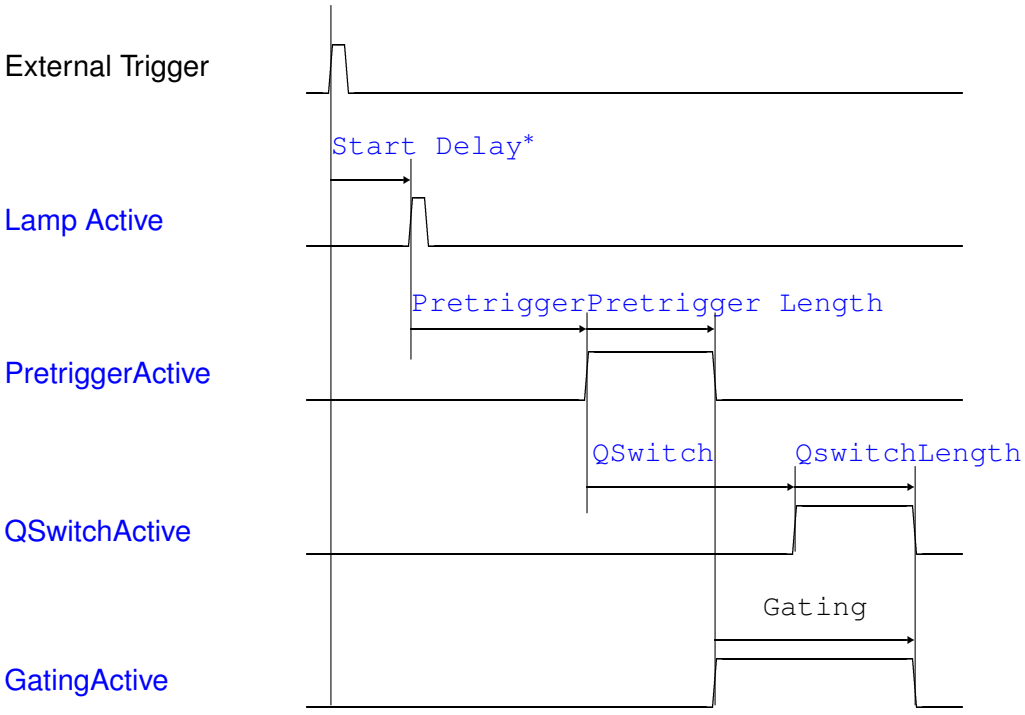
Set the timing parameter in ns at a Trigger Controller. `BoardID` is a board identifier which is used only if the Licel Trigger Module is equipped with more than 1 timing unit. Then `BoardID` is in the range 1 . . . number of timing sub-boards.

`RepetitionRate|StartDelay`	in internal mode delay between two pulses in ns, in external mode start delay between external trigger and lamp in ns. The `StartDelay` is supported at controlers newer than 2007-06
`Pretrigger`	delay between lamp and pretrigger (acquisition) in ns
`PretriggerLength`	length in ns of the pretrigger (acquisition) pulse
`QSwitch`	delay between pretrigger (acquisition) start and Q-Switch start in ns
`QswitchLength`	length in ns of the Q-Switch pulse.
`TriggerPeriod`	estimated time in milliseconds between subsequent laser triggers .

If successful the controller will return:
`TRIGGERTIME executed,`
in the case that the parameters cannot be interpreted the reply is
`TRIGGERTIME: incorrect or invalid parameters.`
The Gating pulse will be high from the end of the Pretrigger pulse till the end of the Q-Switch Pulse. The duration is
`Gate = QSwitch + QswitchLength - PretriggerLength.`

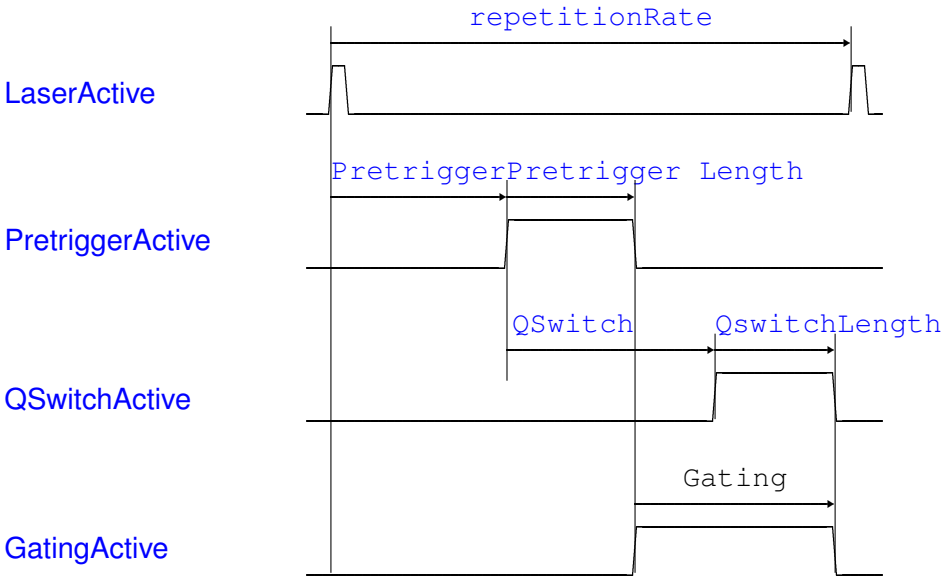
Timing Parameter Explanation

External trigger (MasterTrigger = True)



∗ Available at controllers newer than 2007-06

Internal trigger (MasterTrigger = False)



The Laser Lamp pulse has a fixed length of 5μs.

TRIGGERTIMEM<RepetitionRate> <Pretrigger> <PositionTriggersEnable> <QSwitch> <StartDelay> <TriggerPeriod>

Set the timing parameters and position triggers in ns for a Polarotor Controller.

RepetitionRate	in internal mode delay between two pulses in ns, in external (chopper) mode ignored.
Pretrigger	delay between lamp and pretrigger (acquisition) in ns
PositionTriggersEnable	the 4 least significant bits of this number represent the trigger switches (0 = off, 1 = on) of the triggers at the 4 angular positions of the polarotor. Bit 0 corresponds to the quadrant 1 (00) (see the DRIVESTATUS command)
QSwitch	delay between pretrigger (acquisition) start and Q-Switch start in ns
StartDelay	start delay between internal or external (chopper) trigger and lamp in ns
TriggerPeriod	estimated time in milliseconds between subsequent laser triggers .

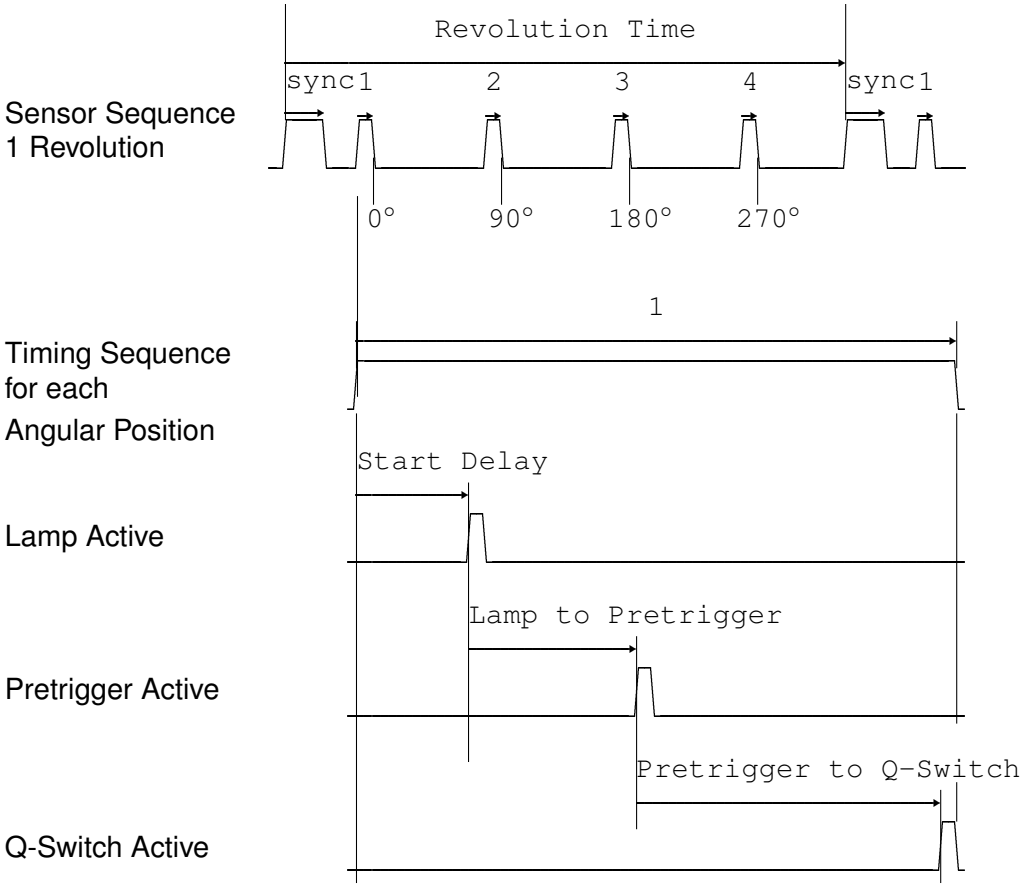
If successful the controller will return:

`TRIGGERTIMEM executed,`

in the case that the parameters cannot be interpreted the reply is

`TRIGGERTIMEM: incorrect or invalid parameters.`

Polarotor Timing Parameters



TRTYPE?

Returns hardware information about the SELECTed transient recorder. If more than one TR is selected, the Status of the selected TR with the lowest device number is returned (e.g. when 5, 7, 12, 14 are selected, then `TRTYPE?` refers to TR #5).

If successful the controller will return the number of ADC bits, the number of PC bits, the length of the FIFO, the binwidth in meters, and the ID of the device in controllers delivered after 11-2013 followed

by a number representing further hardware capabilities of the transient recorder and the the binshift in primary bins. In the example

```
TRTYPE ADC Bits 12 PC Bits 4 FIFOLength 16384 binwidth 3.75 ID 0
HWCAP 0x00000000 binshift 0.0
```

the SELECTed transient recorder is a TR 40-160 with 12 ADC bit device, 4 PC bits, a FIFO length of 16384, a binwidth of 3.75 meters, and the ID 0. No further hardware capabilities (0x00000000) are available. The hardware capabilities are bitwise coded into the returned hexadecimal number: bit 0 indicates whether a separate shotcounter for memory B is available, bit 1 indicates whether a memory C with a separate shotcounter is available. The example is completed by the binshift 0.0 (in primary bins). Further hardware capabilities are (Nov. 2018)

bit 3 separate shot counter D,

bit 4 pretrigger,

bit 5 memory blocking,

bit 6 squared data support,

bit 7 freq divider.

The TRTYPE? request is implemented since spring 2011.

WHITELIST <"Password"> <"Host1"> <"Host2"> <"Host3">

Sets the allowed hosts for secure mode operation. Password is the controller password, and Host# is either

- a host specified by its IP address xx.xx.xx.xx ,
- an IP address range xx.xx.xx.255 ranging from 0 to 255, or
- empty .

The example

```
WHITELIST "Administrator" "192.168.69.255" "213.198.20.19" ""
```

grants secure mode access to clients operating from any IP address between 192.168.69.0 and 192.168.69.255, and furthermore from the IP address 213.198.20.19. If successful the controller will return:

```
WHITELIST executed ,
```

in case of an invalid password or syntax the reply is

```
WHITELIST not accepted .
```

The secure mode must be enabled after specifying the allowed hosts using the ACCESS command. Clients will have to log in using LOGON.

MSHOTSAB?

Returns the shots in each of the memories of all SELECTed transient recorders.
The reply is MSHOTSAB [shotsA(0) shotsB(0) ... shotsA(n-1) shotsB(n-1)]
where n transient recorders have been SELECTed before. shotsAi contains the decimal number of shots of the i^{th} selected transient recorder that have been acquired to memory A and shotsBi those that have been aquired to memory B.
The return values will be only valid if the hardware capabilities of the transient recorder support this. This is indicated by the lowest bit in the HWCAP field of the TRTYPE? command.

SHOTAB?

Returns the shots in each of the memories of the previously SELECTed transient recorder.
The reply is

```
SHOTAB shotsA shotsB
```

shotsA contains the decimal number of shots of the selected transient recorder that have been acquired to memory A and shotsB those that have been aquired to memory B.
The return values will be only valid if the hardware capabilities of the transient recorder support this. This is indicated by the lowest bit in the HWCAP field of the TRTYPE? command.

MPUSHAB

starts a single shot acquisition over multiple transient recorders. For each shot the trigger source will be recorded (trigger A or trigger B).
The mode is activated by

```
MPUSHAB <tr> <bins> <signal type> [<tr> <bins> <signal type>[...]]
```

The command is acknowledged by

```
MPUSHAB executed
```

or rejected with

```
MPUSHAB syntax is wrong
```

or if no push socket has been previously opened with

```
Push socket not ready for transmission.
```

The push mode data will flow over the push socket and show a header field
0xFF;
0xFF;
8 byte time stamp (double)
shot number(2 byte)
Mem A/B (1 byte) ¡...data (16bit wide) ....¿
shot number(2 byte)
Mem A/B (1 byte) ¡...data (16bit wide) ....¿
The push mode is stopped by sending the SLAVE command.

INTERNALTRIGA

Causes a internal trigger and can be used for software testing purposes.
The command is acknowledged by

```
INTERNALTRIGA executed
```

LIMIT <64K|4K >

Switches the maximum number of shots acquired by the transient recorder between 65536 and 4096 shots, where the initial clearing shots are also included. The command will work for 16 bit transient recorders starting from 2011. Starting from 2014 the SETMAXSHOTS allows arbitrary shot numbers for newer transient recorders.
If the transient recorder supports the command, the response will be:

```
LIMIT executed
```

otherwise

```
LIMIT not supported for TR <TRNumber >
```

SETMAXBINS <numMaxBins>

Sets the tracelength the memory configuraton switch 5 is in the ON Position. A user defined tracelength allows a better usage of the acquisition time for high repetition rate systems.

```
SETMAXBINS executed
```

if the command fails

```
SETMAXBINS failed error:  <error number >
```

SETMAXSHOTS <numMaxShots>

Sets the maximum number of shots that the TR should acquire. Per default this number is 4096. The command allows a determined number of shots other than 4096. The functionality is similiar to the LIMIT command but it allows a arbitrary number between 2 and 64K. The command returns if successfully executed

```
SETMAXSHOTS executed
```

or if the command fails

```
SETMAXSHOTS failed error:  <error number >
```

PRETRIG <1|0 >

The command will only work if the hardware capabilities of the transient recorder support this. This is indicated by bit 3 in the HWCAP field of the TRTYPE? command. The pretrigger is 128 bins long. 1 enables the pretrig and 0 disables it. The startup state for a transient recorder is disabled.
The command returns if successfully executed

```
PRETRIG executed
```

or if the command fails

```
PRETRIG failed error:  <error number >
```

BLOCK OFF | A | B | C | D

The command will only work if the hardware capabilities of the transient recorder support this. This is indicated by bit 5 in the HWCAP field of the TRTYPE? command.

If A, B, C or D the transient recorder will ignore trigger for those memories. The typical use case is when the rack trigger A and B are driven but a certain channel should be active only when trigger A or B arrives. BLOCK OFF unblocks all memories.
The command returns if successfully executed

```
BLOCK executed
```

or if the command fails

```
BLOCK failed error:  <error number >
```

FREQDIV?

The command will only work if the hardware capabilities of the transient recorder support this. This is indicated by bit 7 in the HWCAP field of the TRTYPE? command. The command will retrieve actual the frequency divider Possible values are 0-7. Its basically 1<<freqDividerExponent. , to get the actual bin width multiply the bin width returned by TRTYPE with the 1 <<freqDividerExponent. Possible values are 0-7.

The command returns if successfully executed

```
FREQDIV <freqDividerExponent> 0
```

or if the command fails

```
FREQDIV? failed error:  <error number >
```

FREQDIV

The command will only work if the hardware capabilities of the transient recorder support this. This is indicated by bit 7 in the HWCAP field of the `TRTYPE?` command. Set the frequency divider, it changes the sampling rate before the summation, so with `freqDivExponent` 0 and bin width 3.75m you will get 3.75m range resolution. With a `freqDividerExponent` of 3 you will get 30m range resolution. Possible values are 0-7. Its basically `1<<freqDividerExponent`.

The command returns if successfully executed

```
FREQDIV <freqDividerExponent> 0
```

or if the command fails

```
FREQDIV failed error:  <error number >
```

9.2 Data File format

This appendix describes the file format written by `TCPIP Acquis`. The files are interoperable between the different platforms. The file format is a mixed ascii-binary format where the first lines describe the measurement situation, below follow the dataset description and then raw data as 32-bit integer values itself.

9.2.1 Sample file header

```
lc1980211.305684
Berlin   02/08/2019 11:30:23 02/08/2019 11:30:56 0045 0013.384373 0052.542185 00.
0001000 0010 0001000 0010 12 0000000 0000 0000000 0000
0001000 0010 0001000 0010 12 0000000 0000 0000000 0000
1 0 2 02000 1 0800 7.50 00366.o 0 0 00 000 12 001000 0.500 BT0
1 2 2 02000 1 0800 7.50 00366.o 0 0 00 000 12 001000 0.500 S2A0
1 1 2 02000 1 0800 7.50 00366.o 0 0 00 000 00 001000 3.1746 BC0
1 3 2 02000 1 0800 7.50 00366.o 0 0 00 000 00 001000 3.1746 S2P0
1 0 1 02000 1 0800 7.50 00532.o 0 0 00 000 12 001000 0.500 BT0
1 1 1 02000 1 0800 7.50 00532.o 0 0 00 000 00 001000 3.1746 BC0
1 0 1 02000 1 0800 7.50 00532.o 0 0 00 000 12 001000 0.500 BT1
1 1 1 02000 1 0800 7.50 00532.o 0 0 00 000 00 001000 3.1746 BC1
```

Line 1

Filename string, format: `?[?]YYmddHH.MMSSuu[u]`

- *?(?)* - The first 1 or 2 letters can freely be chosen
- *YY* - two digits showing the years in the century
- *m* - the month (hexadecimal, one digit) (December=C)
- *dd* - the day (decimal, two digits)
- *HH* - the hour (decimal, 24 hours per day, two digits)
- *.* - a period (.)
- *MM* - the minute (decimal, two digits)
- *SS* - the seconds (decimal, two digits)
- *uu(u)* - two or three decimal places of the seconds (decimal, two or three digits)

Line 2

Location string with 8 characters

Start Time `dd/mm/YYYY HH:MM:SS`

Stop Time `dd/mm/YYYY HH:MM:SS`

Height a.s.l. four digits (meter)

Longitude 11 digits (including - sign). six digits for decimal grades.

Latitude 11 digits (including - sign). six digits for decimal grades.

zenith angle	four digits in degrees, 1 decimal place
azimuth angle	four digits in degrees, 1 decimal place
info	a custom information field enclosed by quotation marks (””) [optional]

Line 3

Laser 1 Number of shots	*integer 7 digits*
Pulse repetition frequency for Laser 1	*integer 4 digits*
Laser 2 Number of shots	*integer 7 digits*
Pulse repetition frequency for Laser 2	*integer 4 digits*
number of datasets in the file	*integer 2 digits*
Laser 3 Number of shots	*integer 7 digits*
Pulse repetition frequency for Laser 3	*integer 4 digits*
0000000 0000	*(reserved)*
timestamp	*timestamp of the controller corresponding to the data in milliseconds [optional]*

Dataset description

Active	*1 if dataset is present, 0 otherwise*
Dataset type	*0 ≡ Analog, 1 ≡ Photon Counting, 2 ≡ Analog squared, 3 ≡ Photon Counting squared*
Laser source	*one digit Laser 1 ≡ 1, Laser 2 ≡ 2, Laser 3 ≡ 3.*
Number of bins	*5 digits*
Laser polarization	*none ≡ 0, vertical ≡ 1, horizontal ≡ 2, right circular ≡ 3, left circular ≡ 4*
PMT high voltage	*four digits in Volt*
bin width	*in meters four digits including decimal separator (.) and decimal places*
Laser wavelength	*in nm, five digits period*
Polarisation	*one letter, o ≡ no polarisation, s ≡ perpendicular, l ≡ parallel*
0 0	*backward compatibility*
bin shift, whole-number	*bin shift (primary bins, integer rounded down, 2 digits, 00 if not supported or zero)*
bin shift, decimal places	*decimal places of the bin shift (3 digits, 000 if not supported or zero)*
number of ADC bits	*2 digits, in case of an analog dataset, otherwise 0*
number of shots	*6 digits*
analog input range/discriminator level	

	analog input range in Volt in case of analog dataset , discriminator level in case of photon counting, one digit period 3/4 digits.
Dataset descriptor	*BT* $\equiv$ *analog dataset, BC* $\equiv$ *photon counting, the number is the address of the transient recorder as a hexadecimal number, PD* $\equiv$ *Powermeter (Photodiode), PM* $\equiv$ *Powermeter (Powermeter), S2A and S2P* $\equiv$ $s\sqrt{N(N-1)}$ *(analog, PC, respectively,* $s =$ *sample standard deviation,* $N =$ *shot number).*
info	*a custom information field enclosed by quotation marks ("") [optional]*

The data set description is followed by an extra `CRLF`. The datasets are 32bit integer values. Datasets are separated by `CRLF`. The last dataset is followed by a `CRLF`. These `CRLF` are used as markers and can be used as check points for file integrity.

9.3 Standard Deviation Data

This appendix describes how standard deviation data is saved for transient recorders supporting the acquisition of squared data.
The sum of counts c at each bin after acquiring N shots is read from the transient recorders (standard photon counting or analog memories):

$$c_{\text{bin}} = \sum_{i=0}^{N-1} x_i \tag{9.1}$$

with the counts x_i of the ith shot.
The mean value μ (counts per shots) acquired in a bin is then

$$\mu = \frac{c_{\text{bin}}}{N} = \frac{\sum_{i=0}^{N-1} x_i}{N}.$$

The *sample standard deviation* is (https://en.wikipedia.org/wiki/Standard_deviation):

$$s = \sqrt{\frac{\sum_{i=0}^{N-1}(x_i - \mu)^2}{N-1}}. \tag{9.2}$$

From there we get the *standard error of the mean* (https://en.wikipedia.org/wiki/Standard_error):

$$\sigma_\mu \approx \frac{s}{\sqrt{N}}. \tag{9.3}$$

As the x_i are the counts for each shot, but the counts are already summed by the transient recorders there is no direct access to standard deviation nor to standard error when reading from standard analog or PC memories.
The solution is to read squared data c^2_{bin} from transient recorders supporting the summation of squared counts. Then one can calculate the sample standard deviation using

$$\begin{aligned} s &= \sqrt{\frac{c^2_{\text{bin}} - (c_{\text{bin}})^2/N}{N-1}} \\ &= \sqrt{\frac{\sum_{i=0}^{N-1} x_i^2 - (\sum_{i=0}^{N-1} x_i)^2/N}{N-1}} \\ &= \sqrt{\frac{N\sum_{i=0}^{N-1} x_i^2 - (\sum_{i=0}^{N-1} x_i)^2}{N(N-1)}} \\ &= \frac{1}{\sqrt{N(N-1)}}\sqrt{N\sum_{i=0}^{N-1} x_i^2 - (\sum_{i=0}^{N-1} x_i)^2} \end{aligned} \tag{9.4}$$

(https://en.wikipedia.org/wiki/Algorithms_for_calculating_variance).
The acquired squared data sqr_{bin} returned from the transient recorders is

$$sqr_{\text{bin}} = \sum_{i=0}^{N-1} {x_i}^2. \tag{9.5}$$

With this and the acquired data from eq. 9.1 the right square root in eq. 9.4 can be written as

$$sqd_{\text{bin}} = \sqrt{N\sum_{i=0}^{N-1} x_i^2 - (\sum_{i=0}^{N-1} x_i)^2} \tag{9.6}$$

$$= \sqrt{N sqr_{\text{bin}} - (c_{\text{bin}})^2}. \tag{9.7}$$

This will fit into 4 byte numbers used in the standard data files and will therefore be saved by TCPIP Acquis.
Reading sqd_{bin} back from the data files will enable to restore the *sample standard deviation* s and the *standard error of the mean* σ_μ as

$$s = \frac{sqd_{\text{bin}}}{\sqrt{N(N-1)}} \text{ and} \tag{9.8}$$

$$\sigma_\mu = \frac{s}{\sqrt{N}}. \tag{9.9}$$

In TCPIP Acquis and the Viewer software data according to eq. 9.9 is displayed.

9.4 The Initialization File acquis.ini

9.4.1 TR configuration

The initialization file acquis.ini contains definition sections for each transient recorder. The data here corresponds to the values set while configuring the transient recorders. The data entries may appear in a different order within a section named [TR<address>]. Here, the section for the transient recorder with the device address 0 is shown:

```
[TR0]
Discriminator = 0
Range = 0
PM = 0
WavelengthA = 532.000000
PolarisationA = 0
AnalogA = TRUE
A-binsA = 16000
A-reductA = 0
PC A = TRUE
P-binsA = 16000
P-reductA = 0
WavelengthB = 1024.000000
polarisationB = 0
Analog B = FALSE
A-binsB = 0
A-reductB = 0
PC B = FALSE
PC-binsB = 0
PC-reductB = 0
SamplingRate = 20
TRType = 0
TriggerFractionA = 1
TriggerFractionB = 1
Threshold = 0
WavelengthApc = 555.000000
WavelengthBpc = 555.000000
PolarisationApc = 0
PolarisationBpc = 0
PM1pc = 450.000000
PM2 = 450.000000
PM2pc = 450,000000
ShotLimit = 0
ADCBits = 12
PCBits = 4
HWCAP = 0
Binshift = 0.000000
Analog C = FALSE
A-binsC = 16380
A-reduct-C = 0.000000
PC C = FALSE
P-binsC = 16380
P-reductC = 0.000000
WavelengthC = 555.000000
```

```
WavelengthCpc = 555.000000
PolarisationC = 0
PolarisationCpc = 0
PM3 = 450.000000
PM3pc = 450.000000
ID = 0
MemoryDepth = 4
Pretrigger = 0
BlockMemory = FALSE
UserBins = 16384
```

A section always begins with `[TR<n>]` where n indicates the address of the transient recorder.

`Discriminator` Discriminator level between 0 and 63.

`Range` Input range of the transient recorder. Valid values are

Range Value	Input Range
0	0 – 500 mV
1	0 – 100 mV
2	0 – 20 mV

`PM` Photomultiplier voltage (analog, memA).

`WavelengthA` Wavelength 1 (analog).

`PolarisationA` corresponding polarization 1 (analog). Valid values are

Range Value	Input Range
0	no polarization
1	parallel
2	crossed

`AnalogA` (`TRUE` | `FALSE`) Enable or disable analog acquisition for memory A.

`A-binsA` Corresponding number of bins to be read out. The maximum number of bins is given by 16380/($2^{datareduction}$)

`A-reductA` Corresponding data reduction level.

`PC A` (`TRUE` | `FALSE`) Enable or disable photon counting acquisition for memory A.

`P-binsA` Corresponding number of bins to be read out. The maximum number of bins is given by 16380/($2^{datareduction}$)

`P-reductA` Corresponding data reduction level.

`WavelengthB` Wavelength 2 (analog).

`polarisationB` Corresponding polarization 2 (analog). Valid values are as above for `PolarisationA`.

`Analog B` (`TRUE` | `FALSE`) Enable or disable analog acquisition for memory B.

`A-binsB` Corresponding number of bins to be read out. The maximum number of bins is given by 16380/($2^{datareduction}$)

`A-reductB` Corresponding data reduction level.

`PC B` (`TRUE` | `FALSE`) Enable or disable photon counting acquisition for memory B.

`PC-binsB` Corresponding number of bins to be read out. The maximum number of bins is given by 16380/($2^{datareduction}$)

`PC-reductB` Corresponding data reduction level.

`SamplingRate` Sampling rate of the transient recorder.

`TRType` Type of the transient recorder. 0: TR, transient recorder with analog and photon counting acquisition capabilities, 1: PR, pure photon counting device. The type must correspond to the hardware you are addressing (TRxx-xx or PRxx-xx, respectively).

`TriggerFractionA,B` Fraction of the number of shots used for memories A and B. These parameters are neglected if an acquisition uses only one memory. If N shots have been acquired at the transient recorder the number of shots for the channel j is calculated as $N\frac{\text{TriggerFraction}_j}{\sum_{A,B}\text{TriggerFraction}_i}$. The fractions must be set according to the ratio of the trigger frequency inputs for memories A and B at the corresponding transient recorder.

`Threshold` Threshold set in the configuration. Currently not used.

`WavelengthApc` Wavelength 1 (photon counting).

`WavelengthBpc` Wavelength 2 (photon counting).

`PolarisationApc` Polarization 1 (photon counting). Valid values are as above for `PolarisationA`.

`PolarisationBpc` Polarization 2 (photon counting). Valid values are as above for `PolarisationA`.

`PM1pc` Photomultiplier voltage (photon counting, memA).

`PM2` Photomultiplier voltage (analog, memB).

`PM2pc` Photomultiplier voltage (photon counting, memB).

`ShotLimit` Shot limit.

`ADCBits` ADC bits.

`PCBits` PC bits.

`HWCAP` Transient recorder capabilities, for documentation, only.

`Binshift` Bin shift of the transient recorder.

`Analog C` (`TRUE` | `FALSE`) Enable or disable analog acquisition for memory C.

`A-binsC` Corresponding number of bins to be read out. The maximum number of bins is given by 16380/($2^{datareduction}$)

`A-reduct-C` Corresponding data reduction level.

`PC C` (`TRUE` | `FALSE`) Enable or disable photon counting acquisition for memory C.

P-binsC	Corresponding number of bins to be read out. The maximum number of bins is given by 16380/($2^{datareduction}$)
P-reductC	Corresponding data reduction level.
WavelengthC	Wavelength 3 (analog).
WavelengthCpc	Wavelength 3 (photon counting).
PolarisationC	Polarization 3 (analog). Valid values are as above for PolarisationA.
PolarisationCpc	Polarization 3 (photon counting). Valid values are as above for PolarisationA.
PM3	Photomultiplier voltage (analog, memC).
PM3pc	Photomultiplier voltage (photon counting, memC).
ID	Transient recorder ID if available, for documentation.
MemoryDepth	Memory depth if supported.
Pretrigger	(0 | 1) Pretrigger support if available, for documentation.
BlockMemory	(TRUE | FALSE) Memory blocking support.
UserBins	User bins if available

To completely disable a transient recorder AnalogA, PC A, Analog B, and PC B must be set to FALSE.
Important Note: Customers with at least one transient recorder shipped before October 2009 must add for each old transient recorder the following initialization file key to the acquis.ini file when using the TCP/IP 2.44 (Windows or LabVIEW) software or higher:

```
[TR<address>]
PC_Device=FALSE
```

M-Acquis (multi-rack software) users must be aware that the address numbers in the initialization file sections [TR<address>] of transient recorders in further racks are the original device addresses increased by $16 \times$ rackIndex. So the block [TR16] corresponds to the transient recorder with device address 0, block [TR17] corresponds to the transient recorder with device address 1 in the second rack.

9.4.2 TCP/IP Settings

These values are interpreted only in the case that the Windows applications (.exe) are in use. LabVIEW users should enter the appropriate TCP/IP values and save them as default.

```
[TCPIP]
UseValues = TRUE
Port = 2055
IPAddress = 10.49.234.234
;; NoOfControllers is for the multi-rack acquis (M-Acquis)
NoOfControllers = 2

;; [TCPIP00], [TCPIP01], ... are for the multi-rack acquis (M-Acquis)
[TCPIP00]
```

```
UseValues = TRUE
Port = 2055
IPAddress = 10.49.234.234

[TCPIP01]
UseValues = TRUE
Port = 2055
IPAddress = 10.49.234.235

...
```

UseValues, Port, and IPAddress in the section [TCPIP] are used by TCPIP Acquis.exe. The 2-rack software M-Acquis.exe uses NoOfControllers = 2 in the [TCPIP] section and the values in the sections [TCPIP00] and [TCPIP00] for the connections to rack 1 and rack 2, respectively.

9.4.3 Global Configuration Values

In software versions older than TCP/IP 2.50 these values had been stored in the separate initialization file global info.ini.

```
[global_info]
Location = "Berlin"
Longitude = 13,384373
Latitude = 52,542185
Height_asl = 45,000000
working_directory = "/C/temp"
first_letter = "a"
frequency1 = 10,000000
frequency2 = 10,000000
frequency3 = 10,000000
Zenith = 0,000000

[global_info_Laser1]
Laser1_Wavelength0 = 532,000000
```

The values are updated when saving the configuration, wavelengths will be added or removed.

9.4.4 Miscellaneous Parameters

Other parameters are automatically written to acquis.ini, e.g. the number of power meters:

```
[Powermeter]
Number = 0
```

9.5 Monitoring and Controlling TCPIP Acquis from Outside

TCP/IP Server

The basic functions of the TCPIP Acquis software can be accessed from third party applications via TCP/IP. For this TCPIP Acquis implements a TCP/IP Server listening on a defineable port. To activate the TCP/IP server the following initialization file keys in `Acquis.ini` have to be aligned:

```
[TCPIP_API]
Active = TRUE
Port = 2058
```

If `Active` is set `TRUE` a listener will be started using the specified TCP/IP port (`Port = 2058`). The following table lists the supported commands:

Command	Parameter	Description	Reply
VER?		Return the version number as displayed in the Windows title bar	VER <version>
SHOTS	<number>	set the target shot number	SHOTS executed
RECORDS	<number>	set the number of acquisitions (0: unlimited)	RECORDS executed
START		start a single acquisiton	START executed
STOP		stop a running acquisiton	STOP executed
AUTO	<0|1>	start (0) or stop (1) multiple acquisitons	AUTO executed
FILE?		request the name of the last written file	FILE <filepath>
STATUS?		request the current acquisition status	STATUS AUTO <0|1> SHOTS <shots> RECORDS <records>
DIRECTORY	<directory>	set the storage directory of the data files	DIRECTORY executed
SETFIRSTLETTER	<firstletter>	set the first letter of the data files	SETFIRSTLETTER executed
SETFREQ	<number> <frequency>	set the frequency of the laser with the given number (1,2, or 3)	SETFREQ executed
SETZENITH	<zenithAngle>	set the zenith angle	SETZENITH executed
DATASET	<trAddress> <memory> <mode> <active> <noBins> <reduction> <wavelength>	Define a dataset to acquire at the transient recorder with the address trAddress with the given parameters. memory = A|B|C, mode = AN|PC (analog or photon counting), active = 0 (disable) or 1 (enable and use for the next acquisition), noBins, reduction = number of (primary) bins to read with the given reduction, wavelength in nm	DATASET executed
QUIT		exit the acquisition program	QUIT executed

Queue Control

The basic functions of the LabVIEW version of the TCPIP Acquis software can be accessed from third party LabVIEW VIs using LabVIEW's named queue mechanism. Since version 2.31 TCPIP Acquis uses a listening queue named ACQUIS_LISTEN to accept commands, and a reply queue ACQUIS_REPLY to send answers to the commands received via the listening queue. If your TCPIP Acquis is controlled by this queue mechanism please remember to wait for the reply to the command you sent.
The following table lists the queue commands:

Command	Parameter	Description	Reply
SETSHOT	<number>	set the target shot number	SETSHOT <number> executed
SETACQ	<number>	set the number of acquisitions (0: unlimited)	SETACQ <number> executed
START		start a single acquisiton	START executed
STOP		stop a running acquisiton	STOP executed
STARTMULTI		start multiple acquisitons	STARTMULTI executed
GETSHOT		request the current number of shots	SHOT <number>
GETACQ		request the current number of acquisitions	ACQ <number>
GETFILE		request the name of the last written file	FILE <filepath>
SETFIRSTLETTER	<firstletter>	set the first letter of the data files	SETFIRSTLETTER <firstletter> executed
SETDIR	<directory>	set the storage directory of the data files	SETDIR <directory> executed
SETFREQ	<number> <frequency>	set the frequency of the laser with the given number (1,2, or 3)	SETFREQ <number> <frequency> executed
SETZENITH	<zenithAngle>	set the zenith angle	SETZENITH <zenithAngle> executed
DATASET	<trAddress> <memory> <mode> <active> <noBins> <reduction> <wavelength>	Define a dataset to acquire at the transient recorder with the address trAddress with the given parameters. memory = A|B|C, mode = AN|PC (analog or photon counting), active = 0 (disable) or 1 (enable and use for the next acquisition), noBins, reduction = number of (primary) bins to read with the given reduction, wavelength in nm	DATASET executed

Command	Parameter	Description	Reply
`QUIT`		exit the acquisition program	`QUIT executed`

Notifier to Send the File Name

Whenever `TCPIP Acquis` saves a file it will use a named notifier `AQUIS_FILE` to report the full path of the written file. This mechanism can be used by other LabVIEW VIs to monitor whether or not new acquired data is available in the file system for further processing.

9.6 Analysis Example: Gluing Analog and Photon Counting Data

Abstract

The algorithm for combining analog and photon counting data (gluing) is described. A discussion when the signals need to be combined is followed by stepwise procedure to do this with real data.

9.6.1 Introduction

The Licel transient recorder systems have a parallel analog and photon counting detection chain. The combination of both signals gives the high linearity of the analog signal for strong signals and the high sensitivity of the photon counting for weak optical signals. The integration of both detection mechanism into a single device avoids ground loops and other problems that make the combination otherwise cumbersome. The main idea of the signal combination is that there is a region where both signals are valid and have a high signal to noise ratio. For typical Mini-PMT that region extends from 0.5 to 10 MHz in the photon counting. To combine (glue) both signals, the photon counting needs a dead time correction. There are two typical dead-time scenarios, while the Licel photon counter can be best described as nonparalyzable.

9.6.2 Paralyzable System

$$N = S\exp(-S\tau_d) \tag{9.10}$$

Where:
N - is the observed count rate
S - is the true count rate
τ_d - is the system dead time

9.6.3 Nonparalyzable System

$$N = \frac{S}{1 + S * \tau_d} \tag{9.11}$$

N - is the observed count rate
S - is the true countries
τ_d - is the system dead time

While the paralyzable case is nonlinear equation, the nonparalyzable case can be easily inverted to

$$S = \frac{N}{1 - N * \tau_d} \tag{9.12}$$

As both cases are only a theoretical model, they are valid for lower count rates but fail when $S * \tau_d$ becomes larger than one. From a numerical point of view Eq. 9.12 can be only applied to a signal as long as

$$S < \tau_d \tag{9.13}$$

As an example the correction factor for a time constant of 4ns and a observed count rate of 5 MHz is 1.02. As typical averaged maximum observed count rate is 160MHz the correction factor would be 2.77. This would imply an maximum count rate of 470MHz. The glued profiles however show a virtual count rate in the 2GHz region for a 20mV peak.

9.6.4 The glueing algorithm

In the valid region of both signals between the lower toggle rate (typical 0.5MHz) and the upper toggle rate (typical 10MHz) one seeks the linear regression coefficients to transfer the analog data into photon counting data:

$$\sum_{i=1}^{n} (PC(z_i) - (a * Analog(z_i) + b))^2 = min \tag{9.14}$$

The coefficients a and b are applied to the analog signal and above the upper toggle rate the scaled analog is used and below the photon counting data.

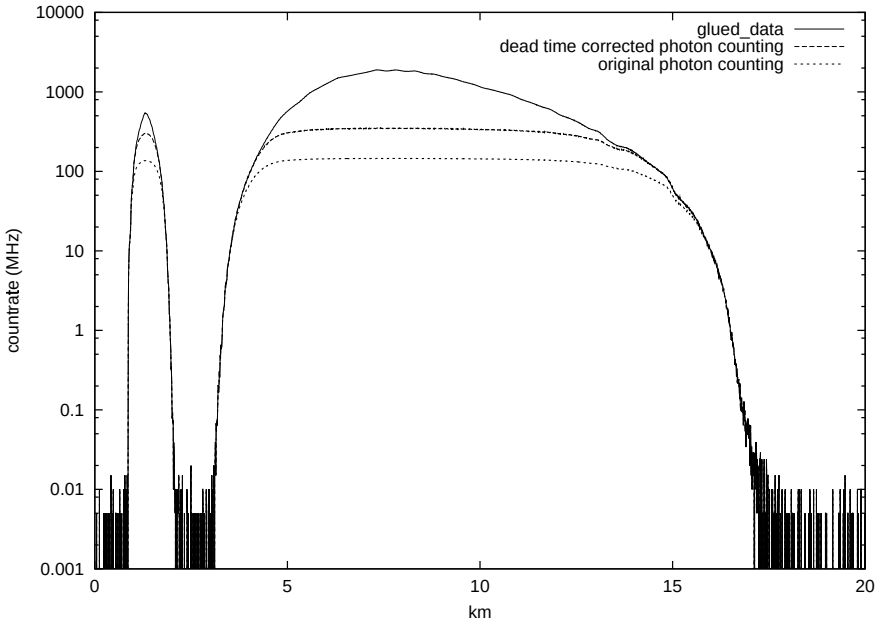


Figure 9.1: Glued data

The zoomed plot shows that the dead time correction function is valid up to 130 MHz.

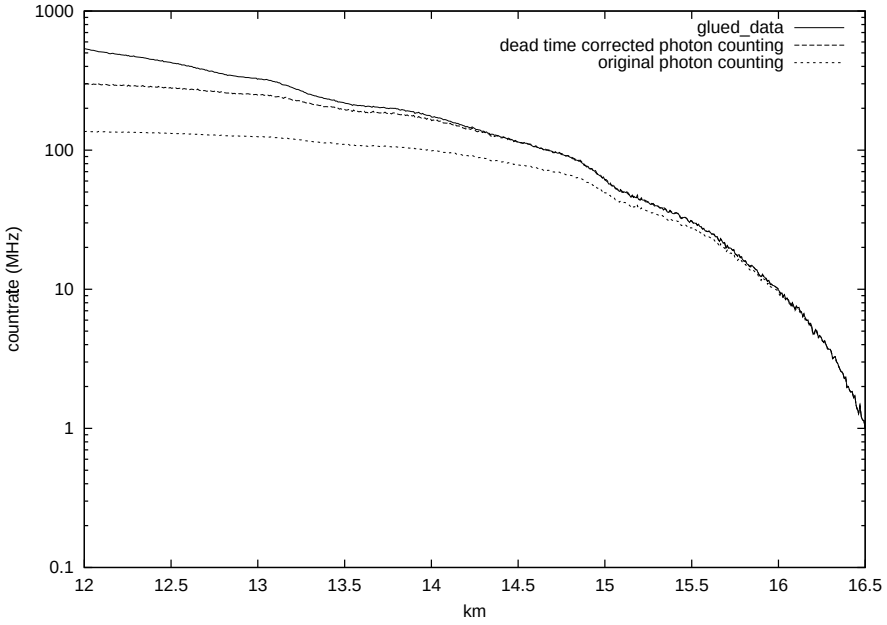


Figure 9.2: Zoomed plot

If one varies the upper toggle frequency between 5 and 10 MHz the standard deviation for the signal maximum is only 3MHz or 0.1%. This proves the numerical stability of the proposed algorithm.
The figure below shows the necessity of applying the dead-time correction first. Without correction the signal maximum becomes stronger dependent from the max. toggle rate.

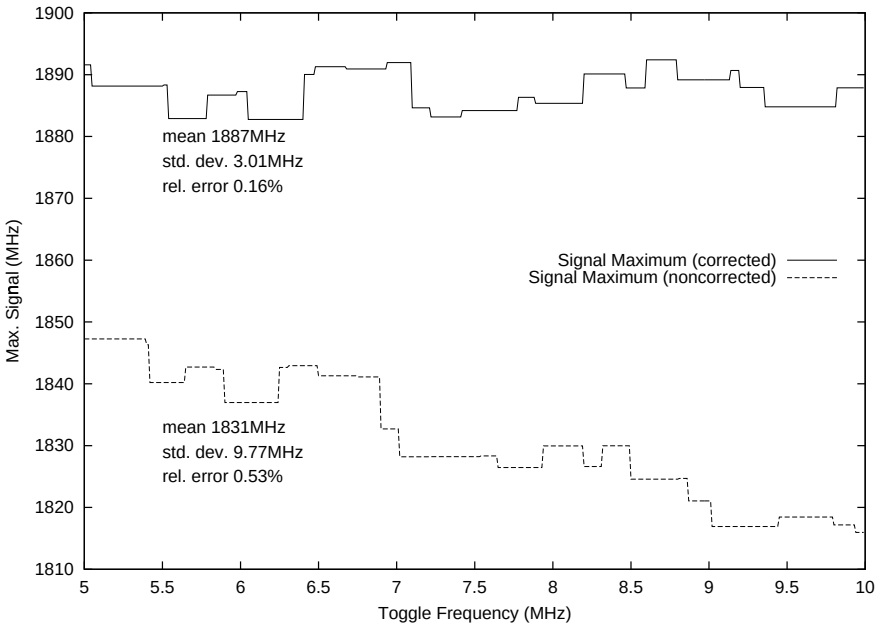


Figure 9.3: Signal maximum for different max. toggle frequencies without dead-time correction

Figure 9.4 demonstrates the advantages of the photon counting in the low light level region. While the analog signal shows the noise coming from the ADC, the photon counting is still able to follow the input signal and extends the dynamic signal range from the analog signal by another 2 orders of magnitude.

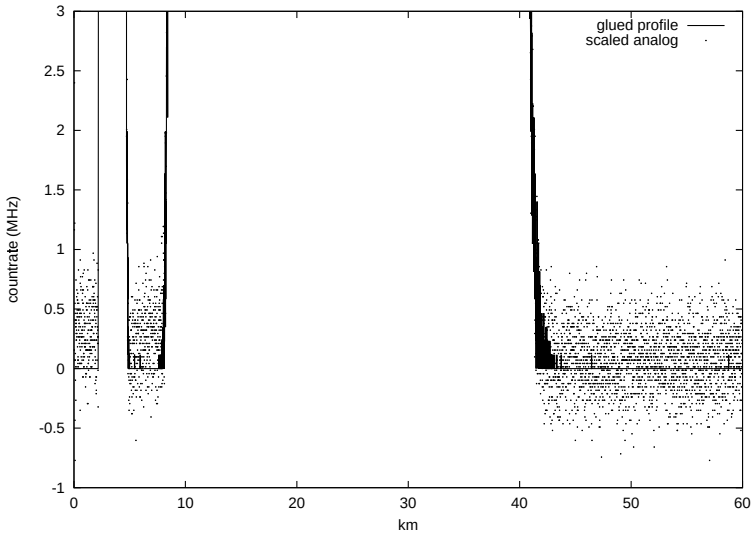


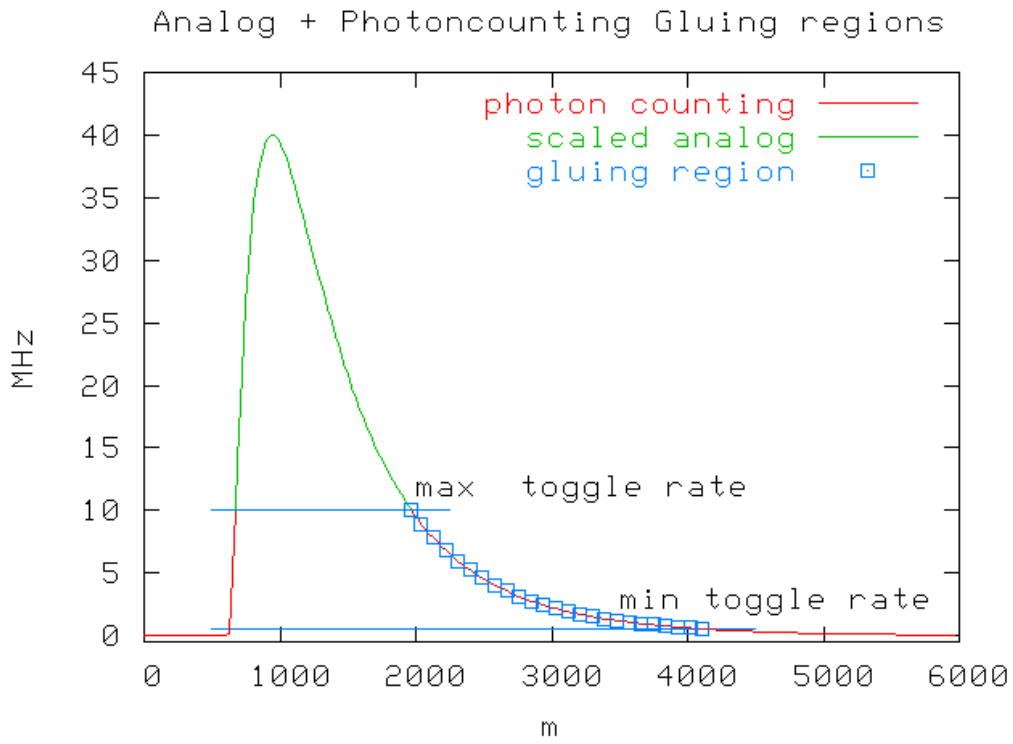
Figure 9.4: Increased dynamic range under low light level conditions

9.6.5 Gluing strategy

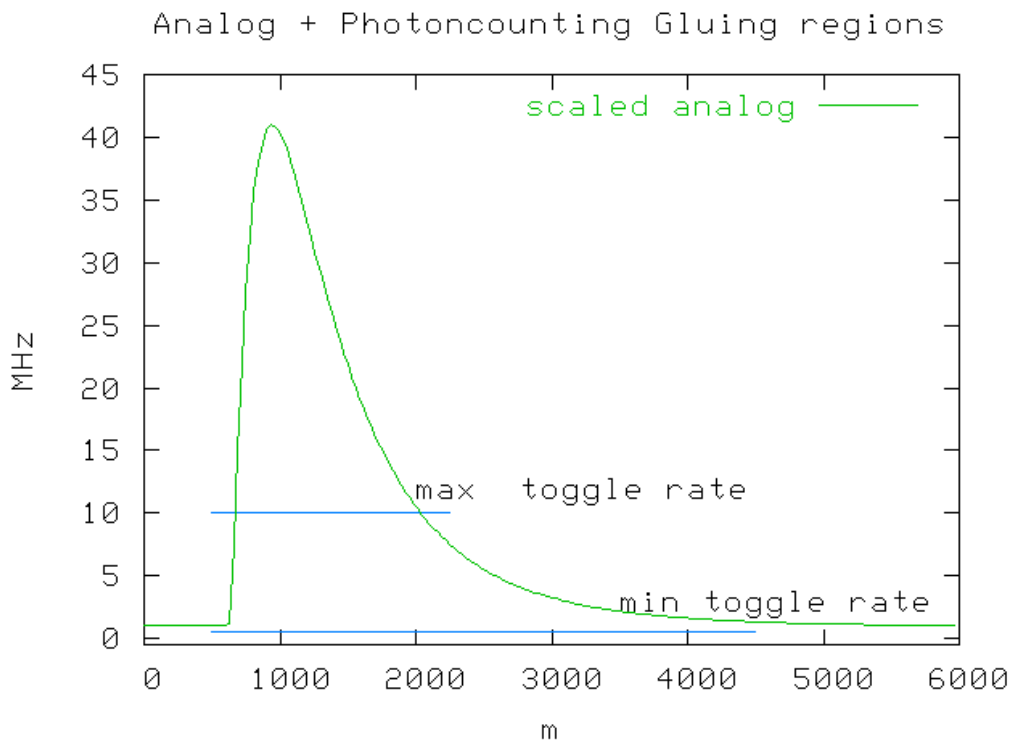
In principle one should glue two signals only if it is necessary. The only scenario when one really need to glue is when:

1. the peak value of the dead-time corrected photon counting is above the maximum toggle rate and
2. the background of the dead-time corrected photon counting is below the minimum toggle rate.

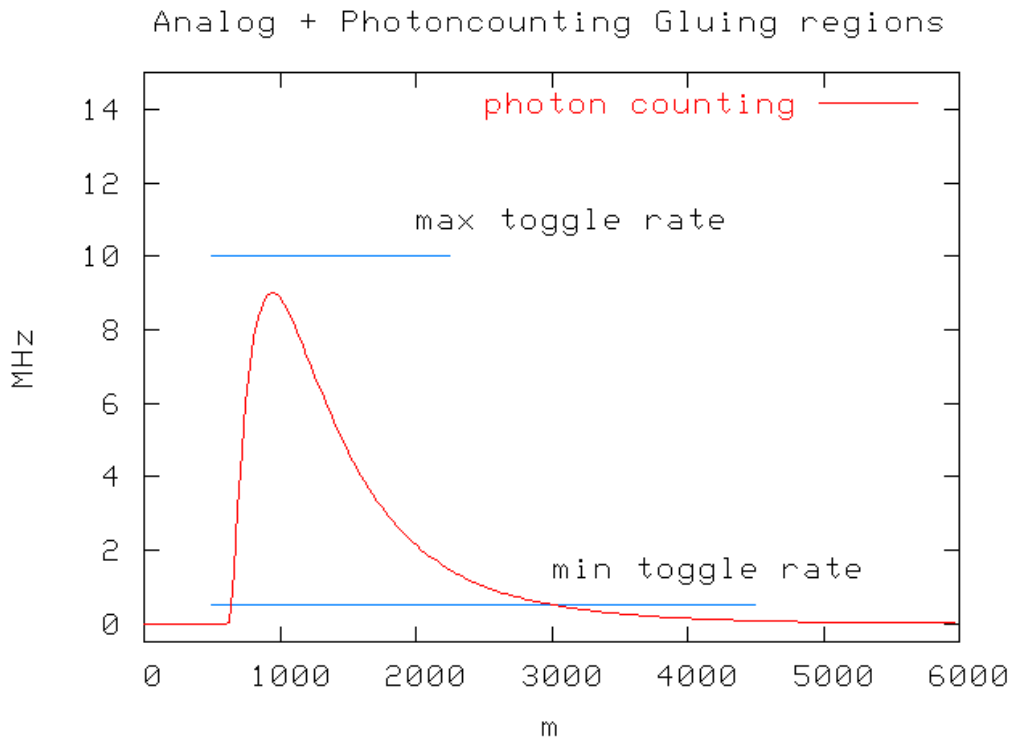
This situation is shown below:



If one assumes that the analog is valid enough to compute a regression curve then there is no need to compute a regression if the photon counting background exceeds the minimum toggle rate. In this case one can use the scaled analog.



If the peak count rate does not exceed the max. toggle rate there is no need to glue either and the dead-time corrected photon counting should be used.



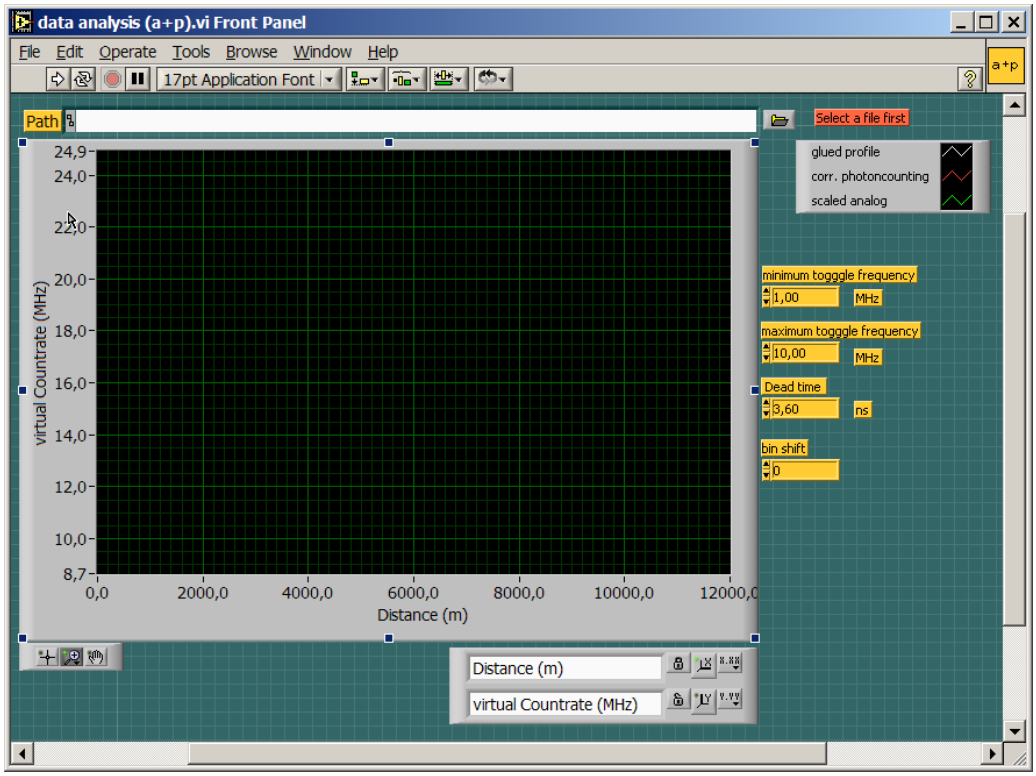
The use of a glued profile instead of a pure photon counting profile if the peak value is only slightly above the max. toggle rate. say at 12 MHz for 10MHz max. toggle rate could also be avoided.

9.6.6 Tutorial

Licel provides a sample code in LabVIEW for combining analog and photon counting data. The sample code assumes that the provided data has been previously recorded with the Acquis Software. One needs a LabVIEW license to look into the code. Reuse of this code in your projects is desired and permitted.

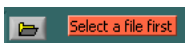
Loading the VI

Please open the `data analysis (a+p).vi` from the `Postan.llb` or the corresponding Windows application from the Windows start menu.



Selecting a data file

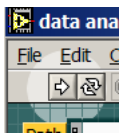
Click first the browse button



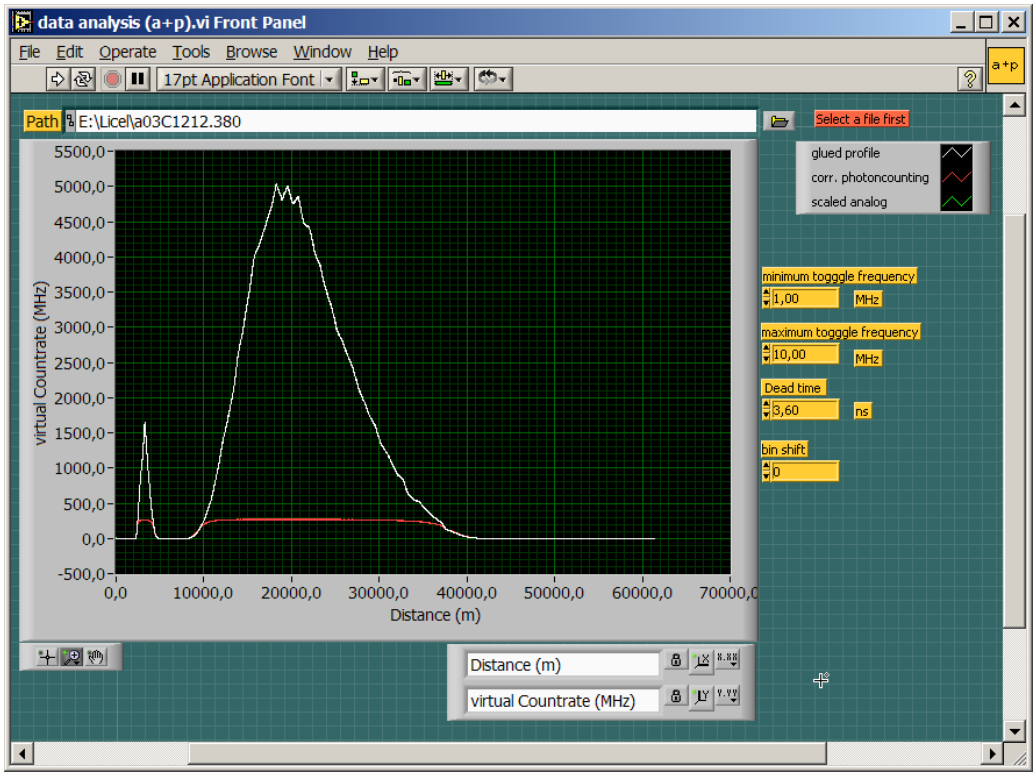
in the upper left part of the vi and select a data file that has previously been recorder with the Acquis-Module. At `http://www.licel.com/download/gluetestfile.zip` one can find the data file which has been used for this demonstration.

First Run

Press the run button in the upper right corner



and one should see the following curves



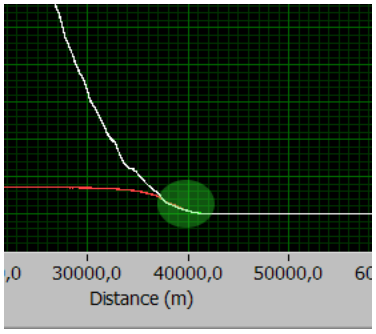
The white curve shows the combined signal.

Bin shift

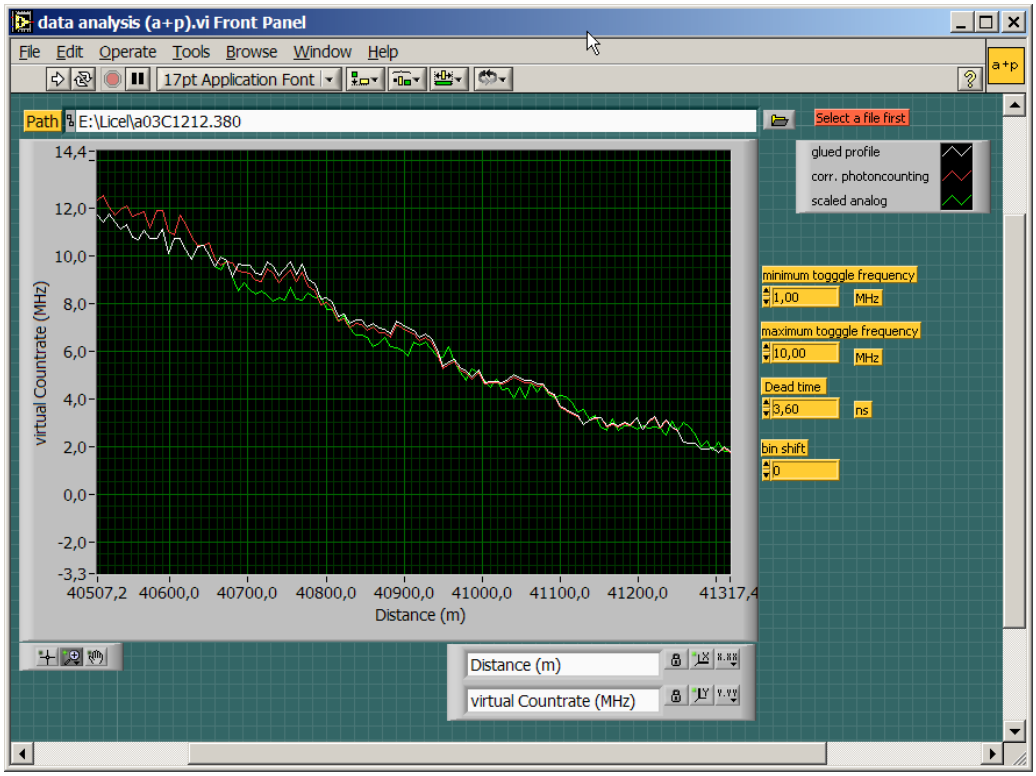
The analog and the photon counting data has a fixed shift between them. This is a result of two factors

1. Analog Bandwidth, the preamplifier contains an antialias filter which has a bandpass of half the sampling frequency this delays the analog signal with respect to the photon counting by 2 bins
2. ADC pipelining, modern ADCs sample the voltage in a multiple step process so that the sample result will be available several clock cycles later after the actual sampling took place.

To demonstrate this zoom into the profile



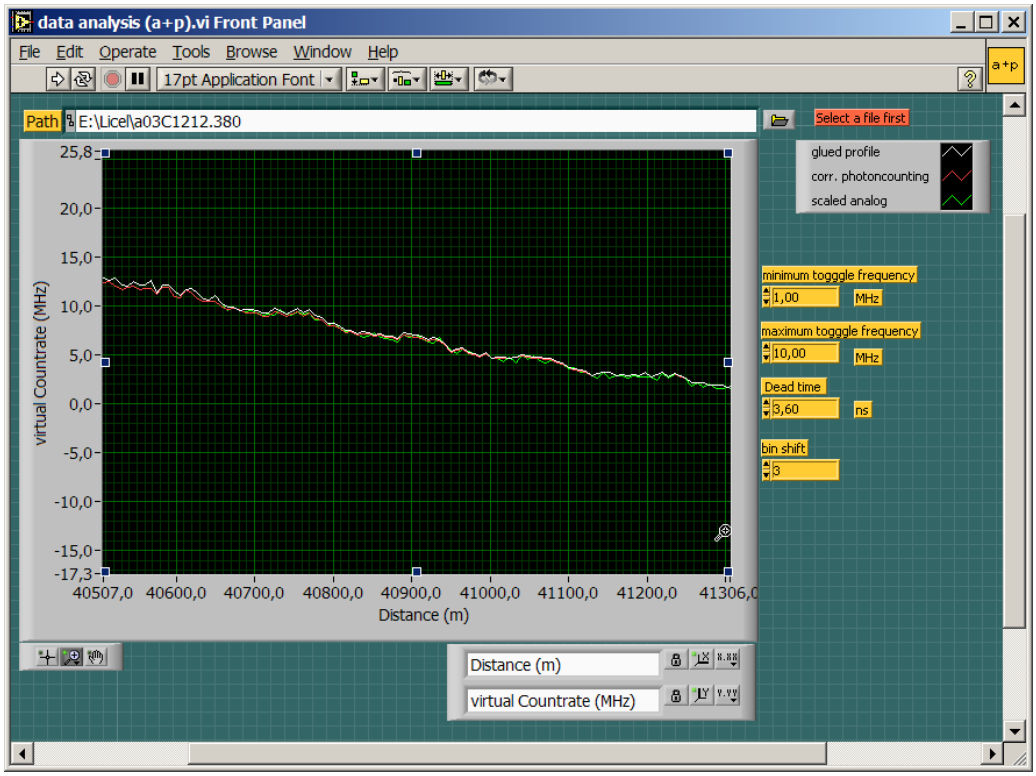
There is a shift of the scaled analog signal versus the photon counting data (the green vs. white curve)



Setting the bin shift to 3 will result in a much more perfect match.

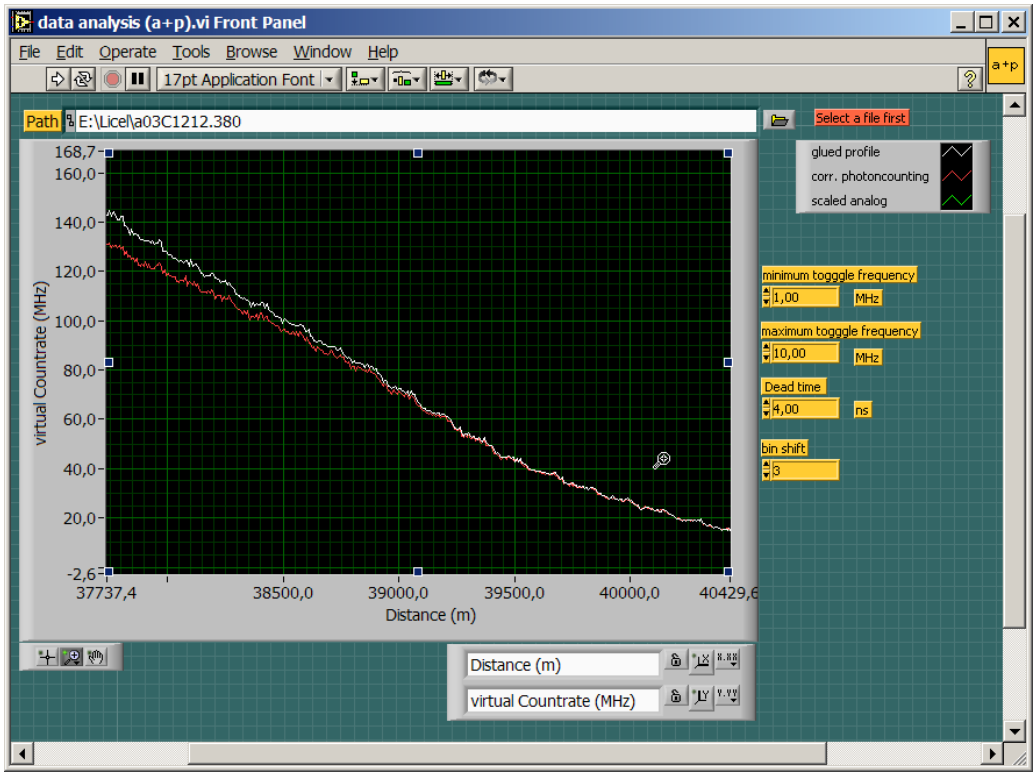
- For 12 bit TR shipped between before 2002 this would be 2-3.
- For 12 bit TR shipped between 2002 and 2018 this would be 9.
- For 16 bit TR shipped between 20010 and 2018 this would be 16.
- For TR shipped since this would be 0-1.

The binshift is a transient property which can be retrieved with the `TRTYPE?` command. Its stored in the data files see the `Data File format` appendix.



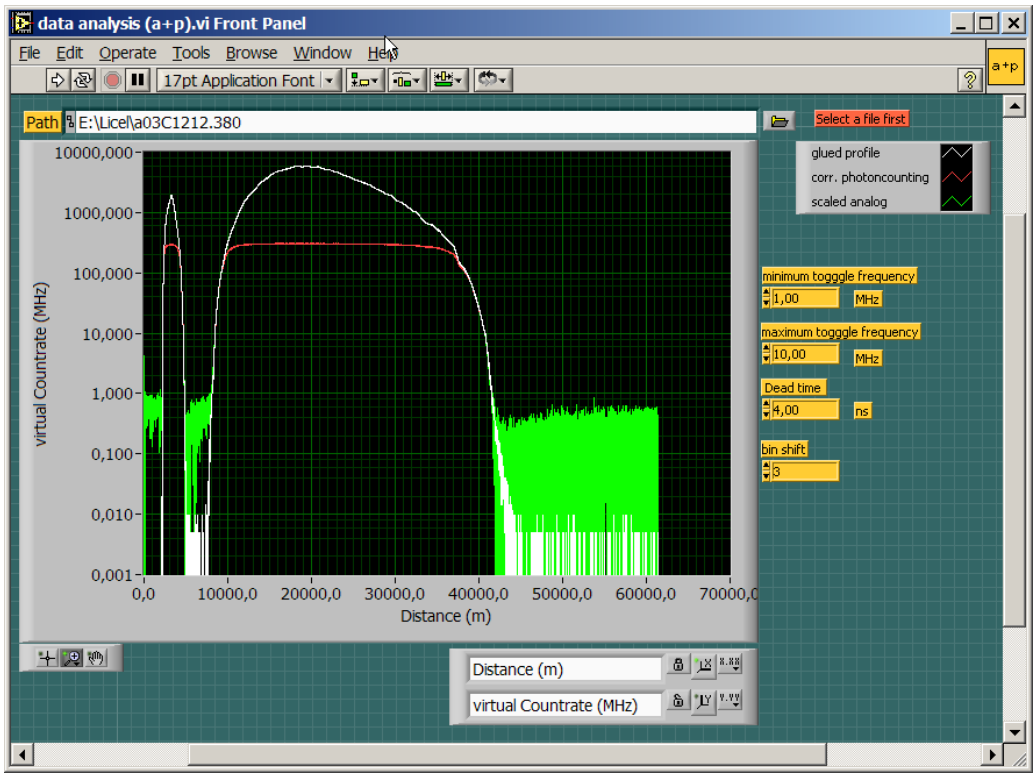
Photon counting dead-time correction

The default value of 280MHz is rather conservative approach for the dead-time correction. Lowering this value increases the dead-time correction. In the region above the max toggle rate a perfect dead-time correction will show a longer region where the glued curve and the dead-time corrected photon counting coincide.



5 Orders of magnitude

Changing the y-scale from linear to logarithmic reveals the potential of this signal combination.



The red curve shows that the photon counting becomes nonlinear and saturates. The green curve shows that signals which are close to the analog baseline are difficult to distinguish. But the combination of both signal prevents the nonlinearity for strong signals and gets the good baseline from the photon counting.

Next steps

Code similar to `data analysis (a+p).vi` needs to be integrated into the data retrieval software. Experience shows that recording background file without a laser signal and subtracting the averaged background from real signals will improve the analog background flatness and give more consistent gluing results. Once the transfer coefficients are found one could use them instead of searching in every signal for a new set of coefficients. The coefficient should stay constant if the detector has the same applied high voltage.

9.7 LabVIEW TCPIP Driver vi Tree

In this subsection an overview about the provided LabVIEW VIs is given.

9.7.1 Top Level VI's

Licel TCPIP VI Tree.vi

Go to the diagram (Ctrl-E) to view the VI Tree Hierarchy and to quickly open any of the VI's included.

Licel TCPIP Activate DHCP Mode.vi

This VI is used to activate DHCP for the transient recorder controller.
This VI uses the default password **Administrator** and the default port **2055**. If the port has been changed, you must change the **current port** to the proper value. The **DHCP port** is the port that will be used for DHCP communication. After DHCP mode has been set, communication will be lost until the acquisition computer is configured for DHCP communication as well.

Licel TCPIP Disable Secure Mode.vi

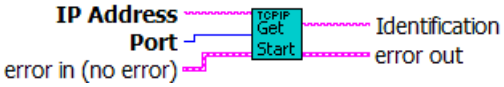
This VI is used to disable the Secure Mode of the Licel Ethernet Controller. The initialization file LicelTCPIP.ini is modified to allow future access without using the Secure Mode login.

Licel TCPIP Enable Secure Mode.vi

This VI is used to enable the Secure Mode of the Licel Ethernet Controller. The initialization file LicelTCPIP.ini is modified to allow future access using the Secure Mode login. This file should be copied to the same directory where Licel TCPIP.llb resides on all PCs from where access is allowed.

Licel TCPIP Getting Started.vi

This VI gets the identification information from the transient recorder controller.



Licel TCPIP Set Fixed IP Address.vi

This VI is used for setting the new IP configuration for the transient recorder controller.

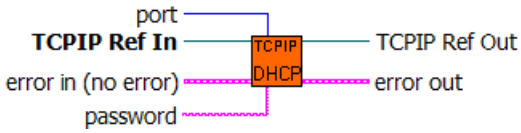
Licel TCPIP Set New Password.vi

This VI is used for setting the new password for the Licel Ethernet Controller.

9.7.2 Controller related VI's

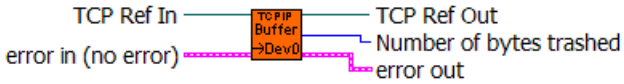
Licel TCPIP Activate DHCP.vi

This VI is used to activate the DHCP mode of the transient recorder controller.
In order to do so, the user must enter the proper password and port number for the controller. After DHCP mode has been set, communication will be lost until the acquisition computer is configured for DHCP communication as well.



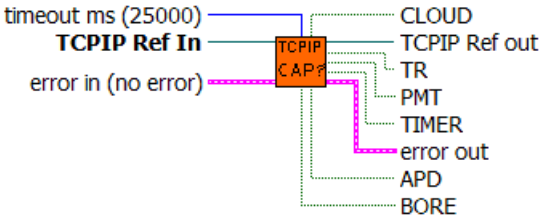
Licel TCPIP Dump TCPIP Buffer.vi

This VI empties the TCPIP buffer by reading all the data that is available in the buffer.
The **Number of bytes trashed** shows how many bytes were read from the buffer and disposed of.



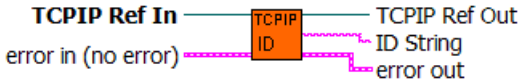
Licel TCPIP Get Capabilities.vi

The vi enables or disables the trigger mode for the Lamp, Pretrigger, Q-Switch, and Gating. The user can also switch between the internal and an external trigger using the External Trigger control.



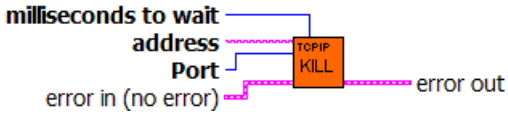
Licel TCPIP Get ID.vi

gets the identification string from the transient recorder controller.



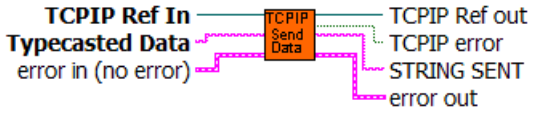
Licel TCPIP Kill Sockets.vi

This VI opens a new connection to the TR and sends the command to close down and reset all TCPIP connections. After doing this, the VI shuts down its TCPIP connection and waits the specified number of milliseconds, **milliseconds to wait**, before returning.



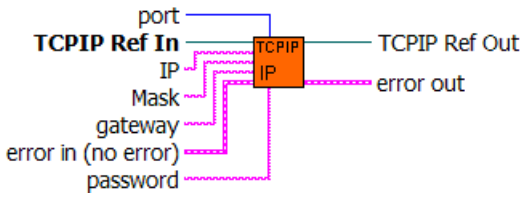
Licel TCPIP Send Data.vi

adds a `CRLF` to the end of the string and sends it via TCPIP using the TCPIP reference input



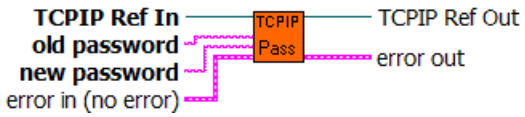
Licel TCPIP Set IP Parameter.vi

This VI is used to configure the transient recorder controller for static IP communication. With it, the values of the **IP** address, **port** number, subnet **mask**, and **gateway** can be set.



Licel TCPIP Set Password.vi

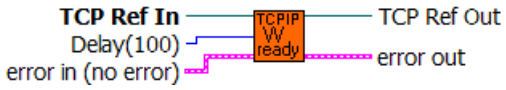
This VI is used for setting the password of the transient recorder controller.
This password must be given in order to change the IP configuration of the controller.



9.7.3 Transient recorder

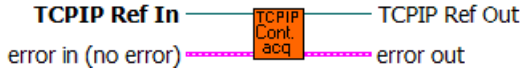
Licel TCPIP Wait For Ready.vi

Waits for return of the device from the armed state. If the waiting time is longer than the time specified by delay than the device remains armed and will be return to the idle state with next reading of binary data



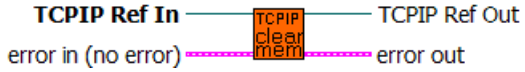
Licel TCPIP Continue Acquisition.vi

Continues the recording process for the specified device without reinitializing the memory.



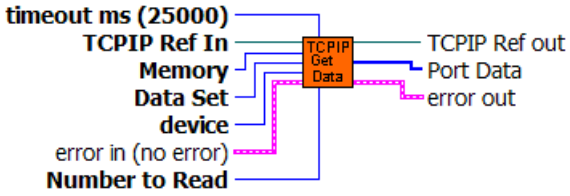
Licel TCPIP Clear Memory.vi

Clears all memories of the specified device.



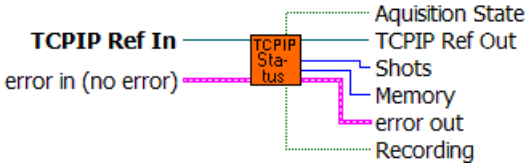
Licel TCPIP Get Datasets.vi

is a vi for reading raw data sets (analog LSW, analog MSW or photon counting and upper word photon counting) from the specified device.



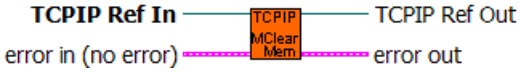
Licel TCPIP Get Status.vi

Returns the status information for the specified device (cycles,memory,acquisition state and whether the device is just recording).
If an error parsing the status information occurs, the VI returns an error 5765.



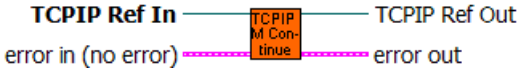
Licel TCPIP Multiple Clear Memory.vi

Clears all memories of the currently selected devices.



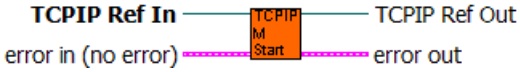
Licel TCPIP Multiple Continue Acqusition.vi

The acquisition process of the selected multiple devices will be restarted without clearing their memories.



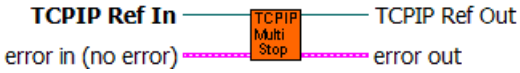
Licel TCPIP Multiple Start.vi

The acquisition process will be started after the next received trigger for multiple devices



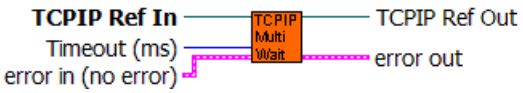
Licel TCPIP Multiple Stop Acqusition.vi

The acquisition process will be stoped after the next received trigger for multiple devices



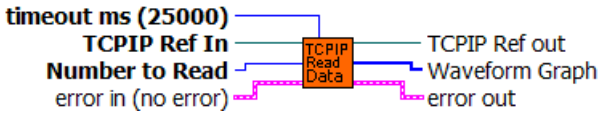
Licel TCPIP Multiple Wait For Ready.vi

The vi waits until all devices returned from the armed state.



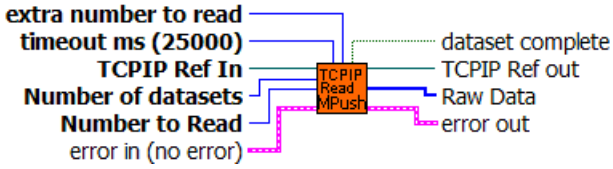
Licel TCPIP Read Data.vi

This VI waits until the number of scans defined by **Number to Read** is available and reads them or returns a timeout error if the **timeout ms** is exceeded.



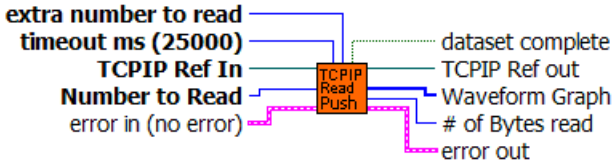
Licel TCPIP Read MPushed Data.vi

This VI reads the pushed data from multiple transient recorders at once. The data from the various transient recorders is concatenated together and must still be separated.



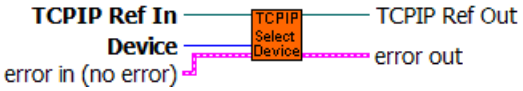
Licel TCPIP Read Pushed Data.vi

This VI is used for reading a single pushed data set.



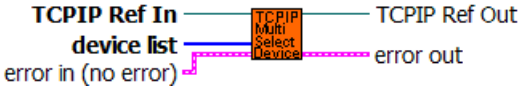
Licel TCPIP Select Device.vi

selects the device specified by the input **device number**. Selecting a device makes it active for all future commands that do not have a required **device number** input. The previously selected devices become deselected when this command is issued.



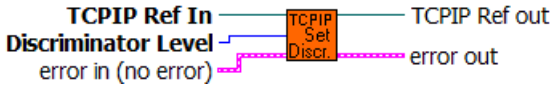
Licel TCPIP Select Multiple Devices.vi

This VI is used to select multiple transient recorders.
The devices corresponding to the numbers in the **device list** array will be selected which means that they will become sensitive to all future commands that do not require a **device number** input. The devices will be deselected if another **select** command is issued.



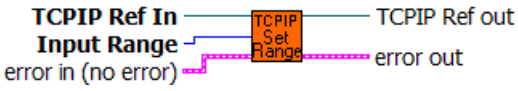
Licel TCPIP Set Discriminator Level.vi

Set the discriminator level between 0 and 63 for the selected transient recorders.



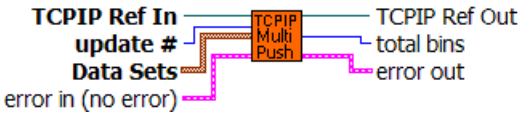
Licel TCPIP Set Input Range.vi

The vi changes the input voltage range.



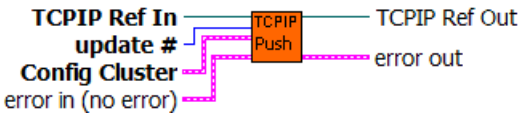
Licel TCPIP Set Multiple Push Mode.vi

This VI is used to start the push mode for one or more devices.
This VI takes the **Data Sets** information and the **update #**, which is the number of laser pulses to acquire, as input parameters. Based upon these inputs, the VI generates and sends a command to start the push mode for the transient recorders specified by **Data Sets**.



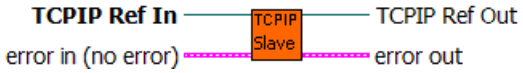
Licel TCPIP Set Push Mode.vi

sets the push mode for the currently selected transient recorder.



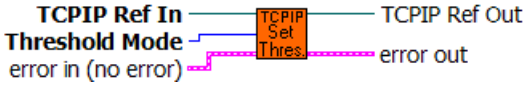
Licel TCPIP Set Slave Mode.vi

This VI stops the push mode and sets the transient recorder controller back in to the slave mode.



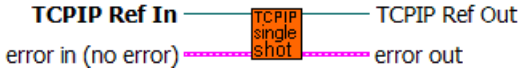
Licel TCPIP Set Threshold Mode.vi

Set Threshold Mode sets the scale of the discriminator level. In the low threshold mode the discriminator level 63 corresponds to -25mV while in the high threshold mode it corresponds to -100mV.



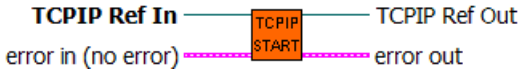
Licel TCPIP Single Shot.vi

Acquires one shot with the currently selected device.



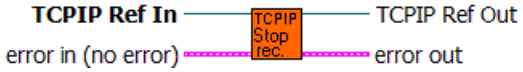
Licel TCPIP Start.vi

starts the currently selected transient recorder.



Licel TCPIP Stop Acqusition.vi

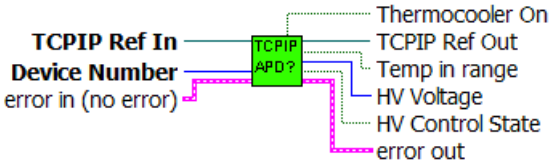
This VI stops the acquisition process after the next received trigger.



9.7.4 APD

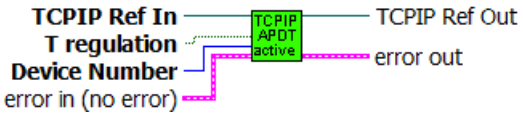
Licel TCPIP APD Get Status.vi

This VI gets the status of the APD with the corresponding device number.
The values that are returned are the
HV Voltage : this is the actual gain voltage
On : this boolean is true if the gain voltage power supply is on,
otherwise it is false **control state** : if true, the APD is being controlled remotely,
if false, then the APD is being controlled locally
T regulation: if true, then the cooling has been activated
if false, then the cooling is inactive; i.e. passive



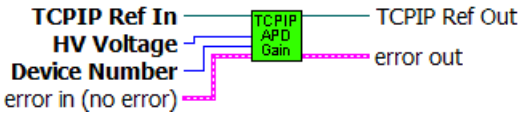
Licel TCPIP APD Set Cooling State.vi

This VI sets the cooling state for the APD with the corresponding device number.
True = the current to the peltier cooling will be activated
False = the cooling will not be activated. Only passive cooling occurs.



Licel TCPIP APD Set Gain.vi

Sets the Gain Voltage for the APD specified by the **Device Number** to the value specified by **HV Voltage**.



9.7.5 PMT

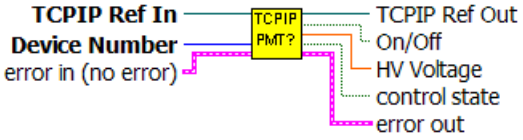
Licel TCPIP PMT Get Status.vi

This VI gets the status of the PMT with the corresponding device number.
The values that are returned are the

HV Voltage : this is the actual gain voltage

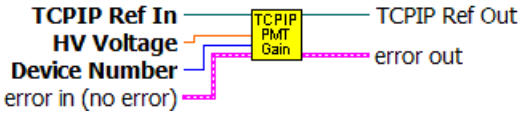
On : this boolean is true if the gain voltage power supply is on, otherwise it is false

control state : if true, the PMT is being controlled remotely, if false, then the PMT is being controlled locally



Licel TCPIP PMT Set Gain.vi

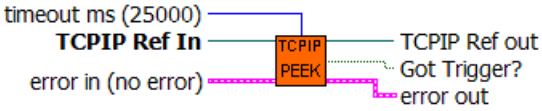
Sets the Gain Voltage for the PMT specified by the Device Number to the value specified by HV Voltage



9.7.6 Trigger

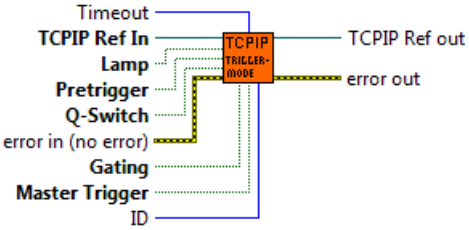
Licel TCPIP Peek Trigger.vi

The vi enables or disables the trigger mode for the Lamp, Pretrigger, Q-Switch, and Gating. The user can also switch between the internal and an external trigger using the External Trigger control.



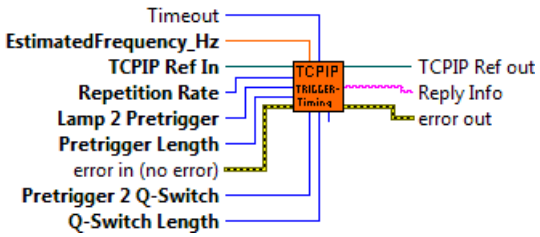
Licel TCPIP Set Trigger Mode.vi

The vi enables or disables the trigger mode for the Lamp, Pretrigger, Q-Switch, and Gating. The user can also switch between the internal and an external trigger using the External Trigger control.



Licel TCPIP Set Trigger Timing.vi

The vi allows the user to set the times in ns for the Lamp, pretrigger delay, pretrigger length, Q-Switch delay and Q-switch length



9.7.7 Power Meter

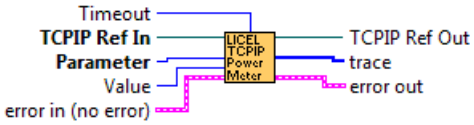
Licel TCPIP POW Parse Trace.vi

Parses the string reply of the **POW** TRACE command and returnes the data as an array.

Licel TCPIP PowerMeter.vi

The vi sends the POW command with the **Parameter** (0: START, 1: STOP, 2: RESET) to the controller.

START causes the controller to send power meter data wheneverr receiving a trigger
STOP stops transferring data.
CHANNEL sets the ADC channel
TRACE starts a single pulse acquisition and returns the last trace of data points

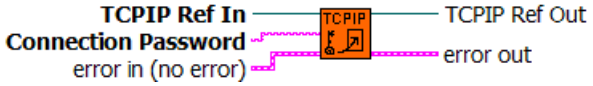


An error is generated if the vi does not receive the appropriate reply (`POW <Parameter>executed` or for TRACE a sequence of decimal string numbers).

9.7.8 Network Security

Licel TCPIP Login Secure Mode.vi

Send the LOGON command to work in secure mode. Reads a string from TCPIP, attempts to convert the string to 2 U32 numbers used to encrypt the password to 2 output U32 numbers using the Blowfish encryption algorithm. These output numbers are converted to a hexadecimal string to be used in the LOGON command. If the LOGON command fails the controller will close the connection without any notification.

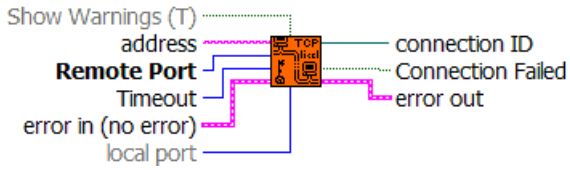


Licel TCPIP Open Secure Mode.vi

Open a TCP/IP connection to the Licel controller in secure mode. The vi tries to open the initialization file LicelTCPIP.ini to read the values for the keys UseSecureMode and SecureModePWD from the SecureMode section:

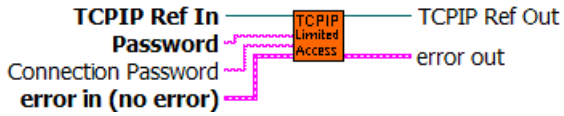
```
[SecureMode]
UseSecureMode=TRUE
SecureModePWD=ConnectMe
```

If the initialization file is found and UseSecureMode is true and SecureModePWD is found the vi will send the password using the LOGON command (Licel TCPIP Login Secure Mode.vi). Otherwise just the TCP/IP connection will be opened.



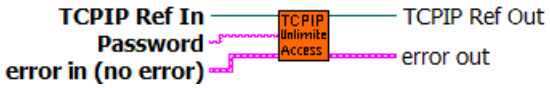
Licel TCPIP Set Access Limited.vi

Enables the limited access to the controller, i.e. activates the secure mode. Access is granted only for IP addresses as specified with the WHITELIST command. Moreover the connection password is specified.



Licel TCPIP Set Access Unlimited.vi

Disables the limited access to the controller, i.e. deactivates the secure mode. Access is granted for everybody.



Licel TCPIP Set Whitelist.vi

This VI is used to set the allowed hosts at the controller. In order to do so, the user must enter the appropriate password and 3 host strings to allowed IP addresses or IP address ranges. Such a string must be specified in the following format:

xx.xx.xx.xx	a single IP address,
xx.xx.xx.255	an IP address range (0:255),
	or may be empty.