

Linear Sensor
Graphical User Interface
User Manual

SGF Ltd.
Pipiske u. 1-5/20
1121 Budapest
Hungary

Contents

- 1. Basic architecture..... 3
- 2. LinearSensorGUI program 4
 - 2.1 Platform 4
 - 2.2 Installation 4
 - 2.3 The config.ini file..... 5
 - 2.4 User interface 7
 - 2.4.1 Menu bar 8
 - 2.4.2 Normal Mode tab panel 10
 - 2.4.3 Calibration Mode tab panel..... 15
 - 2.4.4 Sarcomere Length playback tab panel..... 19
 - 2.5 Definition of IP address..... 20

1. Basic architecture

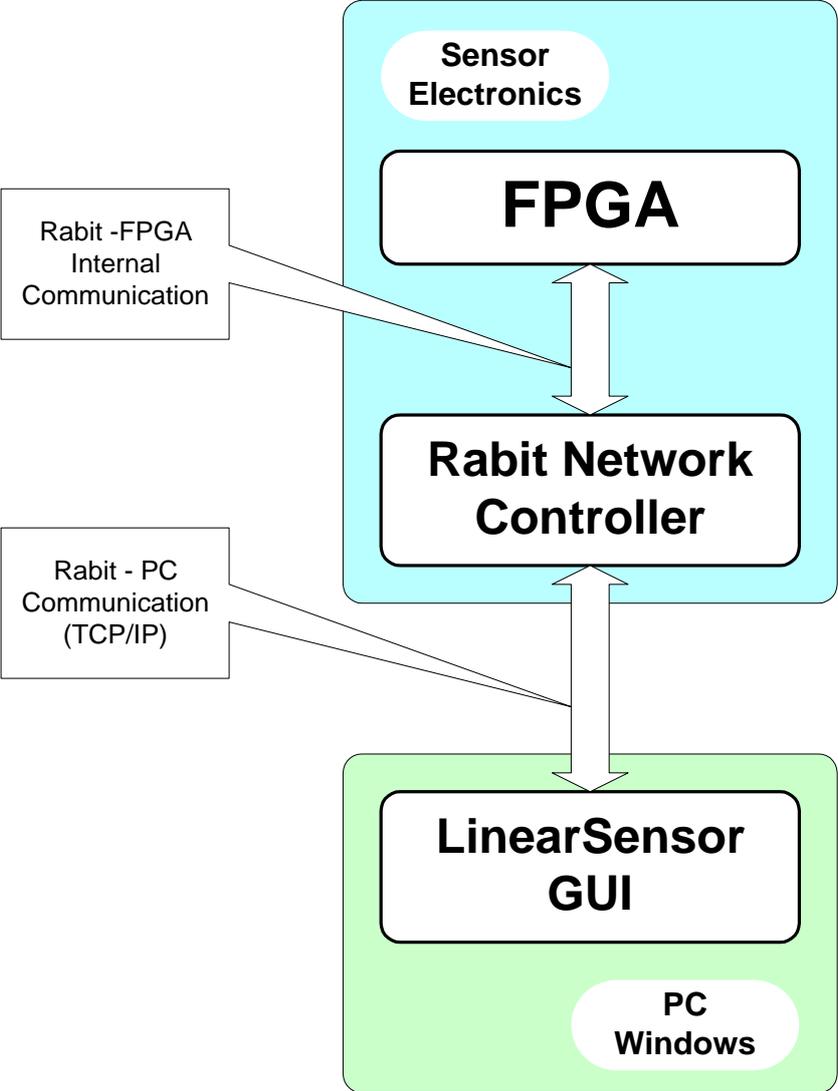


Figure 1. The system architecture

Figure 1 shows the architecture of the LinearSensor monitor and control system. It consists of two parts: one is the embedded measurement computer implemented on an FPGA chip and containing a network adapter card (Rabbit controller). It is referenced further on as the sensor. The other part is the monitoring and controlling computer, running the LinearSensorGUI software referenced further on as LSGUI. This computer is connected to the sensor via Ethernet with TCP/IP communication protocol.

2. LinearSensorGUI program

2.1 Platform

The LinearSensorGUI software has been developed for Windows 2000/XP platforms.

2.2 Installation

The Linear Sensor Graphical User Interface software consists of six files (setup.exe, setup.ini, LinearSensor.msi, instmsiv.exe, instsi.exe and DistFile.cab). The installation process is the same as with other simple Windows based applications. The setup.exe program should be run and only the installation destination folder must be specified, the default directory is "c:\Program Files\LinearSensor". After the installation has successfully been finished, the destination directory contains two important files. One is the application file (LinearSensor.exe) and the other is the configuration file (config.ini). The configuration file contains parameters in ASCII format and will be discussed later in this section. There will be an icon on the desktop:



NOTE: In the case of a secondary installation, the setup.exe program should be started twice. In the first run, it uninstalls the old LinearSensor program and its icon while the second run installs the new program and its icon!

2.3 The config.ini file

The configuration file contains the parameters in ASCII format for easy edition and readability.

- **[name_of_parameter_block]** marks the start of a parameter block.
- **name_of_parameter = value_of_parameter** Each parameter block contains parameters and corresponding values in this form. Every parameter definition must be followed by a New Line character.

The config.ini file contains the following parameter blocks and parameters:

- Port
 - IP IP address of the embedded computer
 - Port Port number
- Directory
 - ImageData default destination for data archivation
- ImagePeriode
 - T period time for image refreshing on the user interface
- Log
 - Size maximum size of the data archive file
- CalTable definition of the calibration table (10 rows, 4 values in each)
 - ROIP n low value of ROI in the n th row of the table
 - ROIW n width of ROI in the n th row of the table
 - PEAK n peak value in the n th row of the table
 - SLEN n sarcomere length value in the n th row of the table
- Interpolation
 - Mode determines the interpolation algorithm
 - 0 Spline interpolation
 - 1 ExtSpline interpolation (combination of linear and spline interpolations)
 - 2 Exponential interpolation

The configuration file with sample parameters:

Note that the *IP* and *ImageData* parameters are of character type so the “ is needed.

```
[Port]
IP = "192.168.1.98"
Port = 5194
[Directory]
ImageData = ""
[ImagePeriod]
T = 200
[Log]
Size = 10000
[CalTable]
ROI1 = 2
ROIW1 = 100
PEAK1 = 1770
SLEN1 = 1.54
ROI2 = 900
ROIW2 = 200
PEAK2 = 1025
SLEN2 = 1.27
ROI3 = 1200
ROIW3 = 200
PEAK3 = 1202
SLEN3 = 1.1
ROI4 = 1500
ROIW4 = 200
PEAK4 = 1502
SLEN4 = 1
ROI5 = 0
ROIW5 = 0
PEAK5 = 0
SLEN5 = 0
ROI6 = 0
ROIW6 = 0
PEAK6 = 0
SLEN6 = 0
ROI7 = 0
ROIW7 = 0
PEAK7 = 0
SLEN7 = 0
ROI8 = 0
ROIW8 = 0
PEAK8 = 0
SLEN8 = 0
ROI9 = 0
ROIW9 = 0
PEAK9 = 0
SLEN9 = 0
ROI10 = 0
ROIW10 = 0
PEAK10 = 0
SLEN10 = 0

[Interpolation]
Mode = 1
```

2.4 User interface

After the installation, the LinearSensor program can be started by double clicking on its icon. The following panel appears on the screen:

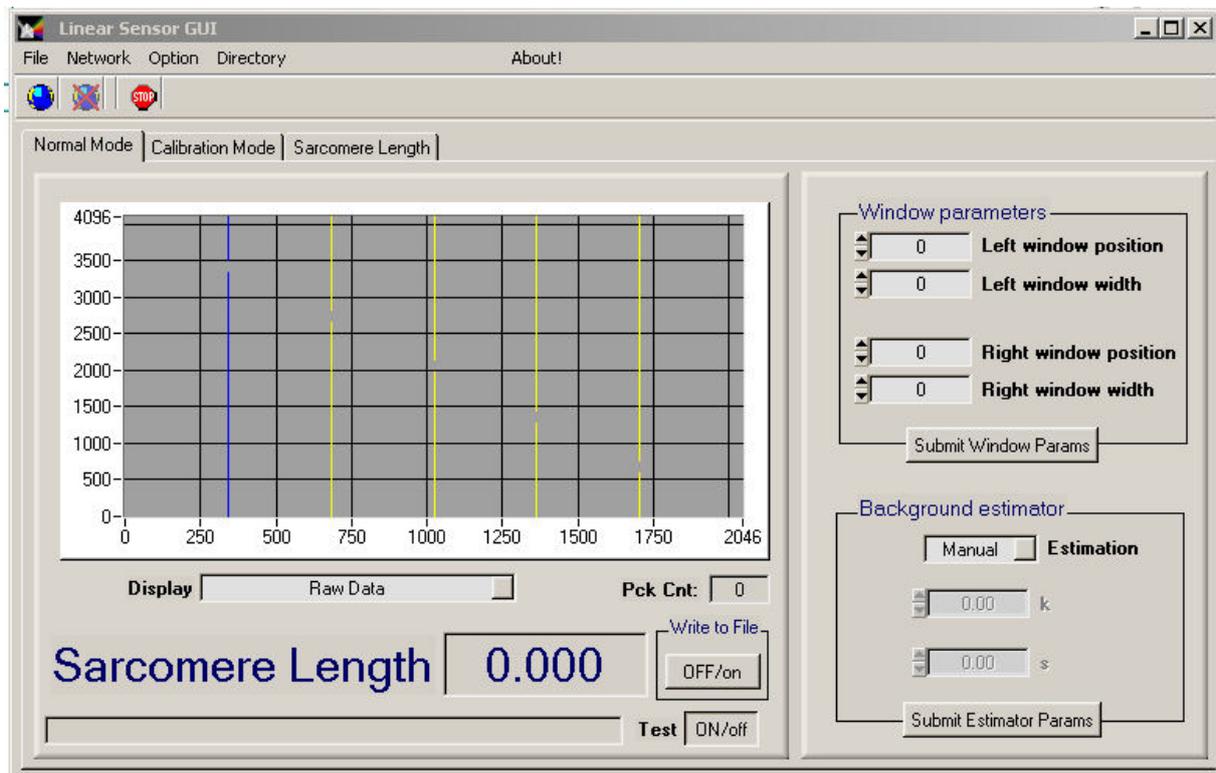


Figure 2. The user interface

Figure 2. shows the user interface of the LinearSensorGUI software. The menu bar contains 5 menus and 3 image buttons. The main area of the window contains 3 tab panels (*Normal Mode*, *Calibration Mode*, *Sarcomere Length*).

2.4.1 Menu bar



Figure 3. Menu bar

Figure 3. shows the menu bar. In the first line there is the name of program and on the right side there are usual Windows panel size handling buttons. In the lower line there are three buttons:



--- Establishes TCP/IP connection between GUI and the sensor (the button is active only if the connection is not established)



--- Closes TCP/IP connection between GUI and the sensor (the button is active only if the connection is established)



--- Exits from GUI

File menu



Select Data File

Archived sarcomere length data can be loaded from file. If the selected file is valid and contains sarcomere length data, the “Sarcomere Length” tab panel becomes active automatically and the software switches to data playback mode.

Exit

Exits from the LinearSensor software.

Network menu



Register to Sensor

Establishes TCP/IP connection between GUI and the sensor (active only if the connection is not established).

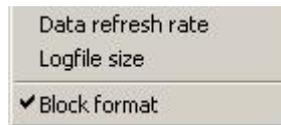
Disconnect from Sensor

Closes TCP/IP connection between GUI and the sensor (active only if the connection is established).

Definition of TCP address

The IP address and port of the server can be edited on a popup IP edit panel. The values specified on the IP edit panel are saved in the config.ini file. For further details see Section 2.5.

Option menu



Data refresh rate

The refresh rate of the image can be set to a minimum value of 100ms. If less than 100 ms is specified the refresh rate will be 100ms automatically. The specified value is saved in the config.ini file.

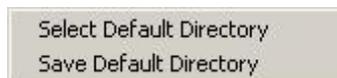
Logfile size

Determines the maximum size of the sarcomere length data archive file. If an opened file size reaches this value while data archiving, the file will be closed and a new file will be opened. The specified value is saved in the config.ini file.

Block Format

The format of the archive file can be specified. If this option is checked then the archive file will contain 10 sarcomere length values in one row. Otherwise it contains 1 sarcomere length value per row.

Directory menu



Select Default Directory

The archive directory can be selected by this menu. The archive files will be placed to the specified folder.

Save Default Directory

Saves the default folder in the config.ini file.

About menu

Identifies the LinearSensorGUI program version as usual in Windows environment.

2.4.2 Normal Mode tab panel

In normal mode, the LinearSensorGUI software sends periodically a data request packet to the FPGA. The FPGA answers with both an image packet and a sarcomere length packet. The sarcomere length packet contains all the measured sarcomere length values since the last sarcomere length packet sent. The sensor measures with higher rate than the GUIs image refresh rate. The measured but untransmitted images are lost but the sarcomere length values are collected in the FPGA and sent lossless in burst mode to the GUI. If the data archive mode is switched on, the GUI saves all the measured sarcomere length values to archive files.

The Normal Mode tab panel is used to visualize data received from the sensor and to set the internal state of the FPGA. The state can be set by changing values in a few fields. The field indicates the state of values with different colours:

 Left window position --- Green background in an entry field means data has been read or loaded to the sensor. The field surely contains the same value as the corresponding state variable in the embedded computer.

 Left window position --- Yellow background in an entry field means data has been modified but not uploaded to the sensor. The value in this field may be different from that in the FPGA.

 Left window position --- Grey background in entry field means the sensor has not been connected to the LinearSensorGUI.

 Left window position --- Red border around an entry field means data is invalid. It is either lower or higher than the possible range of the value.

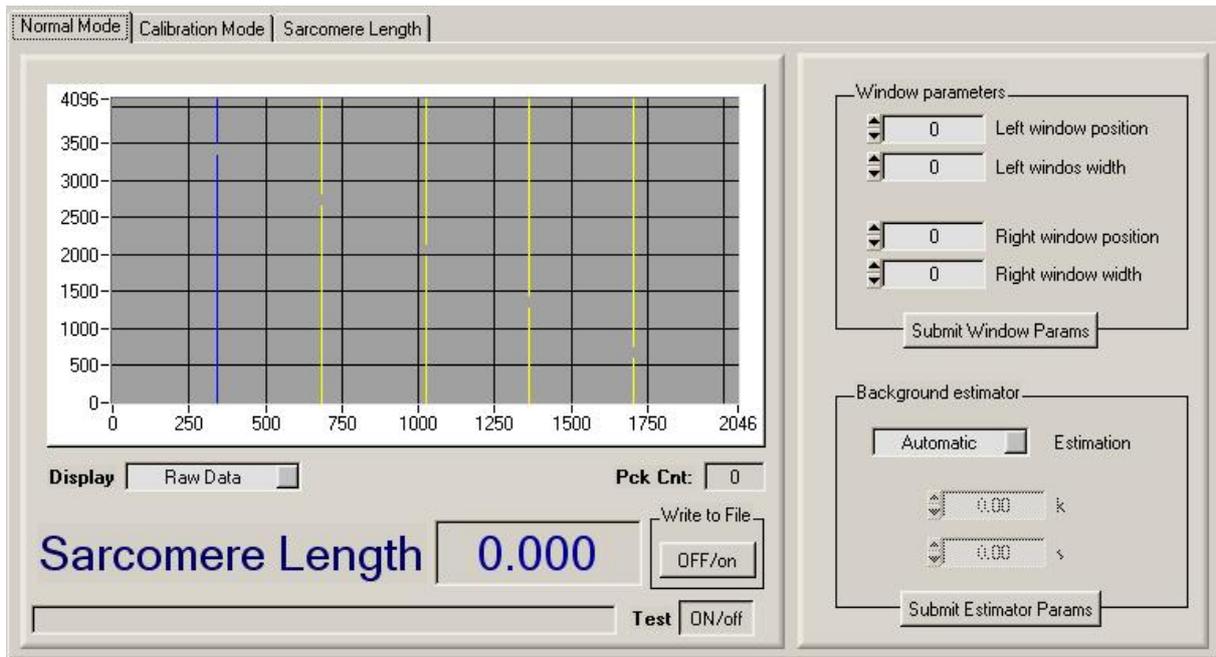


Figure 4. Normal mode tab panel

Figure 4. shows the Normal Mode tab panel with the following sections:

Image data graph

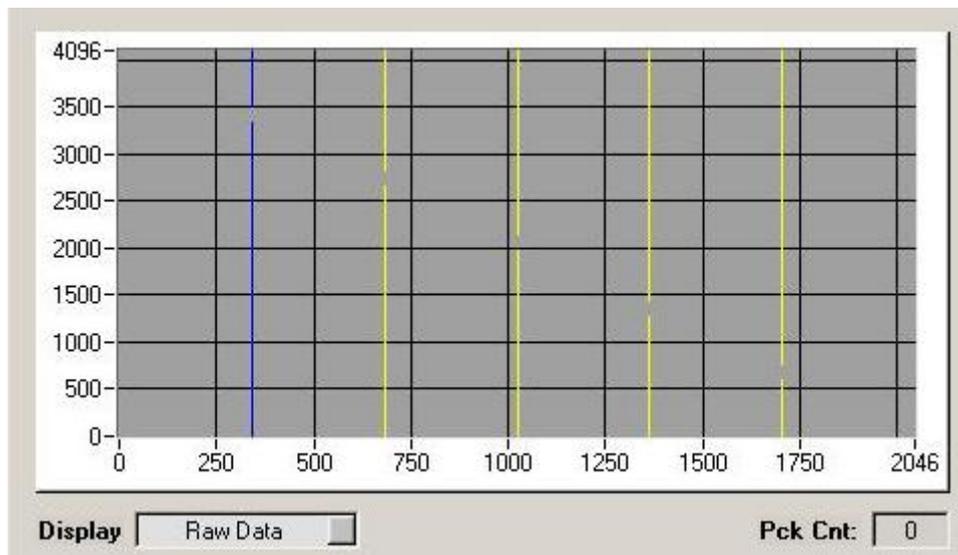


Figure 5. The image data graph

The sensor sends the image data in one packet. An image data packet contains 2047 12-bit pixel values and one 12-bit peak value. The image data graph shows the graphical representation of the received image. If an image data arrives from the sensor, a red curve on the graph will visualize the image data from the packet. The graph contains 1 blue and 4 yellow markers. The blue marker shows

the pixel position of the peak value in the last received image data packet (the peak value is calculated and sent by the FPGA). The 4 yellow markers show the positions of the left and right windows.

Display mode

<i>RAW Data</i>	The FPGA sends non-background subtracted image data so the graph shows non-background subtracted image.
<i>BG Subtracted</i>	The FPGA sends background subtracted image data so the curve on the graph shows background subtracted image.
<i>Calculated BG</i>	The FPGA sends non-background subtracted image data. The LinearSensorGUI software calculates the background of the measured RAW image from the received values, the background estimator and the window settings.

If the current state of the background estimator on the GUI is automatic then the background is calculated using the following equations and parameters:

$$\bar{y}_L = \frac{\sum_{i=x_{LL}}^{x_{LH}} y(x_i)}{x_{LH} - x_{LL}} \quad \bar{y}_R = \frac{\sum_{i=x_{RL}}^{x_{RH}} y(x_i)}{x_{RH} - x_{RL}}$$

$$x_L = \frac{x_{LL} + x_{LH}}{2} \quad x_R = \frac{x_{RL} + x_{RH}}{2}$$

$$k = e^{(\ln \bar{y}_L + s * x_L)} \quad s = \frac{\ln \bar{y}_L + \ln \bar{y}_R}{x_R - x_L}$$

x_{LL} : left window low

x_{LH} : left window high

x_{RL} : right window low

x_{RH} : right window high

If the current state of the background estimator on the GUI is manual then the given K and S parameters are used for the calculation.

The background calculation if the K and S parameters are preset either on the GUI or from the above calculation, the background is:

$$f(x) = k * e^{-s * x}$$

Every change of the display mode forces the GUI to send the complete status set on the normal mode panel. Thus a change in the display mode will submit all other possibly uncommitted values set on the panel.

Sarcomere length



Figure 6. The sarcomere length field

The sarcomere length values are collected in the embedded computer and sent to the LinearSensorGUI in burst mode. The sarcomere length field shows the last value from the last received burst. If the Write to File switch button is in OFF mode then the other sarcomere length values are unused. If the button is in ON mode than the received sarcomere length values are placed to an archive file in the specified default directory.

Window parameters

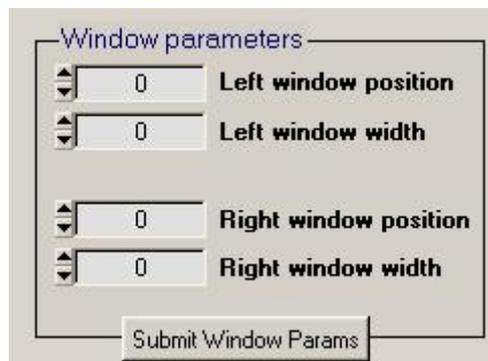


Figure 7. The window parameter box

The Window parameters box contains the four windows setting parameters. There are two windows (left/ right) each described by 2 parameters: a position parameter (e.g. the pixel value of the lower border of the window) and a width parameter. The values edited on this box are only committed and become actual in the embedded computer by submitting them with the Submit Window Params button. Note that parameter submitting forces the GUI software to send all the status parameters set on the GUI and not only these 4 values!

Background estimator box

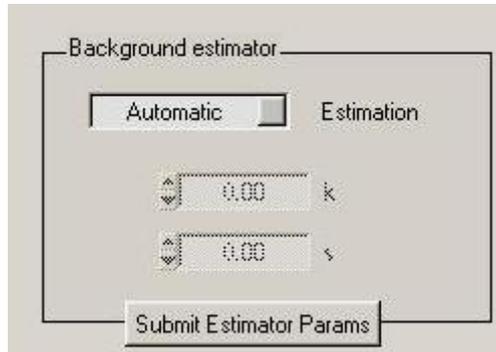


Figure 8. The Background estimator box

On the background estimator box, the method of background estimation can be set. In Automatic mode, the FPGA automatically calculates the K and S parameters for the estimation using the window parameters. If the estimator is in Manual mode, then the FPGA will calculate the background using the K and S parameters given on the GUI. The values edited on this box are only committed and become actual in the embedded computer by submitting them with the Submit Estimator Params button. Note that parameter submitting forces the GUI software to send all the status parameters set on the GUI and not only the parameters from this box!

2.4.3 Calibration Mode tab panel

In calibration mode, the user has to fill a calibration table containing 10 rows. The correct sequence of filling a calibration row is the following:

1. Defining the left border of the Range of Interest (ROI P)
2. Defining the width of the Range of Interest (ROI W)
3. After defining the ROI W value, the LSGUI software will automatically send a command to the embedded computer to activate the current settings in the sensor. The sensor automatically answers with an image packet. From this image packet (that has been measured with the given parameters), the peak value inside the ROI will automatically be presented by the LSGUI software. The “wait” (sand-glass) state of the mouse cursor indicates that the GUI software is waiting for the image data. (The user has to wait a few seconds.)
4. After receiving the image packet, the Peak field will automatically be filled.
5. The user has to set the correct sarcomere length value for the arrived peak position.

Filling of the rows is optional but it is strongly recommended to fill minimum 4 lines. After the desired amount of lines has been correctly filled, the LSGUI software will count the values of the lookup table using the selected interpolation algorithm and the presented sarcomere length – peak position value pairs as fix points. The software draws the calculated curve to the “Sarcomere Length” graph (visualized lookup table). If the user accepts the lookup table, it can be sent to the sensor by clicking the Submit button.

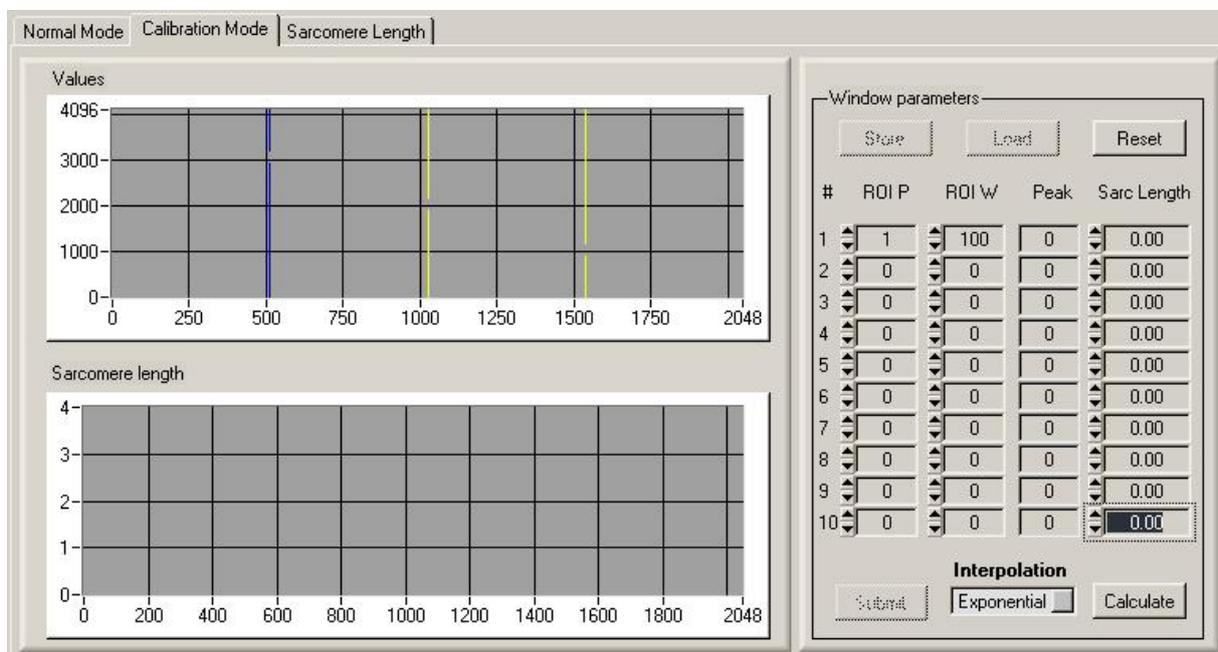


Figure 9. The calibration tab panel

The **left side** of the calibration tab panel contains two graphs. The Values graph shows the last received image data, the actual ROI (yellow markers) and the actual peak position (blue marker). The Sarcomere length graph shows the fix points and after using the Calculate button, the counted lookup table is also showed.

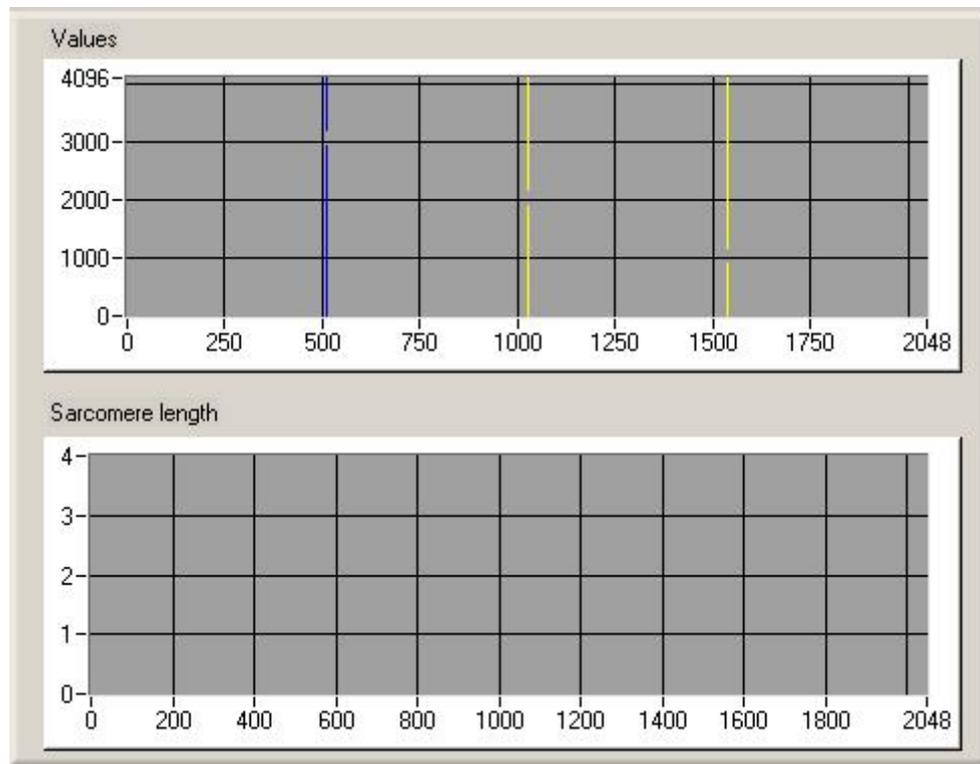


Figure 10. The graphs of the calibration panel

The **right side** of the calibration panel contains the calibration table. The calibration values can be set on the fields of this table. The calibration table can be saved in XML format and a saved table can be loaded.

Calibration table XML format

```
<?xml version="1.0" encoding="windows-1250" ?>
<!-- date:2007-01-10 -->
<CalibrationTable>
  <CalibrationPoint RoiPos="10" RoiWidth="100" Peek="23"
  SarcomereLength="2.33"/>
  <CalibrationPoint RoiPos="200" RoiWidth="100" Peek="135"
  SarcomereLength="1.9"/>
  <CalibrationPoint RoiPos="300" RoiWidth="100" Peek="267"
  SarcomereLength="3.2"/>
</CalibrationTable>
```

Window parameters

Store Load Reset

#	ROI P	ROI W	Peak	Sarc Length
1	1	100	0	0.00
2	0	0	0	0.00
3	0	0	0	0.00
4	0	0	0	0.00
5	0	0	0	0.00
6	0	0	0	0.00
7	0	0	0	0.00
8	0	0	0	0.00
9	0	0	0	0.00
10	0	0	0	0.00

Interpolation

Submit Exponential Calculate

Figure 11. Calibration table

Buttons of the calibration table:

Store

Saves the calibration table to a file. The file will contain the table in XML format.

Load

Loads the calibration table from a file.

Reset

Clear the calibration table. Resets all rows to zero.

Calculate

Calculates the lookup table from the given fix points.

Submit

Sends the lookup table to the embedded computer.

Interpolation algorithms:

Exponential

Calculates the lookup table from the fixed points using standard exponential calibration. The algorithm calculates (a, b) parameters and the lookup table (z_i) is calculated using these parameters in the following way:

$$z_i = ae^{bx_i}$$

The two parameters are calculated so that the mean square error (mse) is minimized.

$$mse = \frac{\sum_{i=0}^{n-1} |z_i - y_i|^2}{n}$$

n: number of fixed points

z_i : value of the calculated curve at the position of the i-th fixed point

y_i : value of the i-th fixed point

Spline

Fits a cubic spline to the fixed points.

ExtSpline

Calculates the border points (for $x=0$ and $x = 2047$ positions) using linear interpolation from the fixed points. Then calculates the lookup table by fitting a cubic spline to the fixed points and border points.

2.4.4 Sarcomere Length playback tab panel

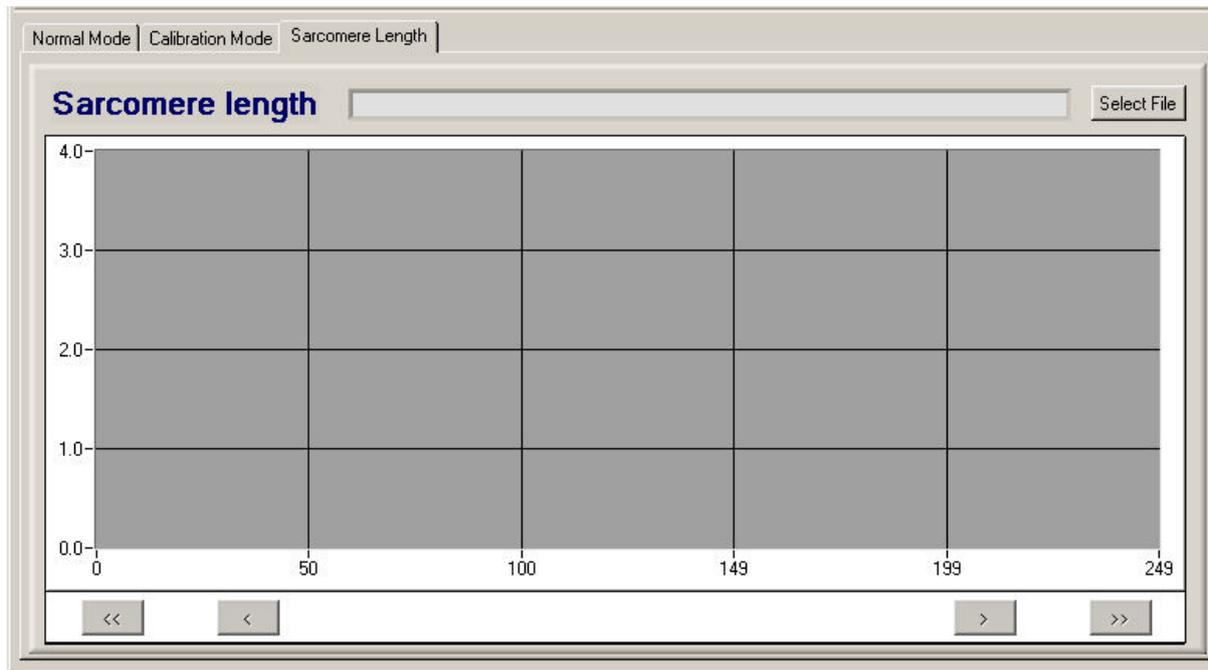


Figure 12. The playback panel

The content of a saved archive file can be visualized on this panel. The archive file contains sarcomere length values.

Select File

Selects an archive file

>

Short step forwards in the file

>>

Long step forwards in the file (10*short)

<

Short step backwards in the file

<<

Long step backwards in the file (10*short)

2.5 Definition of IP address

Some limited operations can be made on the IP address with a popup panel from the Network menu (see Section 2.4.1).



Figure 13. IP address panel

The sensor actually operates on a fixed private TCP/IP address of 192.168.1.98. This value is hardcoded into the sensors' embedded processor, therefore it cannot be modified! The network has to be correctly configured for normal operations. This means that no other element on the same local area network may use the 192.168.1.98 IP address and that the active components of the network (switches, routers, etc.) should work properly.

Local

Sets the loopback IP address to 127.0.0.1 and Port to 5193

Default

Sets the 192.168.1.98 Address and 5194 Port values

Replace

Changes the current address to the specified. This should be used if the sensors' embedded processor connected to the GUI PC through a router and the router replaces the hardcoded TCP/IP address.

OK

Commits and saves the new IP address and Port into the config.ini file

Cancel

Cancels the address change