Solarscan Software System

User Manual

Version 1.0

Document History

Version	Author	Changes
Version 1.0 2018-8-14	AO	Initial Version.

Content

1 Overview

The Solarscan Software system runs a set of hardware devices for measuring the solar radiation on a day by day basis in an automated process using a spectral radiometer.

Originally the Solarscan software system has been designed for scanning spectrometers of the manufacturer BENTHAM using additional instruments as “companion devices”. It includes software for calibrating a spectral radiometer with respect to wavelength accuracy and radiometric response and it includes a system for running devices such as an erymeter, a pyranometer or a digital multimeter as “companion devices”.

The latest instrument added to the Solarscan software system is the BTS2048 array spectrometer manufactured by Gigahertz-Optik.

For the BTS2048 the above mentioned calibration of wavelength accuracy and radiometric response is not necessary any more and in most cases the instrument is not run with companion devices. The spectrometers are calibrated (and recalibrated in regular time intervals) by the manufacturer Gigahertz-Optik.

Nevertheless – if desired - the software calibration can be applied to the array spectrometers and companion devices can be added.

This handbook covers the parametrization of the solarscan system using the BTS2048 array spectrometer and the process of the software calibration.

The parametrization of the scanning spectrometers and the diverse companion devices is planned to be documented in separate handbooks.

1.1 Supported windows platforms

The solarscan software is designed and tested for Windows XP, Windows 7 and Windows 10.

1.2 Supported computer hardware

The solarscan software requires no special computer hardware, except for the monitor. The monitor should be at least full HD resolution (1920 x 1080). The size of the monitor must be 15.6 inch or larger otherwise the fonts used are displayed too small.

1.3 Necessary computer interfaces

The computer needs USB interfaces and/or Ethernet interfaces. When installing the BTS2048 in most cases two Ethernet interfaces are necessary. One for the device and one for the connection to the internal LAN. As many modern Laptops have just USB-C connectivity, the appropriate adapters might be necessary.

Please note:

In many cases the connection to the internal LAN is realized via a VPN connection. Such a VPN connection must not block the connection to the BTS2048. Please refer to you internal IT administrator and ask whether the VPN connection keeps other IP connections open, once it is established.

1.4 Supported measurement and control devices

Fig. 1 shows an overview of the measurement and control devices supported by the Solarscan Software system.

All instruments are controlled via USB, Ethernet or GPIB.

The central device is the BTS2048. It measures the solar radiation via a diffuser integrated into the housing.

On the instrument platform (typically on the roof of the measurement station) the BTS2048 and a set of companion devices like a webcam, an erymeter or sensors for temperature measurement are mounted.

For radiometric calibration of the system, a calibration lamp is connected to the power supply. The calibration lamp is put on top of the diffusor. It includes a sensor head with a silizium diode for monitoring the flux of the lamp during the measurement of the system response of the spectrometer.

When checking or calibrating the wavelengths of the spectral radiometer system a HG pen ray lamp is manually plugged into the housing of the monitor lamp.

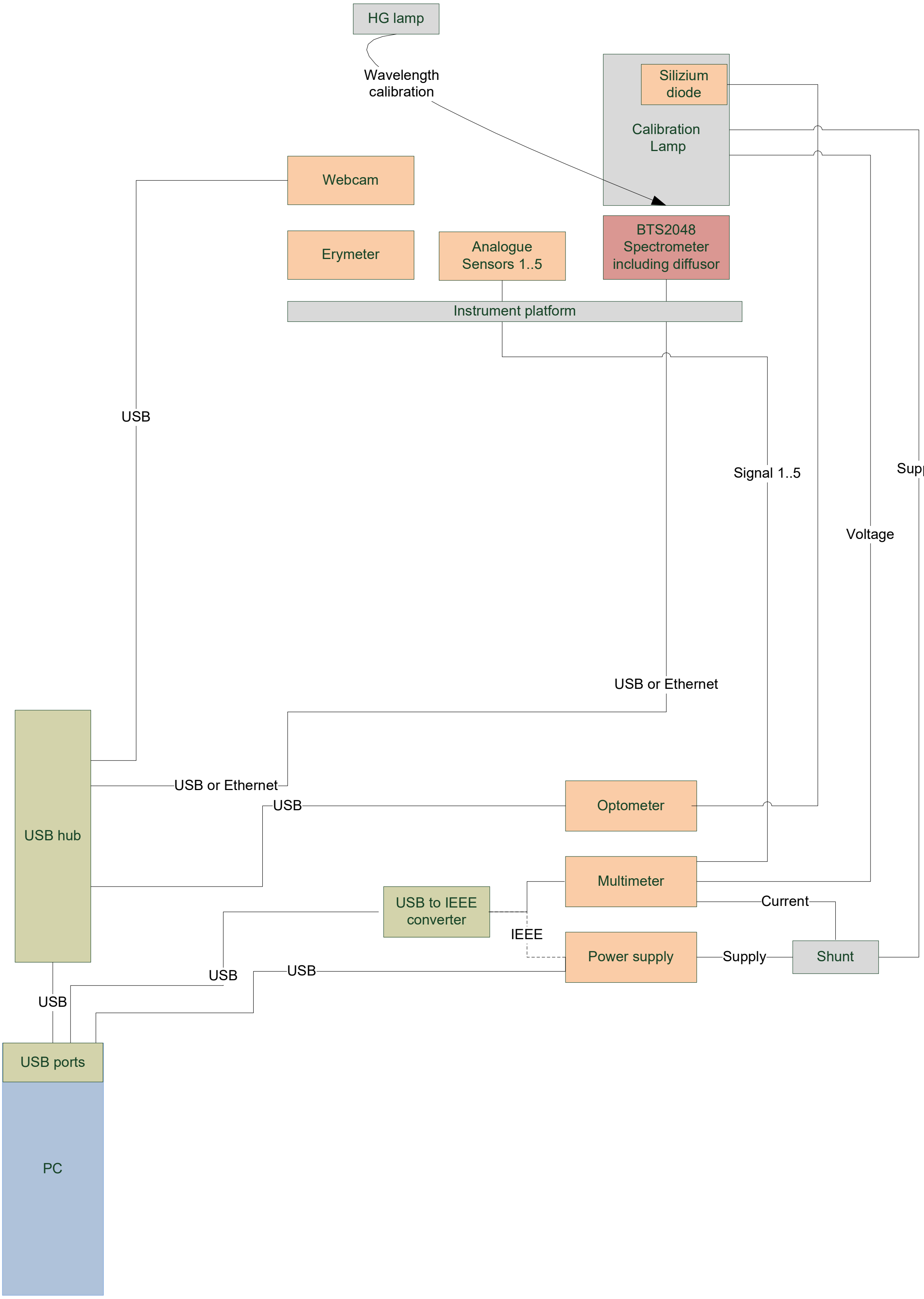


Fig. 1 Hardware and signal flow of the solarscan system using the BTS2048 array spectrometer. The diffusor is integrated into the BTS2048 housing. The diagram shows the maximum possible setup. In most cases just the BTS2048 is used, running the internal radiometric calibration of the instrument.

1.5 Solarscan application overview

The solarscan application comprises the following software modules:

Module	***Usage***
Scheduler.exe	A command line program. Performes the continuous measurement of spectra and point data. Writes results into the local database.
SolarscanGUI.exe	User interface connected with the local database. Designed to observe the progress of the measurements done by the scheduler. Allows to display and analyze the data acquired per day.
SolarscanCalibrationGUI.exe	User interface to check wavelength and radiometric calibration as well as to execute wavelength and/or radiometric calibration.
ListCamera.exe	A utility to identify the webcam connected to the Windows DirectX port.

1.5.1 Configuration

All configuration parameters are contained in text files. The root folder for these configuration files is C:\Solarscan\Config. The root folder and the folder structure itself must not be changed.

Please note:
The content of the folder C:\Solarscan and its subfolders is never changed by the Solarscan software installer.

1.5.2 Software for automated daily measurements

Fig. 2 shows the Solarscan software system for automated daily measurements:

A central database receives all data measured by the Scheduler.exe program doing the automated daily measurements. The Scheduler runs the different hardware modules or group of hardware modules in separate tasks, so called measurement tasks. The advantage of this architecture is, if one of the tasks fails, the other tasks do not stop. A special form of a measurement task is a data export task. The data export tasks are used for writing the file formats used by the Bfs/UBA UV measurement network: the OR0, SC0 and RED file formats (please refer to the BfS/UBA for the documentation of these file formats).
The SolarscanGUI connects to the database and allows to graphically visualize the measured data. A dedicated TCP port allows to supervise the health state of the Scheduler and the different measurement tasks.

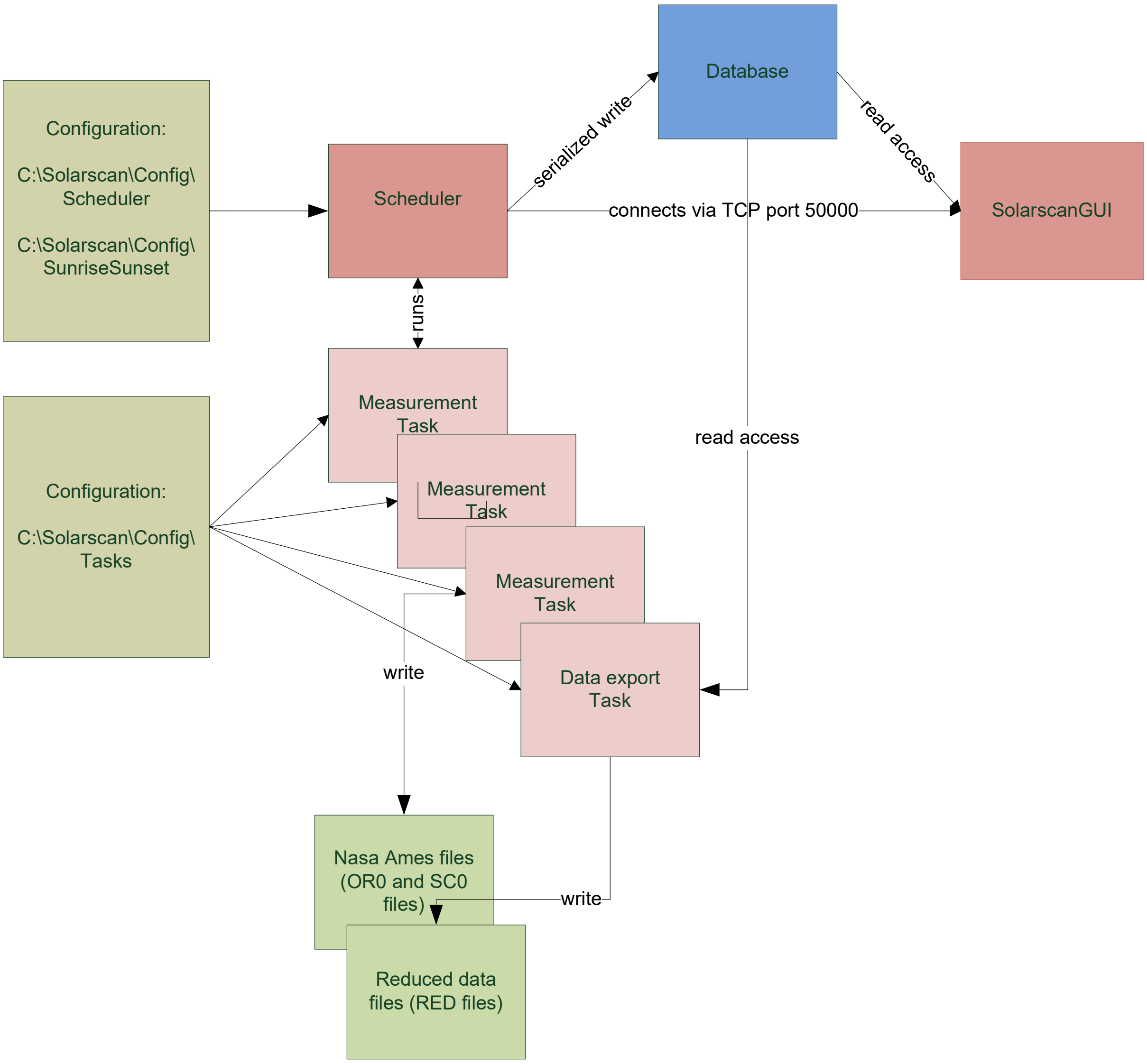


Fig. 2 The Solarscan software system for automated daily measurements.

1.5.3 Calibration software

The software to be used for radiometric and wavelength calibration is the Solarscan-CalibrationGUI. This program accepts the same parameter files for setting up the hardware for calibration, as the Scheduler does for setting up the hardware for the daily measurements. Internally it runs the same measurement tasks (see Fig. 3). Because of this, the SolarscanCalibrationGUI might also be used for testing the parameter files used for setting up the respective measurement tasks in the Scheduler.

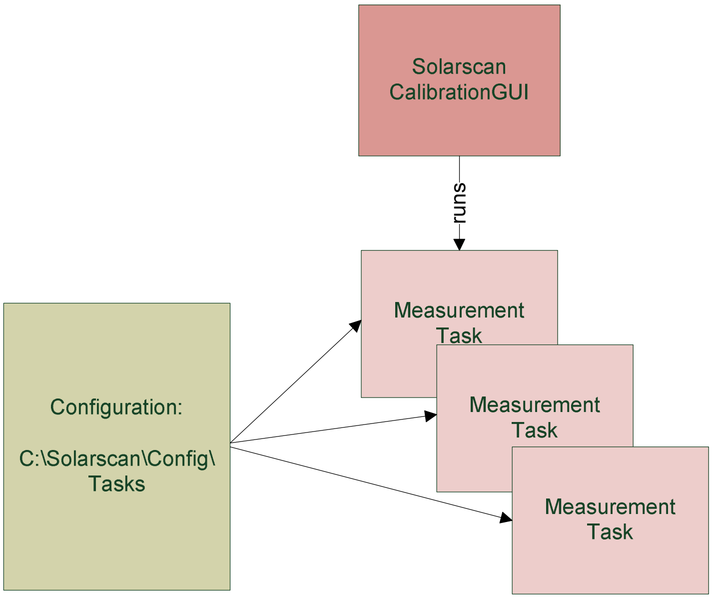


Fig. 3 The SolarscanCalibrationGUI runs the same measurement tasks as the Scheduler.

2 Installation

For installing the Solarscan software system under Windwos you need to login as an administrator.

2.1 Tools

2.1.1 Notepad++

The notepad++ program is recommended for convenient editing of the configuration files.

The program can be installed using the following link:

https://notepad-plus-plus.org/download/v7.5.7.html

2.1.2 Firefox

The firefox web browser is needed for the “Apache CouchDB” database tools (please refer to the CouchDB website http://couchdb.apache.org/ for documentation).

The browser can be installed using the following link:

https://www.mozilla.org/de/firefox/

2.2 Hardware drivers

For the BTS2048 no hardware driver is necessary. The Instrument maps to a generic USB port or connects via Ethernet. The Ethernet connection is tranparent to the Solarscan software system. It is handled by the software layer delivered by Gigahertz-Optik.

2.3 Solarscan software system

The Solarscan software system consists of 3 parts. The CouchDB database, the Solarscan application modules and the Solarscan Configuration folder. Each part requires a separate installation.

2.3.1 CouchDB Database

The CouchDB is the database engine used in the Solarscan software system.

For installing the engine please start:

*Deploy\Software\
CouchDBsetup-couchdb-1.0.2_otp_R14B01_spidermonkey_1.8.5.exe*

and follow the instructions.

2.3.2 Solarscan application modules

For installing the Solarscan application modules run the installer:

Deploy\Software\SolarscanApp\SolarScan-installer.msi

And follow the instructions.

2.3.3 The Solarscan Configuration Folder

Copy your individual configuration folder to C:\. The folder is delivered as a backup copy on a CD by Gigahertz-Optik GmbH. The name of the folder is Solarscan.

The folder Solarscan looks like this:

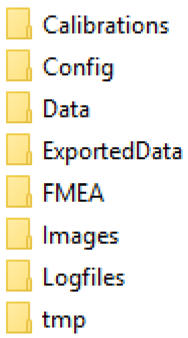


Fig. 4 The content of the configuration folder C:\Solarscan. The folder must be copied to C:.

3 Documentation of the Solarscan software system

3.1 Scheduler

The Scheduler is started with the start_scheduler.bat script. The script is located in the program folder “C:\Program Files (x86)\Gigahertz-Optik GmbH\SolarScan”. An icon for starting the script is installed on the desktop.

The script calls the program Scheduler.exe with 4 command line parameters, one for each configuration file:

```
scheduler.exe
C:\SolarScan\Config\Scheduler\Scheduler.cfg
C:\SolarScan\Config\Scheduler\Schedule_dev.dat
C:\SolarScan\Config\Station\Station.dat
C:\SolarScan\Config\SunriseSunset\Schedule_sun.dat
```

The configuration files define global settings as well as time windows, measurement intervals and device configurations for the measurement tasks.

The functionality of the Scheduler is relatively simple (see Fig. 5): On a per day basis for each measurement task registered, the Scheduler runs three activities within the measurement intervals of each time window. The actions are: prepare, measure, cleanup. The activities repeat until the time window closes.

Please note:
The prepare action is done before a measurement interval starts. It overlaps into the measurement interval before. Thus the time for the measure action must be less than the time interval minus the time allocated for the prepare action.

The time for the prepare action is configured in the device file of a measurement task. How long a measurement lasts depends on the configuration of the individual measurement task. If the measure action lasts too long, the next measurement interval is skipped by the Scheduler.

The concrete actions done in these three functions depend on the device.

I.e. for a scanning spectral radiometer the prepare function is used for adjusting the gain such that the point in time of the first wavelength measured is very close to the start of the time interval.

For the BTS2048 array spectrometer the prepare function is used for listing the available calibrations, for setting global parameters and for reading or measuring the premeasured offsets.

The measure action is used for the acquisition of the spectral data.

In case a scanning spectrometer is used, the cleanup action closes the shutter and drives the spectrometer back to the start wavelength. For the BTS2048 array spectrometer the cleanup action is empty.

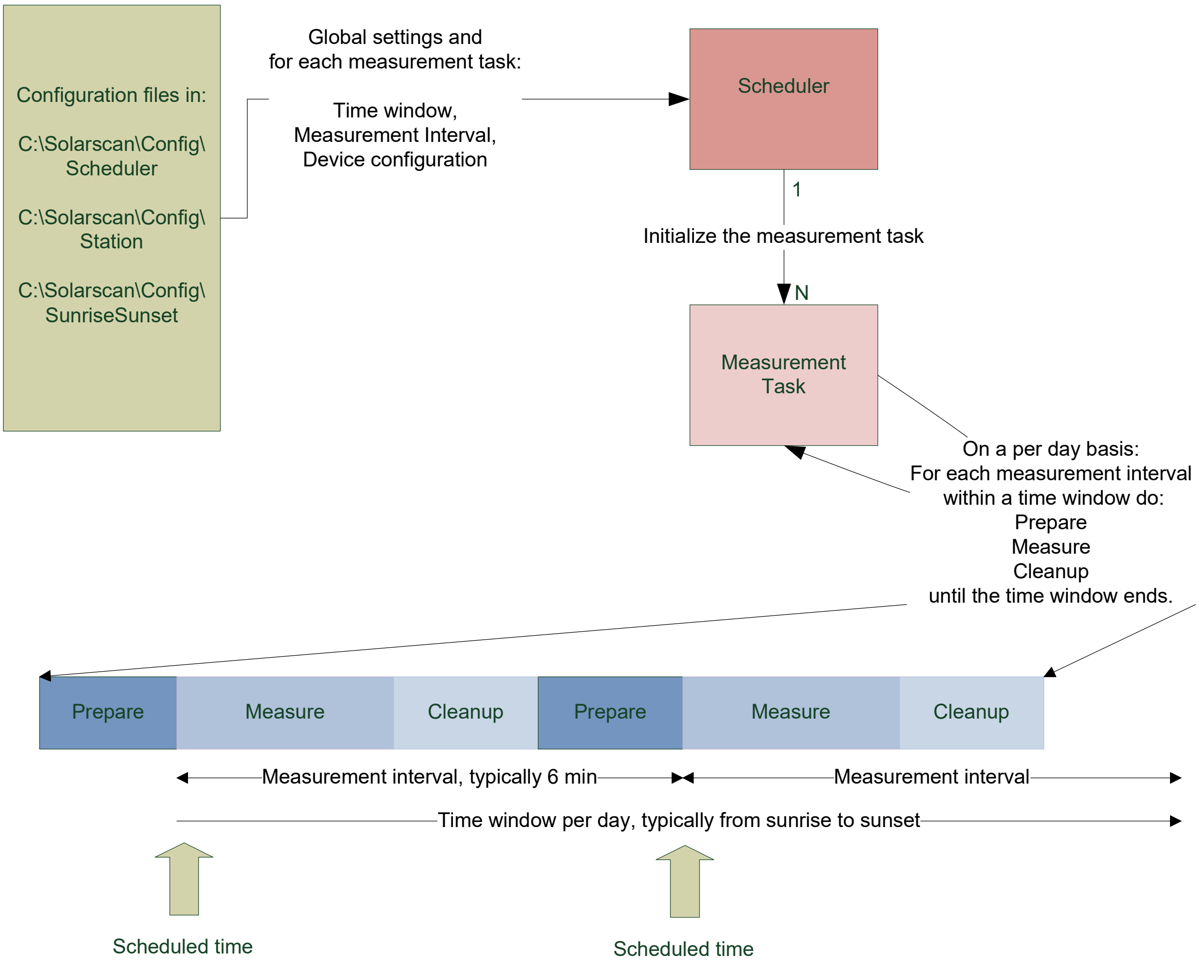


Fig. 5 The functionality of the Scheduler: The Scheduler reads a set of configuration files and runs – based on the information in the configuration files - a set of tasks (threads). Each task performs the actions "Prepare", "Measure" and "Cleanup" repeatedly in intervals of a measurement interval until the daily time window ends.

3.1.1 Global settings

The first command line parameter of the Scheduler.exe program defines the parameter file for the global settings.

3.1.1.1 Sample file

```
[Logging]
Directory=c:\SolarScan\Logfiles
Level=DEBUG|MESSAGE|WARNING|ERROR
KeepLogFiles=10

[Database]
Type=CouchDB
```

```
Host=localhost
Port=5984
Name=solarscan
User=
Password=

[Station]
StationId=${ComputerName}

[Network]
Port=50000
```

3.1.1.2 Parameters

[Logging]

Directory

Directory name for placing the log files. The Scheduler.exe must have write access to this directory.

All logfiles created by the Scheduler are placed under this root folder. For each month a subfolder is opened, numbered from 1-12. For each day per month a subfolder is opened under the month sub-folder, numbered from 1..31.

Default: none, parameter must be specified

Type: string

Level

Debug level. A combination of the keywords:
`DEBUG`
`MESSAGE`
`WARNING`
`ERROR`
Each combination must be concatenated by the pipe sign.

Default: `DEBUG|MESSAGE|WARNING|ERROR`

Type: string

KeepLogFiles

Number of months the logfiles are kept.

Default: `10`

Type: integer

[Database]

Type

Type of the database. At the moment only the CouchDB database is supported. If the entry is wrong a warning is logged and the entry is ignored.

Default: `CouchDB`

Type: string

Host

Network ID of the computer where the database is running. The IP must be the same as the one that is specified as parameter bind_address in the the httpd section of the CouchDB' s default.ini file:

```
[httpd]
port = 5984
bind_address = 127.0.0.1
max_connections = 2048
```

Unless the path is changed during the installation of the CouchDB this file is located in:

C:\Program Files (x86)\Apache Software Foundation\CouchDB\etc\couchdb

Default: `127.0.0.1`

Type: string

Port

TCP port number of the database. This port number must be the same as the one that is specified as parameter port in the the httpd section of the CouchDB' s default.ini file:

```
[httpd]
port = 5984
bind_address = 127.0.0.1
max_connections = 2048
```

Unless the path is changed during the installation of the CouchDB this file is located in:

C:\Program Files (x86)\Apache Software Foundation\CouchDB\etc\couchdb

Default: `5984`

Type: integer

Name

Name of the database used by the Scheduler. The database is created at start of the scheduler if it does not exist.

Default: `solarscan`

Type: string

User

Username of the database account

Default: empty string

Type: string

Password

Password of the database account

Default: empty string

Type: string

[Station]

StationId

Tag used in the header of the data files created (the so called OR0 files).

Allowed values are: Text or `${ComputerName}`

When specifying `${ComputerName}` the Computername defined within the Windows operating system will be used.

Default: `XX`

Type: string

[Network]

Port

TCP port number of the direct connection between the Scheduler and the SolarScanGUI (see Fig. 2). The connection is used for sending the logging information to the SolarScanGUI as well as to send the heart beat and the status of the measurement tasks.

Default: `50000`

Type: string

[Calibrations]

Active

Path where the location of the active calibration is stored. In the configuration delivered this is the path:

C:\SolarScan\Config\CurrentCalibration\currentcalibration.txt

Default: none, parameter must be specified

Type: string

3.1.2 Configuring the schedule

The second command line parameter of the Scheduler.exe program is the file defining the daily start and stop times of each task, the pace of each task, the task name and the parameter file of each task.

The file is organized in rows. Each row holds five entries:

```
Starttime  Endtime   Interval  Taskname      DeviceFile
```

`Starttime` and `Endtime` define the Time window per day in UTC time.
`Interval` defines the measurement interval. For the BTS2048 2 minutes are typically used.

`Taskname` is the name identifying the measurement task. The following tasks are relevant for running the BTS2048:

Taskname	***Explanation***
`BTS2048PRE`	This task is used for running the lengthy (1h or more) offset measurement for all possible integration times upfront, at the beginning of a day. Data are stored in a file defined by the parameter `PremeasuredOffsetFile` (see 3.1.6.2).
`BTS2048`	Runs the acquisition of the spectral data. Data are stored in the CouchDB database.
`NasaAmes`	Runs the file export of the spectral data (“OR0 files”).
`ReducedData`	Runs the file export of the reduced data (“RED files”).

`DeviceFile` is the path of a file defining the parameters of the task.

The following sample file defines a schedule for a setup using the BTS2048:

```
# Comment

02:00:00   02:10:00   00:02:00   BTS2048PRE     C:/SolarScan/Config/Tasks/BTS2048PRE.dev
sunrise    sunset     00:02:00   BTS2048        C:/SolarScan/Config/Tasks/BTS2048.dev
sunrise    sunset     00:06:00   NasaAmes       C:/SolarScan/Config/Tasks/NasaAmes_Normal.dat
sunrise    sunset     00:06:00   ReducedData    C:/Solarscan/Config/Tasks/ReducedData.dat
```

In this example the daily measurement starts with the task `BTS2048PRE`, measuring the offset at 2’ clock in the morning. Such a measurement takes up to 4 hours, so the time window of 10 minutes will be finished once the offset measurement is done.

In the next line, at sunrise, the spectral measurement task `BTS2048` starts at a pace of 2 minutes.

The third and forth lines define the tasks `NasaAmes` and `ReducedData`, exporting the text files needed by the BfS/UBA UV measurement net. In the sample shown, the files are created every 6 minutes.

`sunrise` and `sunset` are keywords. The lookup table behind these keywords is defined in the configuration file which is listed in the forth position of the Scheduler.exe command line parameters (see 3.1).

3.1.3 Station settings

3.1.3.1 General

The third command line parameter of the Scheduler.exe program defines the parameter file for the station settings.

3.1.3.2 Sample file

```
Prefix=NH
ONAME=S.LORENZ
ORG=BfS
ELONG=11.583
NLAT=48.227
HEIGHT=493
MNAME=s-UV-Mo-Net
```

3.1.3.3 Parameters

Prefix

Prefix for the filenames of the NasaAmes and RED files.

Default: XX

Type: string

ONAME

ONAME entry in the NasaAmes and RED files.

Default: ############

Type: string

ORG

ORG entry in the NasaAmes and RED files

Default: ############

Type: string

ELONG

Geographical longitude of station in deg. Part of the SNAME entry in the NasaAmes and RED files.

Default: 0.0

Type: float

NLAT

Geographical latitude of station in deg. Part of the SNAME entry in the NasaAmes and RED files.

Default: 0.0

Type: float

HEIGHT

Geographical Height of station in m. Part of the SNAME entry in the NasaAmes and RED files.

Default: 0

Type: integer

MNAME

MNAME entry in the NasaAmes and RED files

Default: ############

Type: integer

3.1.4 Configuring sunrise/sunset times

3.1.4.1 General

The forth command line parameter of the Scheduler.exe program defines the parameter file for the sunrise and sunset times. The idea of this configuration file is to centrally define variable time windows between sunrise and sunset. These times are then referenced by the symbolic names `sunrise` and `sunset` when setting up the daily time windows of the measurement tasks.

It is just necessary to define the points in time where these time windows change over the year. The format of the file is the following: In the `MMDD` column the month and day where the times change are listed. The other two columns define the sunrise and sunset times respectively. To define seconds is optional.

All times are to be entered in UTC time.

3.1.4.2 Sample file

```
# All times in UTC
```

```
# http://www.srrb.noaa.gov/highlights/sunrise/sunrise.html
#  date      sunrise     sunset
#  MMDD       hh:mm[:ss]  hh:mm[:ss]
   0102       07:17       15:05
   0201       06:49       15:53
   0301       05:54       16:45
   0401       04:42       17:40
```

3.1.5 Configuring the measurement tasks

3.1.6 BTS2048

This task is used for measuring the solar spectrum. In principle it is possible to include the premeasurement of the offsets in the prepare action of this task. As the premeasurement of the offsets is a very lengthy task, it does not fit into the time window of a normal task. Because of this it is possible to configure the premasurement of the offsets in a specialized task doing just this. The name of the specialised task is BTS2048PRE (see 3.1.7).

3.1.6.1 Sample file

```
/*--------------------------------------------------*/
/*                 Device setup                     */
/*--------------------------------------------------*/
[Device]
Type   = Bentham # for database compatibility
Serial = 35861

[Data]
DataSet = NormalScan

/*--------------------------------------------------*/
/*                 Bentham calibration file         */
/*--------------------------------------------------*/
[Calibration]
#CalibrationFile                                                          =
C:\SolarScan\Config\Calibration\20110714_2123_01L1700_SRN_12257.dat
CalibrationDir = auto
Unit=W/sqm/nm


/*--------------------------------------------------*/
/*               Initialize                         */
/*--------------------------------------------------*/
[Initialize]
CalibrationEntry = 48
CoolingFitted = true
UseCooling = true
UseInternalCal = true
UsePixelLinearization = true
UseAdvancedNoiseReduction = true
MaxMeasurementTimeInUs = 55000000
IterationsToAvoidOverload = 100
#UseHighResolutionMode = true #evtl. auskommentieren
NrOfScansForPremeasurement = 1
NrOfScansForMeasurement = 1
UsePremeasurement = true #default false
LoadPremeasuredOffsetFile = true
PremeasuredOffsetFile = C:\Solarscan\tmp\BTS2048PremeasuredOffset.bin
```

```
JustDoPremeasurement = false
MaxValidMeasurementTimeInUs = 15000000000 #default 170000000
MinValidMeasurementTimeInUs = 1000 #default 1000

/*--------------------------------------------------*/
/*               Prepare                            */
/*--------------------------------------------------*/
PrepareDuration = 10000


/*--------------------------------------------------*/
/*               Wavelength                        */
/*--------------------------------------------------*/
[Wavelength]
StartWavelength = 290.0
EndWavelength   = 400.0
WavelengthStep  = 0.1
#WavelengthFile   = C:/SolarScan/Config/Wavelengths/NORMAL_419.WVL


/*--------------------------------------------------*/
/*       Peak checking                              */
/*--------------------------------------------------*/
[PeakCheck]
PeakCheckFile=C:\SolarScan\Config\PeakCheck\FraunhoferK.txt
```

3.1.6.2 Parameters

[Device]

Type

The instrument type in a xy dataset written to the database. The instrument type is a key for selecting data from the database. It must not be changed. For compatibility reasons the BTS2048 gets the same instrument type as the scanning spectrometers manufactured by Bentham. I

Default: `Bentham`

Type: string

Serial

The instrument id in a xy dataset written to the database. The parameter is typically set to the serial number of the device.

Default: none, parameter must be specified

Type: string

[Data]

DataSet

This is the dataset field in a xy dataset written to the database. The field is used for differentiating the normal and the special scan datasets when exporting the data to the NasaAmes files. The parameter must be set to either NormalScan or SpecialScan. For scanning spectrometers these two scans are scans with different wavelength resolutions. The special scan is used for a high resolution spectrum acquired once a day. For the BTS2048 array spectrometer we do not need this concept as the length of a measurement is independent from the spectral resolution. For compatibility it is possible to tag the measurement as a `NormalScan` or as a `SpecialScan`.

Default: `NormalScan`

Type: string

[Calibration]

CalibrationFile

The path of the radiometric calibration of the spectrometer. Only relevant if the internal calibration of the BTS2048 is not used, i.e. if `UseInternalCal = false` and if the parameter `CalibrationDir` is empty.

Default: none, parameter must be specified – but only of the parameter `CalibrationDir` is empty. Otherwise the parameter is not read.

Type: string

CalibrationDir

A folder with at least two files. One with the extension wca, which is the wavelength calibration file. The other with the extension cal - this is the radiometric calibration file. If the keyword `auto` is specified, the active calibration is used (see parameter Active in 3.1.1.2).

Only relevant if the internal calibration of the BTS2048 is not used, i.e. if `UseInternalCal = false.`

Default: none, parameter must be specified.

Type: string

[Initialize]

CalibrationEntry

The BTS2048 is fitted with a set of calibrations. These datasets are numbered. For a measurement it is necessary to define which of those calibrations shall be used. The available calibrations are listed during the prepare action of the measurement task.

Default: none, parameter must be specified

Type: integer

CoolingFitted

Flag indicating whether the sensor cooling option is fitted.

Default: `true`

Type: boolean

UseCooling

Flag indicating whether the sensor cooling shall be activated. If `CoolingFitted=false` (see above) the parameter UseCooling is ignored.

Default: `true`

Type: boolean

UseInternalCal

Flag indicating whether the internal calibration of the BTS2048 shall be used. We recommend to use the internal calibration.

Default: `true`

Type: boolean

UsePixelLinearization

Switch to activate/deactivate the linearization of the signal done by the BTS2048 software layer. Must be set to true for getting the correct calibrated signal.

Default: `true`

Type: boolean

UseAdvancedNoiseReduction

Switch to activate/deactivate the so called “advanced noise reduction” of the signal done by the BTS2048 software layer. Must be set to true for getting the best results in the UV.

Default: `true`

Type: boolean

MaxMeasurementTimeInUs

Defines the total time used for a single measurement. When measuring UV spectra a single measurement is a combination of several spectra measured with different filters and (dependent on the Parameter UseHighResolutionMode, see below) integration times. The parameter defines the maximum total amount of time of such a combined measurement. The unit is microseconds. Values of 2 to 360000000 (6 minutes) are allowed.

We found that a value of 55000000 (55 seconds) already gives a very good result for UV spectra. When measuring the offset we recommend to set the value to 179000000 (179 seconds).

Default: `55000000`

Type: integer

IterationsToAvoidOverload

When measuring solar spectra it is always possible to get an overload of the sensor due to a change of the signal during the measurement. Thus we repeat a measurement until we get a valid signal. This parameter defines how many times a measurement may be repeated without throwing an error.

Default: `100`

Type: integer

UseHighResolutionMode

Activates/deactivates the so called high resolution mode of the BTS2048. When activated, the measurement is a combination of several single spectra measured at different integration times. The idea is to improve the signal to noise ratio in the weak UV region without overloading the sensor at longer wavelengths. As per today, it is recommended to deactivate this option.

Default: `false`

Type: boolean

NrOfScansForPremeasurement

Defines the number of spectra measured for calculating the mean value of the premeasured spectral offset. The higher this number is, the better the signal to noise ratio of the offset measurement.

Default: `70`

Type: integer

NrOfScansForMeasurement

Defines the number of spectra measured for calculating the mean value of the final spectrum. The higher this number is, the better the signal to noise ratio of the final spectrum – as long as the source is stable. Because the solar spectrum is never really stable we recommend to set this parameter to 1.

Default: `1`

Type: integer

UsePreMeasurement

If set to true, the spectral offset is taken from the premeasured offset (either from memory or from file). If the parameter `LoadPreMeasuredOffsetFile` is set to `true`, the offset is read from file. In contrast to this, if `LoadPreMeasuredOffsetFile` is set to `false`, the measurement of the offset is done in the prepare action of the measurement task. And finally, if the parameter `PremeasuredOffsetFile` is not set to `NotInUse`, the premeasured offset is written to a file with the filename defined in the parameter `PremeasuredOffsetFile`.

Default: `true`

Type: boolean

LoadPremeasuredOffsetFile

Flag to indicate whether the premeasured offset file shall be loaded in the prepare action of the task. Only relevant if `UsePreMeasurement = true`.

Default: `true`

Type: boolean

JustDoPremeasurement

Flag to indicate whether just the premeasurement of the offset file shall be done in the prepare action of the task and nothing in the measure action.

Only senseful if `UsePreMeasurement = true`.

Default: `false`

Type: boolean

PremeasuredOffsetFile

Path of the file to store the premeasured offset file or to read the premeasured offset. If the name is `NotInUse` nothing is written to disk and the data are kept in memory until the end of the daily time window. The folder, i.e. `C:\\Solarscan\\tmp` must already exist. Only relevant if `UsePre-Measurement=true` and (for reading) and if `LoadPremeasuredOffsetFile=true`.

Default: `C:\\Solarscan\\tmp\\BTS2048PremeasuredOffset.bin`

Type: string

MaxValidMeasurementTimeInUs

This parameter is an upper limit for the allowed total measurement time as reported by the BTS2048 software layer after a measurement has been done. If the reported measurement time is longer than this threshold, the measurement is repeated. The maximum number of times a measurement is repetad is set in `IterationsToAvoidOverload`.

Default: `179000000`

Type: integer

MinValidMeasurementTimeInUs

This parameter is a lower limit for the allowed total measurement time as reported by the BTS2048 software layer after a measurement has been done. If the reported measurement time is shorter than this threshold, the measurement is repeated. The maximum number of times a measurement is repetad is set in `IterationsToAvoidOverload`.

Default: `6000`

Type: integer

[Prepare]

PrepareDuration

Length of the prepare action in milliseconds.

Default: 10000

Type: integer

[Wavelength]

StartWavelength

First wavelength in an aquired spectrum. Must be `0.0` if the wavelengths are read from file.

Default: `0.0`

Type: float

EndWavelength

Last wavelength in an aquired spectrum. Must be `0.0` if the wavelengths are read from file.

Default: `0.0`

Type: float

WavelengthStep

Wavelength step in an aquired spectrum. Must be `0.0` if the wavelengths are read from file.

Default: `0.0`

Type: float

WavelengthFile

File holding the wavelengths. Reading the wavelengths from file allows to define non equidistant wavelength tables.

Default: empty string

Type: string

[PeakCheck]

3.1.7 BTS2048PRE

This task is performing the premeasurement of the offset data. The premeasurement is done in the prepare action of the task. Nothing is done in the other actions.

3.1.7.1 Sample file

```
/*--------------------------------------------------*/
/*                 Device setup                     */
/*--------------------------------------------------*/
[Device]
Type   = Bentham
Serial = 35861


/*--------------------------------------------------*/
/*               Initialize                         */
/*--------------------------------------------------*/
[Initialize]
CalibrationEntry = 48
UseCooling = true
UsePixelLinearization = true
UseAdvancedNoiseReduction = true
MaxMeasurementTimeInUs = 179000000
#UseHighResolutionMode = true
NrOfScansForPremeasurement = 70
PremeasuredOffsetFile = C:\Solarscan\tmp\BTS2048PremeasuredOffset.bin
```

```
/*--------------------------------------------------*/
/*                Prepare                           */
/*--------------------------------------------------*/
PrepareDuration = 10000
```

3.1.7.2 Parameters

The subset of the parameters that are meaningful for the BTS2048PRE task are already described above (see 3.1.6.2). The subset comprises the following parameters:

CalibrationEntry

See 3.1.6.2

CoolingFitted

See 3.1.6.2

UseCooling

See 3.1.6.2

UsePixelLinearization

See 3.1.6.2

UseAdvancedNoiseReduction

See 3.1.6.2

MaxMeasurementTimeInUs

See 3.1.6.2

UseHighResolutionMode

See 3.1.6.2

NrOfScansForPreMeasurement

See 3.1.6.2

PreMeasuredOffsetFile

See 3.1.6.2

PrepareDuration

See 3.1.6.2

3.2 SolarscanGUI

3.2.1 Overview

The purpose of the SolarscanGUI is two things:

- Live view for the data measured by the Scheduler
- Inspection (on a per day basis) of data in the database

The SolarscanGUI is started with the SolarscanGUI shortcut installed on the Desktop. The SolarscanGUI is connected to the database and it is connected to the

Scheduler. Both connections are network connections. Such the SolarscanGUI can run on a remote PC, as long as a network connection to the measurement system exists.

The default link to the database is:

http://<hostname>:5984/<database name>

The default link to the Scheduler is:

<hostname>:50000

The ports of the two network addresses as well as the database name are configured in the first of the four configuration files the Scheduler takes as command line parameters (see 3.1). If the SolarscanGUI runs on the same machine as the Scheduler and if the database also runs on that machine (which is the default) we can set the hostname to : 127.0.0.1, or localhost. The default for the database name is solarscan.

At the very first start of the Scheduler on a machine where it has never been installed these two connections must be entered manually into the respective fields of the SolarscanGUI:

Fig. 6 Enter the database connection in the Database field. Press Connect if a new connection is selected.

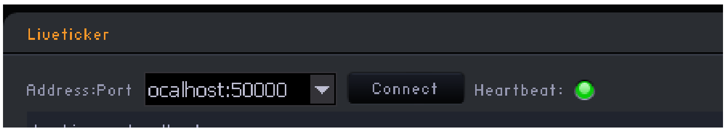


Fig. 7 Enter the network identification of the computer running the Scheduler.

The following screenshot shows an overview of the windows available in the SolarscanGUI:

Splitter – grab and move with mouse to enlarge/shrink part of the screen (works in th evertical direction only)

Fig. 8 The window layout of the SolarscanGUI. Three splitters allow to adjust the layout.

All windows are organized in a single screen. To enlarge/shrink a window use the mouse and grab one of the splitters to pull in the vertical direction (when grabbed with the mouse the splitter gets yellow).

3.2.2 Detailed description

3.2.2.1 Database connection window

TBC

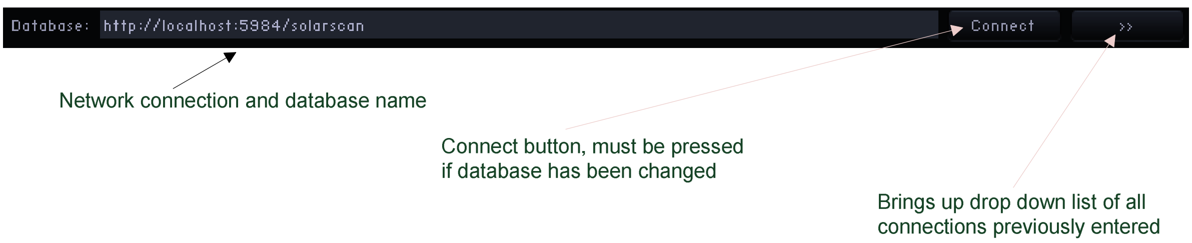


Fig. 9 The database connection window.

3.2.2.2 Database information window

TBC

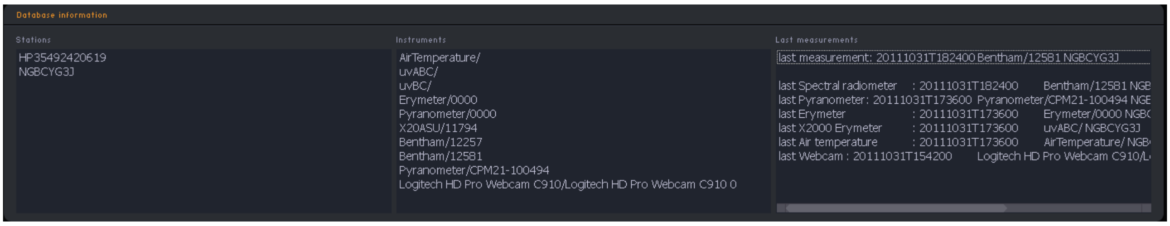


Fig. 10 The Database Information Window

3.2.2.3 Live ticker and dashboard window

Here the log of the scheduler is listed as a permanent log and the status of all tasks run by the scheduler is shown. A green “LED” indicates that the task is running, a red “Led” is indicating that an error has occurred. If anerror has occurred it is listed next to the task.

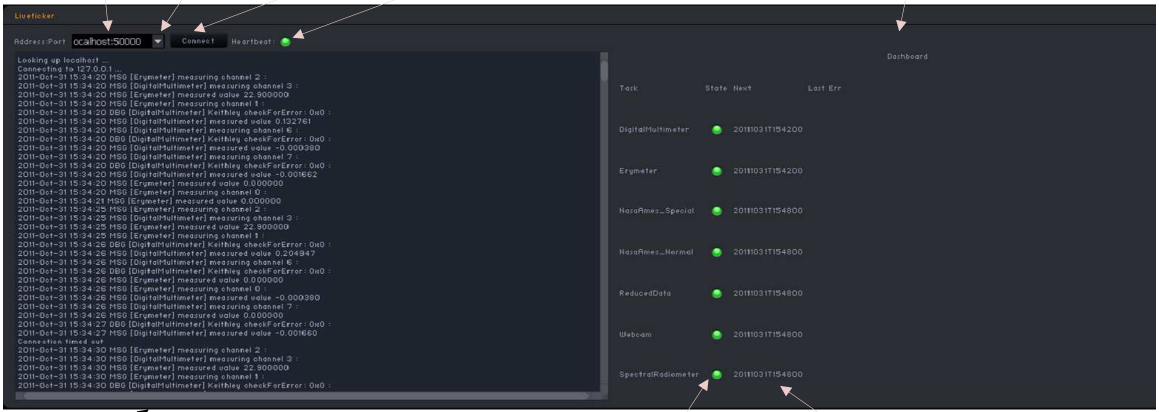


Fig. 11 The Live Ticker and dashboard window.

3.2.2.4 Spectra window

In the spectra windows spectra that have been written to the database can be inspected. If Auto Update is selected the windows get updated with every new spectrum measured.

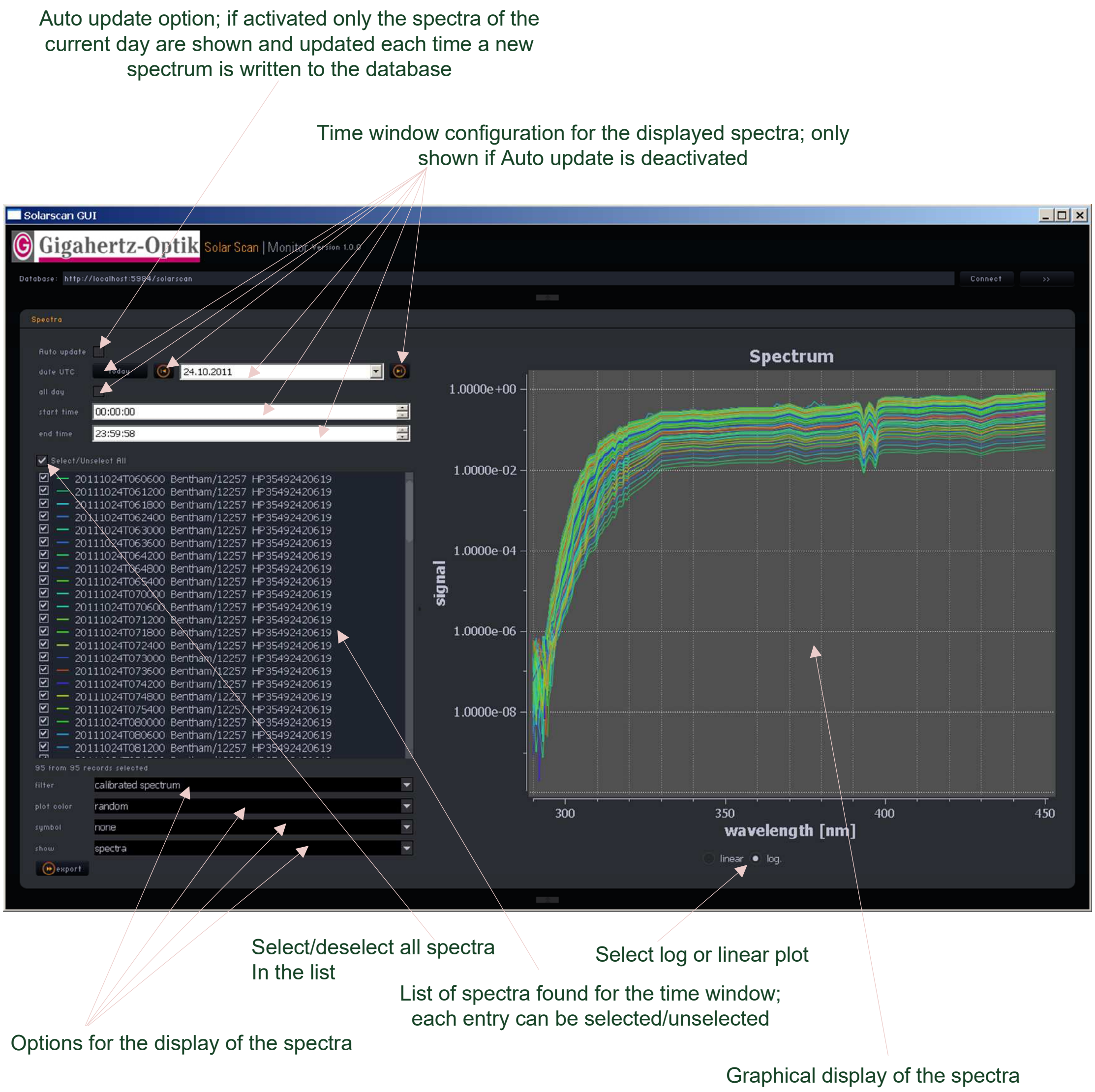


Fig. 12 The Spectra window

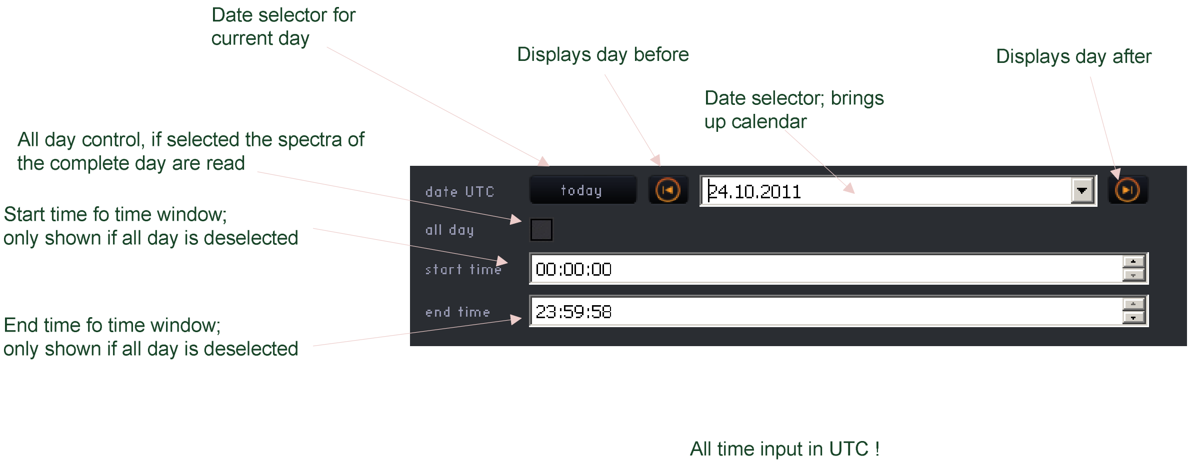


Fig. 13 Controls for setting database filters for the data shown in the spectra window

3.2.2.5 Time series window

The time series window shows data either calculated from the measured spectra or measured by additional devices. The data are shown as functions of the time.

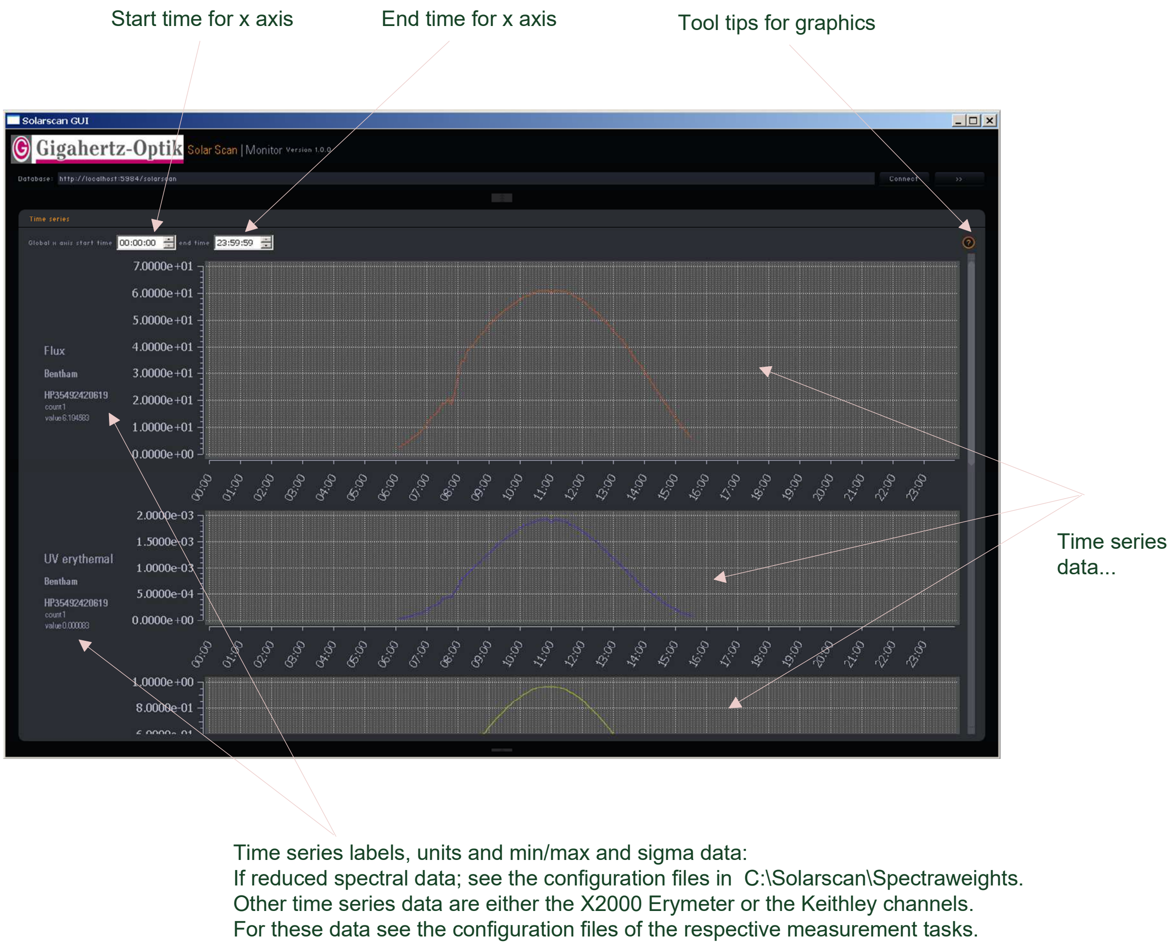


Fig. 14 The Time series window.

3.3 SolarscanCalibrationGUI

3.3.1 Overview

The SolarscanCalibrationGUI serves the following purposes:

- Provide a flow for system check, wavelength calibration and radiometric calibration such that minimum user interaction is necessary and such that the quality control of the results is done by the software (think of it as an expert system).
- Start/stop the calibration lamp control.
- Inspect the results of the single steps in the calibration flow.
- Inspect the time series plots of the calibration lamp control.
- Manage configuration files.

Fig. 15 shows the single elements of the GUI. The start button on top of the GUI starts a calibration run. Below we find three panels. The left one is called the calibration task panel. This panel will be filled with the parameters and controls of the single steps in the calibration flow. The second one is a button and a status display controlling the digital IO of the ASU cover. The third one is the Start/Stop button and several status displays for the calibration lamp control.
Below these three panels a window is placed showing the log output of all hard and software components active.
The lower part of the GUI is a Tabbed window where the results of the calibration steps are shown. For each step in the calibration flow a separate tab windows exists. In addition a window exists where the time series plots of the calibration lamp control can be inspected and one for managing the configuration files.

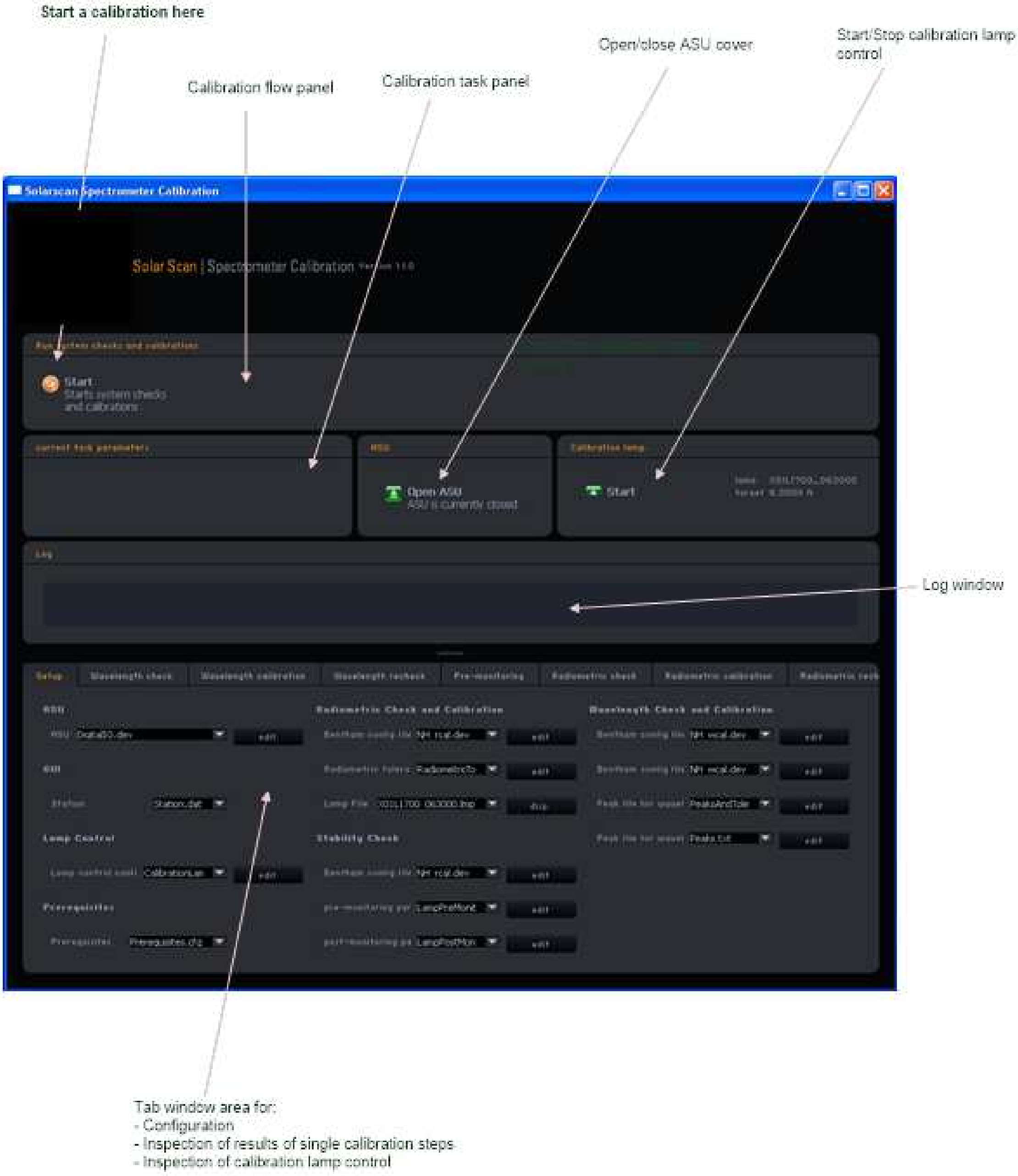
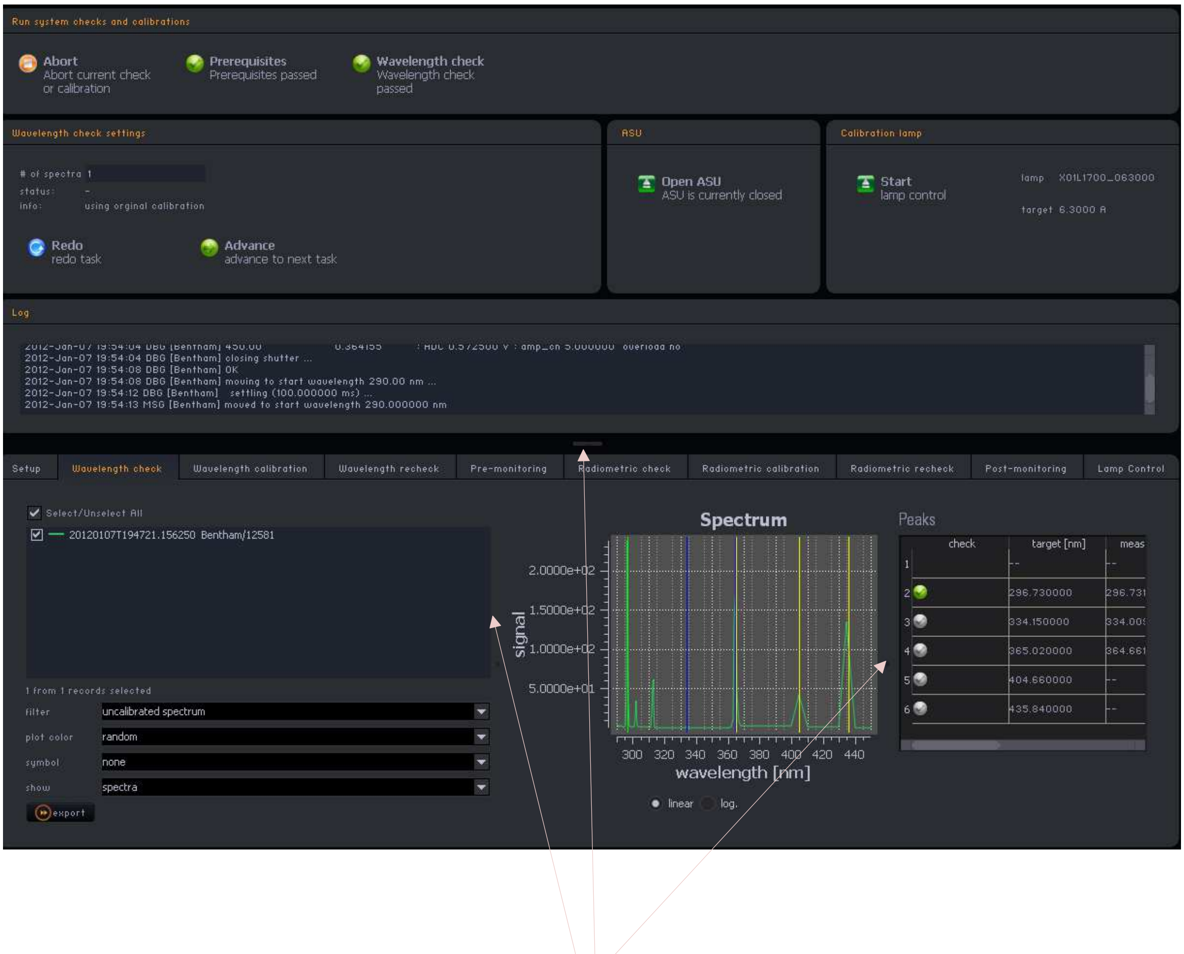


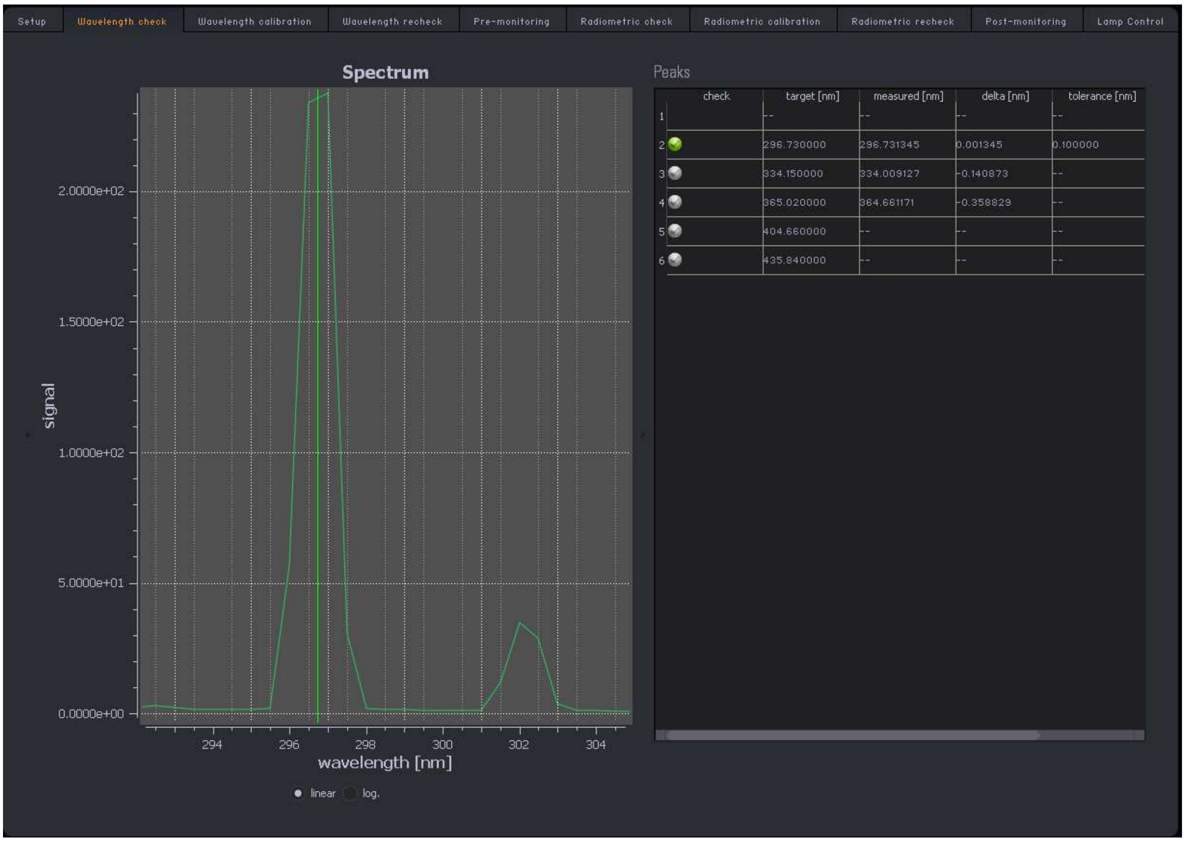
Fig. 15 Overview of the SolarscanCalibrationGUI application

The result panels in the tabbed window can be manipulated in size and adjusted such that only the plot and the result table are shown. In Fig. 16 this is shown for the result panel of the wavelength check step:



Handles for enlarging the result panel and for adjusting the space used for the result table and the plot.

Fig. 16 The result panel of the wavelength check step in default state. Using the three handles allows to zoom such a result window.



Enlarged and adjusted result panel – all the space is used for showing the plot and the resutl table

Fig. 17 The result panel of the wavelength check step in the zoomed state.

This behaviour is implemented for all the result panels.

3.3.2 The calibration flow

Fig. 18 shows a chart of the calibration flow:

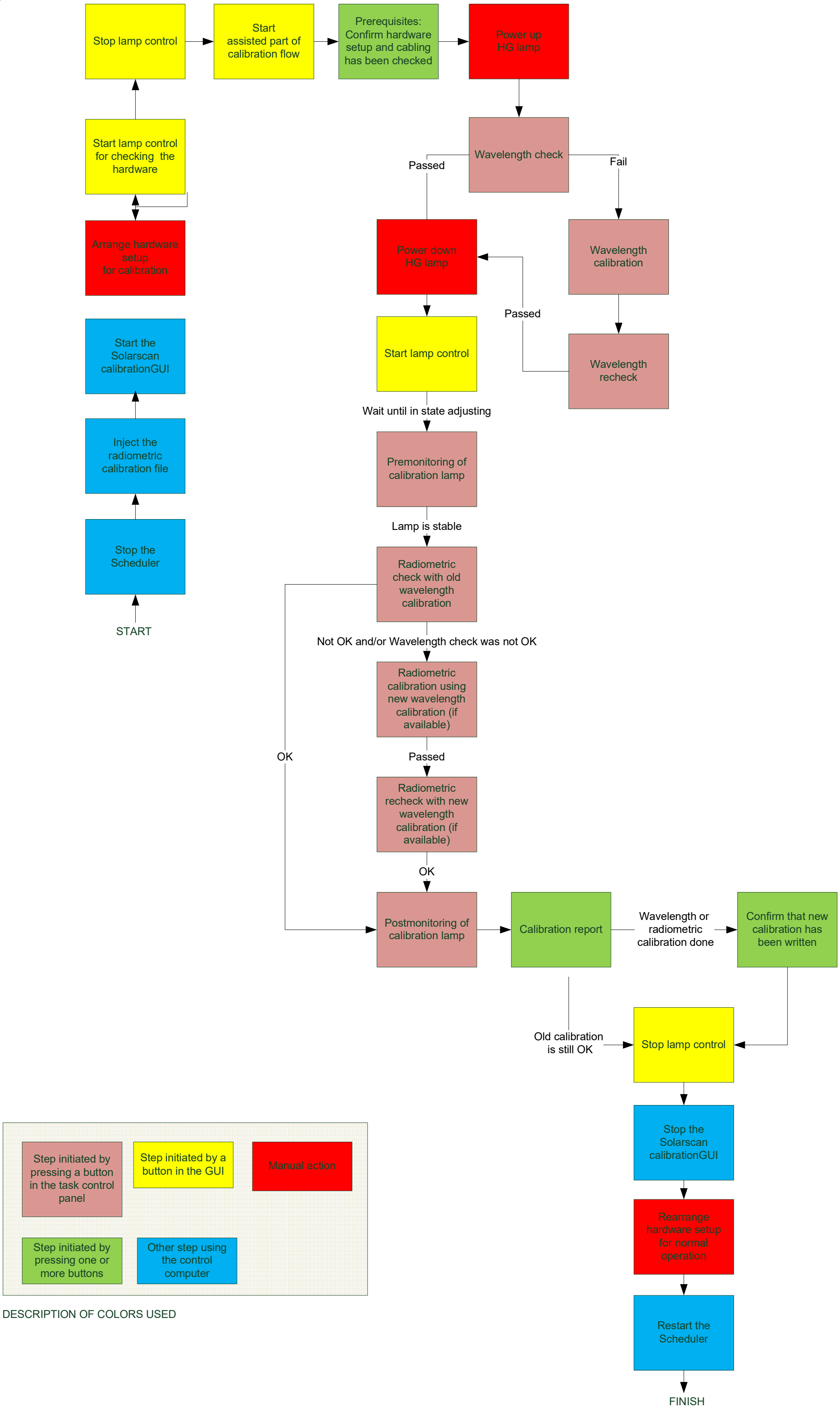


Fig. 18 The calibration flowchart

Description of the colors used in the calibration flowchart

The yellow boxes show the actions to be initiated by explicitely pressing one of the two separate software buttons:

- Start or stop the calibration lamp control
- Start the assisted part of the calibration flow

The bright red boxes represent software-independent pure manual steps to be done by the user within the calibration flow. These are two steps:

- Arranging the complete hardware setup for doing the calibration
- Starting and Stopping the HG pen ray lamp at the appropriate points within the flow.

The light red boxes are the different steps within the calibration flow covered by the software flow control. The user must use the Start, Redo or Advance knobs of the calibration task control panel to progress within the flow (see Fig. 19)

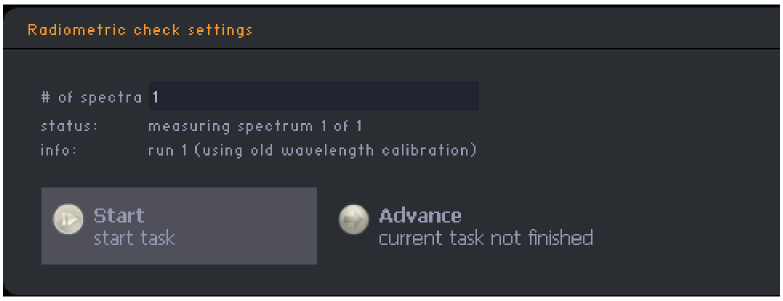


Fig. 19 The task control panel for initiating the different steps of the software flow control (light red boxes in the calibration flow chart above).

The green boxes represent steps to be confirmed by the user.

The blue boxes show the steps done automatically.

Description of the flow

In the following we describe the different steps of the calibration flow.

3.3.2.1 Stop the Scheduler

Close the command window where the Scheduler is running or press Ctrl + C.

3.3.2.2 Arrange hardware setup for calibration

- Clean the diffusor
- Horizontally adjust the diffusor
- Position the calibration lamp in the defined distance to diffusor

- Disconnect the ASU monitor lamp
- Connect the cables of the calibration lamp instead of the cables of the monitor lamp of the ASU unit (see Fig. 1)
- Insert the HG lamp
- Connect the USB cable of the monitor diode

3.3.2.3 Inject the radiometric calibration file

Task description

For calibrating you need the calibration data of the radiometric calibration lamp that will be used. Please check if the file of the calibration lamp is already copied to the folder:
C:\SolarScan\Config\CalibrationGUI\Radiometric

If this is not the case please do copy the file to this folder.

If not already done please rename the filename to the format <name>_######.lmp. ###### stands for the target current the lamp should be operated at. I.e. <name>_063000.lmp would stand for a lamp that must be operated at 6.3000 Amps.

3.3.2.4 Start the SolarscanCalibration GUI

Task description

Start the SolarscanCalibrationGUI application

Please note:

The SolarscanCalibrationGUI must always be started <u>after</u> injecting a new calibration file because otherwise the calibration file will not be “seen” by the application.

Pelase note:

The SolarscanCalibrationGUI must always be started <u>after</u> shutting down the Scheduler because for most parts it uses the same hardware.

Starting the task

Use the icon on the Desktop for starting the SolarscanCalibrationGUI or the entry in the Windows Start menu.

3.3.2.5 Start lamp control

Task description

This step is a “preflight check”. Before we use the hardware we want to know whether everything is correctly cabled and working properly. For this we drive the lamp control up for test. If no error message comes up and the lamp reaches its state “adjusting” we did everything right. The spectrometer needs no preflight check because it is the first instrument to be used in the software assisted calibration flow.

Starting the task

Please refer to “Starting and stopping the lamp control” under 3.3.2.14 in order to learn how the lamp control is started.

3.3.2.6 Stop lamp control

Task description

After we checked the hardware setup we have to drive down the lamp control again because the calibration flow starts with using the HG lamp and not with using the radiometric calibration lamp.

Starting the task

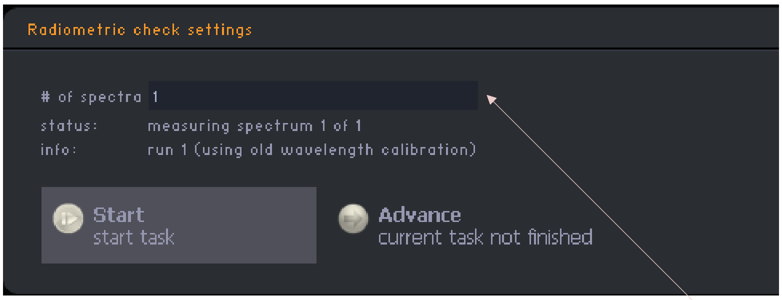
Please refer to “Starting and stopping the lamp control” under 3.3.2.14 in order to learn how the lamp control is stopped.

3.3.2.7 Start assisted part of the calibration flow

Task description

The assisted parts of the calibration flow just require pressing the Start and Advance buttons in the task control panel. If it is desired, the Redo button can be used as many times as wanted. Using the Redo button repeats the specific task.

Redoing a task might make sense because some tasks provide parameters (i.e. the number of spectra to be measured, see Fig. 20). These parameters might be changed in a second run in order to achieve a better result. This is important because once a run with the spectrometer has been started, it cannot be interrupted any more.



Parameter of the radiometric check task:
The number of spectra to be measured can be configured. If more than one spectrum is measured the mean is calculated.

Fig. 20 The task panel provides task specific parameters. In this case it shows the number of spectra to be measured for the radiometric check. If more than one spectrum is to be measured the mean will be calculated.

Starting the task

For starting the assisted parts of the calibration flow press the Start button:

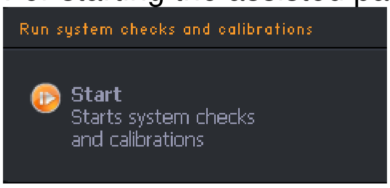


Fig. 21 The Start button of the software assisted calibration flow

This immediately brings up the Prerequisites box:

3.3.2.8 Prerequisites

Task description

The prerequisites box is designed for confirmation that everything has been properly prepared for a precise and accurate check of the calibration or a renewal of the calibration.

The user must confirm every single step and eventually read the informations behind the round buttons indicated with an “I” (see Fig. 22):

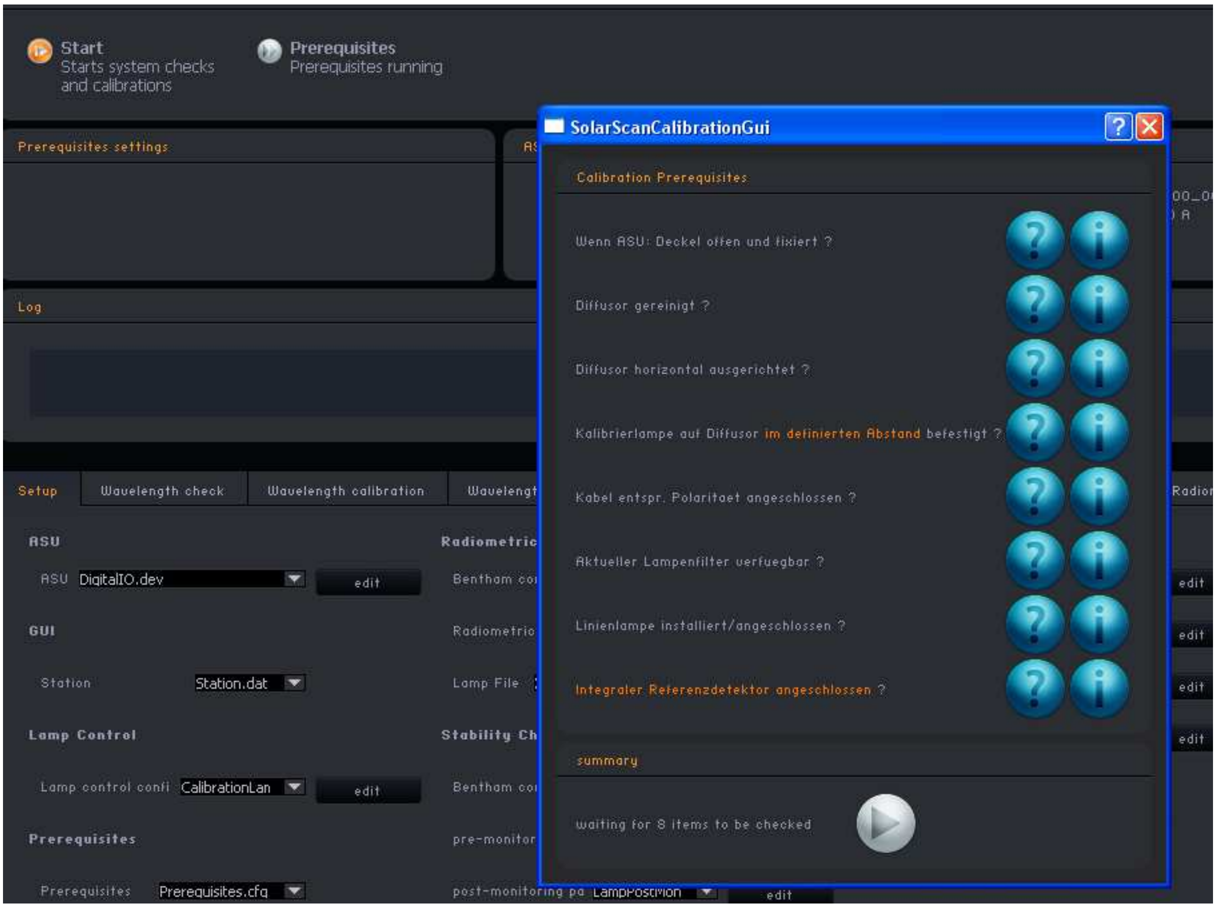


Fig. 22 The prerequisites box in its initial state.

Accepting a single step is done by pressing the green check box in the “Select new state” box coming up when pressing on the round buttons with a question mark (see Fig. 23):

Fig. 23 The select new state box of the prerequisites box

As soon as the user has accepted all the single steps the arrow button at the bottom of the prerequisites box gets green. Pressing this button brings the user to the next step in the software assisted calibration flow which is the wavelength check step (see Fig. 24):

Fig. 24 The prerequisistes box after all items are confirmed

3.3.2.9 Power up the HG lamp

This is a pure manual step without software support. The HG lamp must be switched on.

3.3.2.10 Wavelength check

Task description

The wavelength check is the first step after confirming the prerequisites. In this step the position of a single HG line is measured and protocolled.

Starting the task

For starting the wavelength check press Start in the task panel. The Start knob then gets white and the wavelength check starts (see Fig. 25):

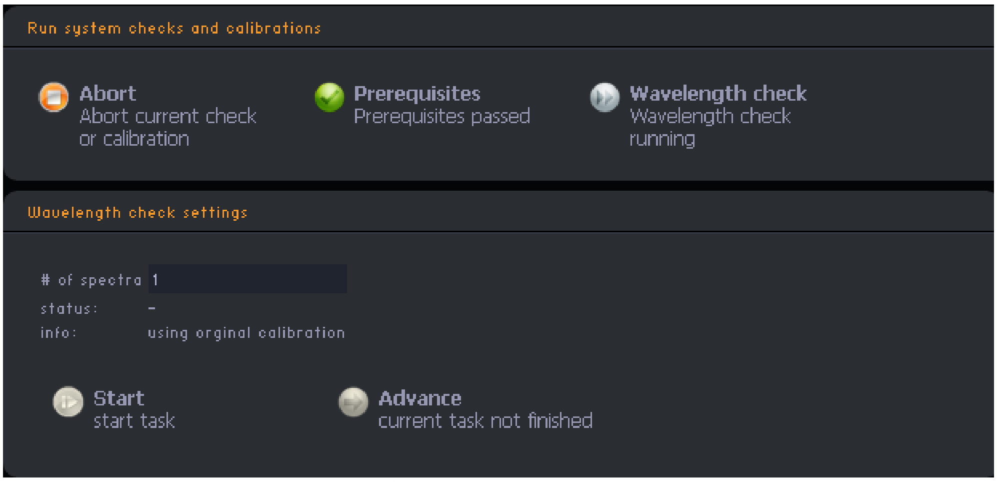


Fig. 25 The task control panel after starting the wavelength check

Result window

If the wavelength check passes the result window looks like Fig. 26:

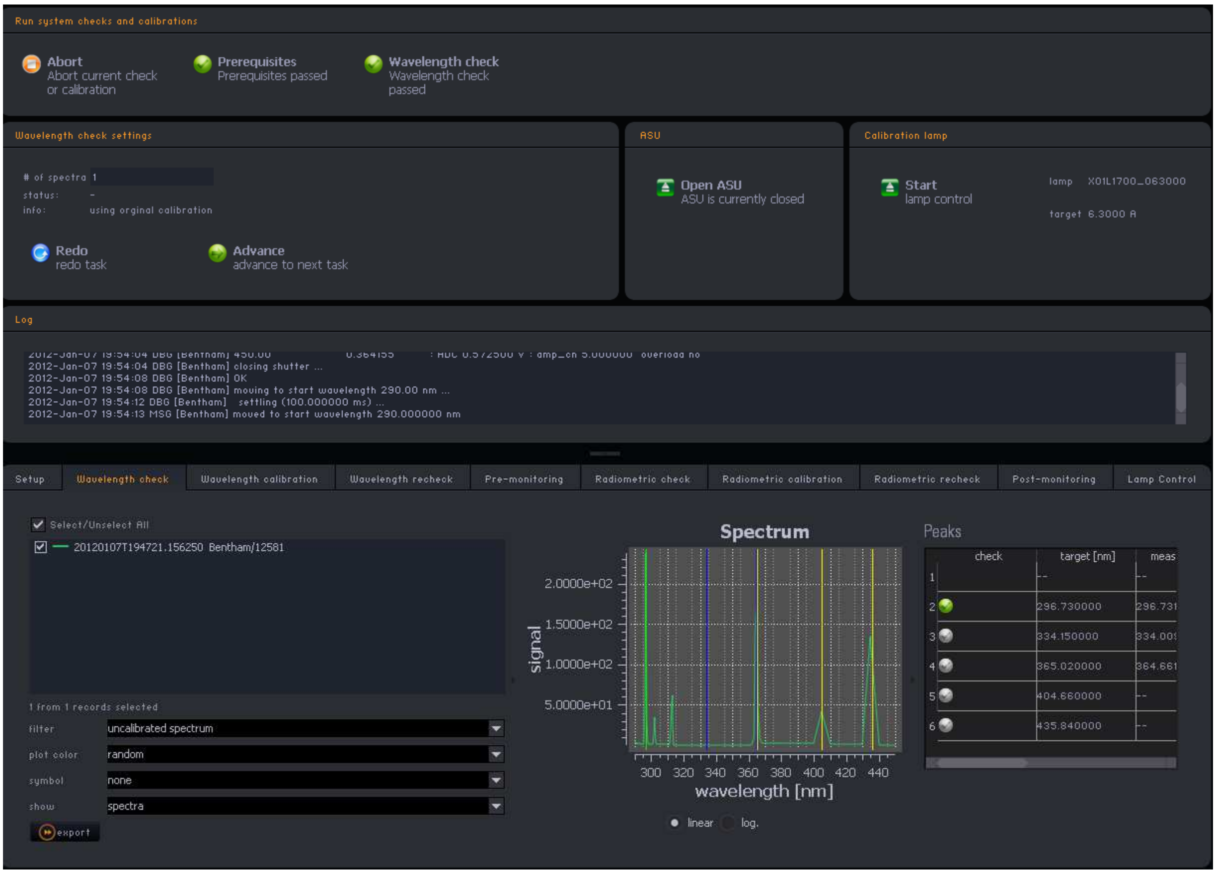


Fig. 26 The screen after the wavelength check step has been passed.

The result window might be zoomed for better inspection of the results. How this might look like is shown in Fig. 27:

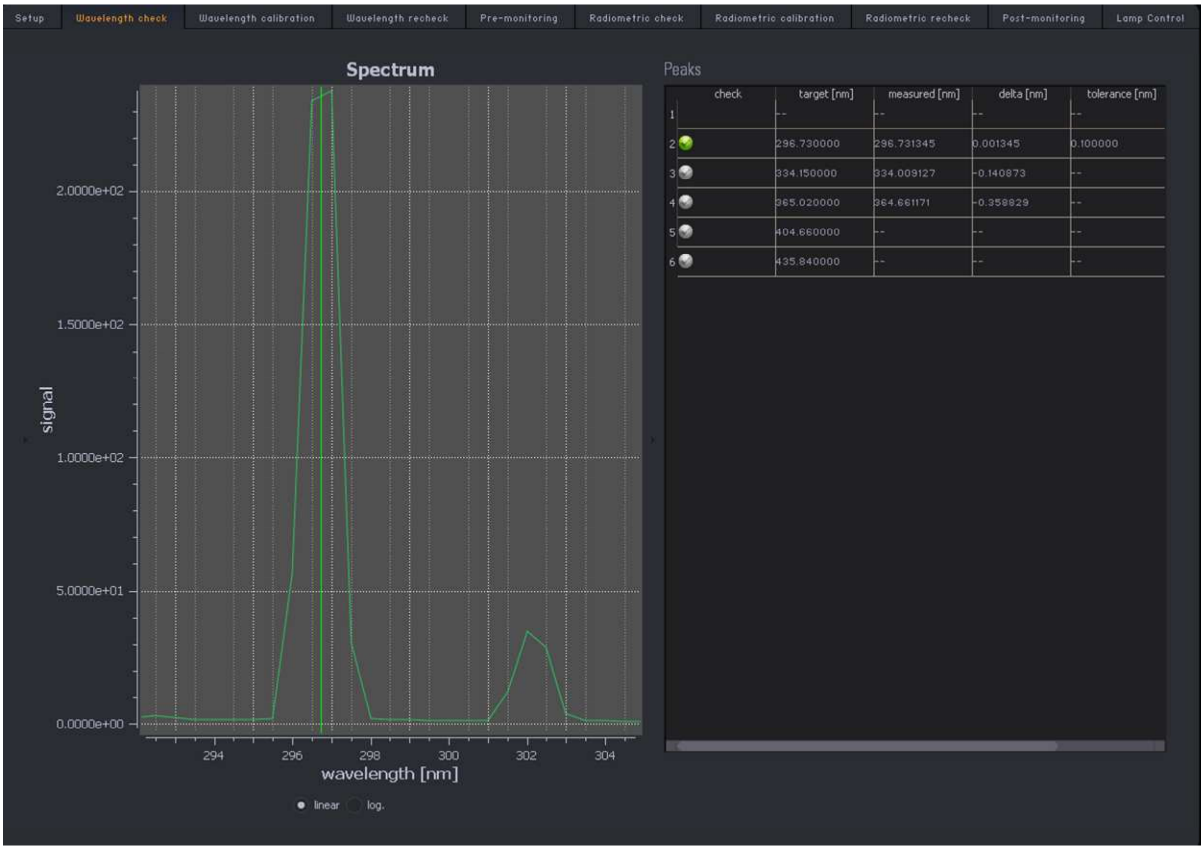


Fig. 27 The wavelength check result window in a zoomed state

If the wavelength check failes, the next step in the calibration flow is the wavelength calibration. If it finished with “passed”, the next step is powering down the HG lamp.

3.3.2.11 Wavelength calibration

Task description

During wavelength calibration the peak position of a single line of the HG lamp is measured using the full width half height method. From this the system derives a wavelength offset. For calibrating the wavelength scale this offset is later added to the nominal wavelength position when sending a command for wavelength positioning.

Starting the task

For starting the wavelength calibration press Advance and then Start in the task panel. The Start knob then gets white and the wavelength check starts (see Fig. 28):

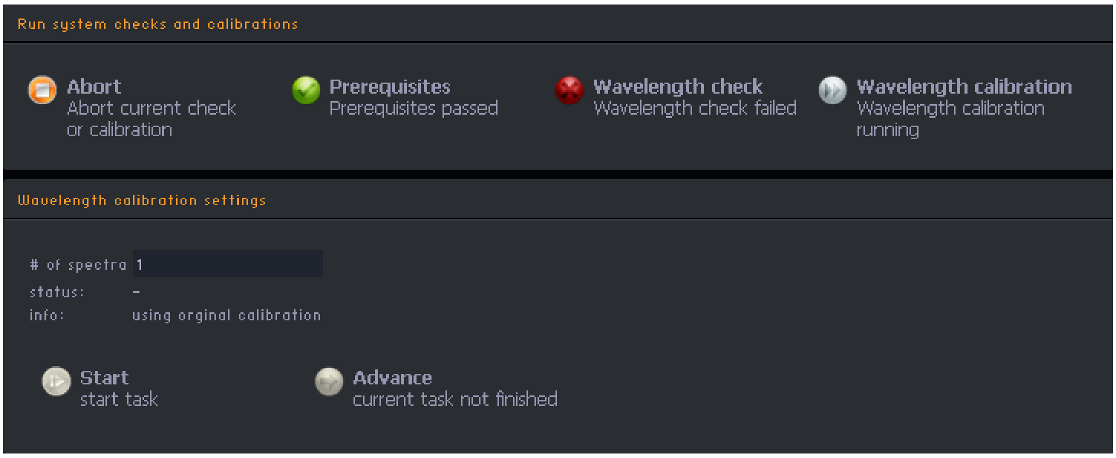


Fig. 28 The task control panel after starting the wavelength calibration

The result window of the wavelength calibration is very similar to the result window of the wavelength check (see Fig. 29 where a part of the result screen is shown):

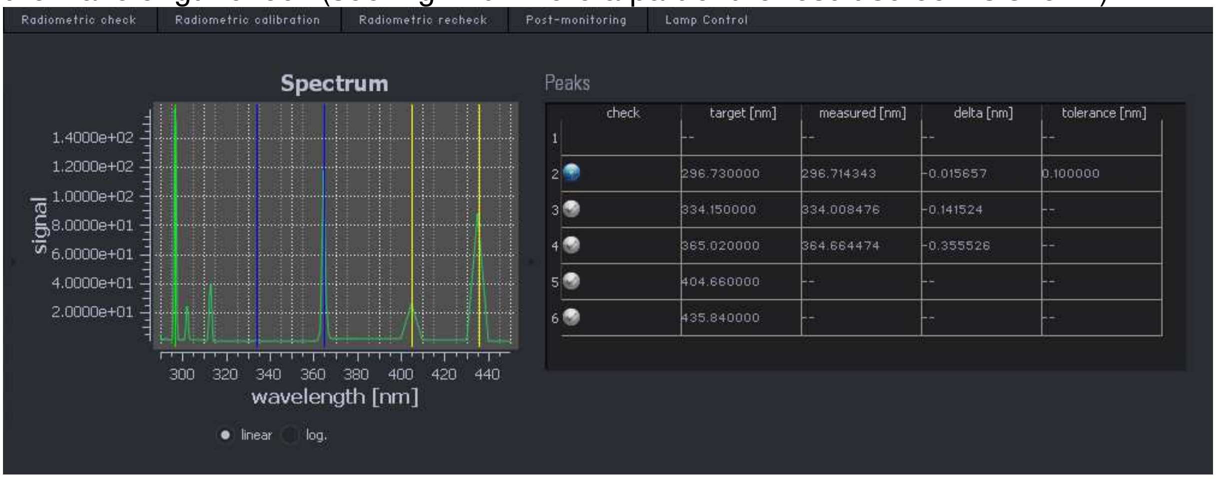


Fig. 29 Part of the result window of the wavelength calibration (zoomed state).

The table in the result window lists the target peak position, the measured peak position and the difference between target and measured position. The tolerance is only shown for completeness. The blue icon indicates that we deal with a step where a calibration is done and not a pass/fail check (which would be indicated with a green and red icon respectively). There are more wavelengths shown but these are not used in the calibration. These wavelengths are just measured for the protocol.

3.3.2.12 Wavelength recheck

Task description

The wavelength recheck does the same as the wavelength check but applying the wavelength offset measured in the step before. It proves that the wavelength offset calculated before leads to a valid wavelength calibration.

For starting the wavelength check press Advance and then Start in the task panel. The Start knob then gets white and the wavelength check starts (see Fig. 30):

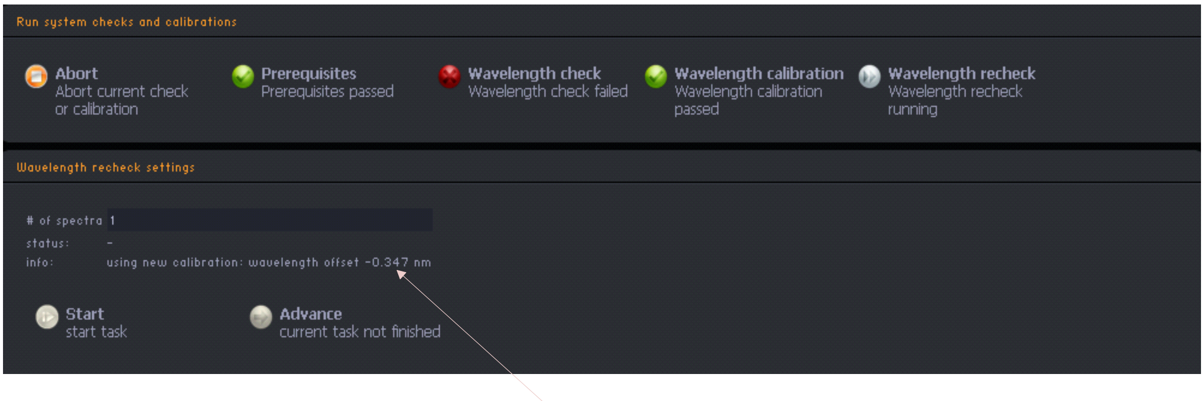


Fig. 30 The task control panel after starting the wavelength recheck

Result window

The result window of the wavelength recheck is identical to the result window of the wavelength check; see Fig. 26 and Fig. 27 for this.

3.3.2.13 Power down the HG lamp

This is a pure manual step without software support. The HG lamp must be powered down.

3.3.2.14 Starting and stopping the lamp control

Task description

The lamp control operates a calibration lamp and after start first drives up a current ramp until the current of the lamp reaches a desired value.

As long as the ramp is driven up (or down when stopping the lamp) the current and the voltage of the lamp are measured by the multimeter card implemented in the power supply.

At the point in time when the desired current value is reached, the measurement of voltage and current is done with the Keithley precision digital multimeter. It measures the voltage of the lamp and the voltage at a precision shunt for acquiring a value for the current. This current value is then continuously adjusted within narrow control limits by tuning the current of the power supply up or down.

Starting the task

Before starting the lamp control the lamp calibration file must be selected. This is done in the Setup panel (see Fig. 31):

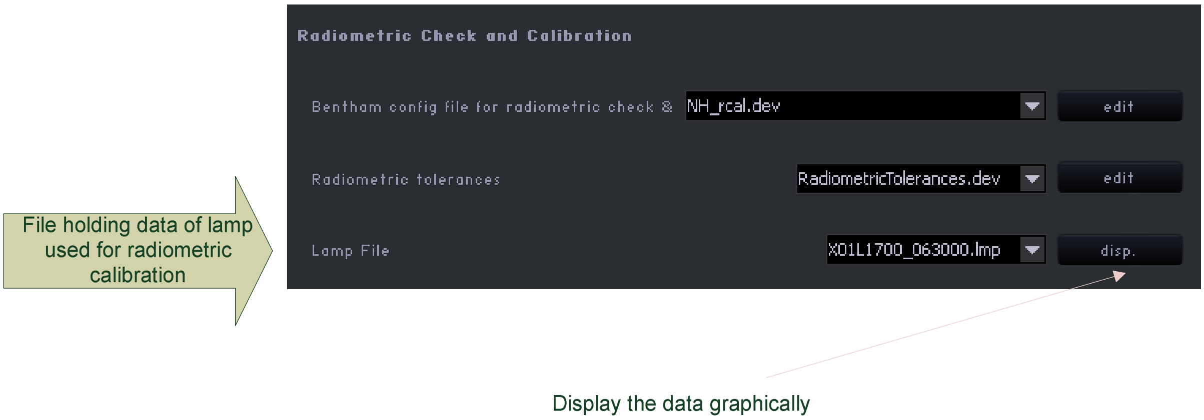


Fig. 31 The configuration file with the calibration data of the lamp. A lamp with a target current of 6.3000 A is selected.

Please note:
The target current of the calibration lamp is taken from the filename of the calibration lamp (see 3.3.3.7).

The lamp control can then be started (and stopped) any time by pressing the Start or Stop knob. Please note that the Start knob immediately turns into the Stop knob and vice versa.

The Start knob first brings up a dialog box for confirming the target current for operating the lamp (see Fig. 32):

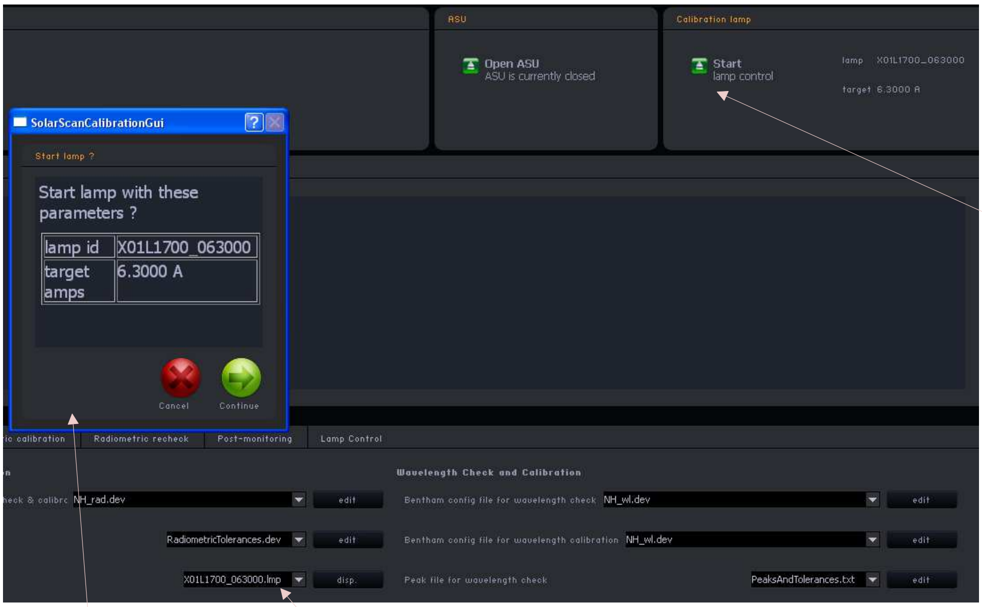


Fig. 32 The dialog box for confirming the target current to operate the lamp. Please note that the target current is derived from the filename holding the calibration data of the lamp.

After confirming the target current, the lamp control starts operating. It ramps up the lamp and finally achieves the state “adjusting”, see Fig. 33:

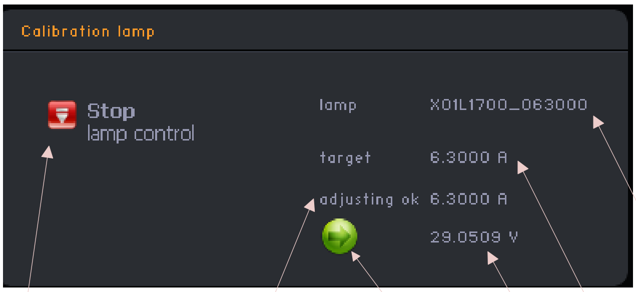


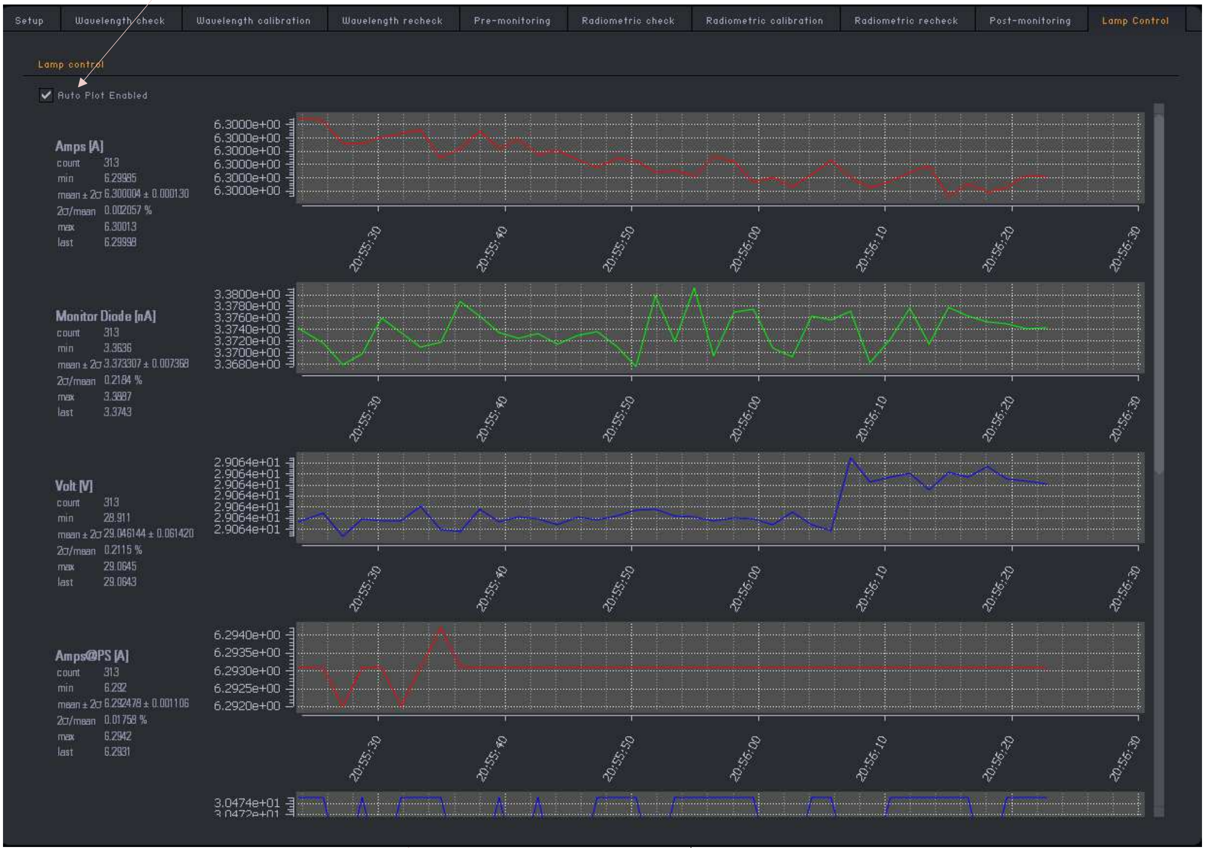
Fig. 33 The state of the lamp control is shown next to the Start/Stop button of the lamp control.

Result window

In the lamp control result window all parameters of the lamp control are plotted. The plots have two states: “Auto plot enabled/disabled”. If Auto plot is enabled (see Fig. 34) a live update of the plots is done. If Auto plot is disabled the full plots are shown and no live update happens (see Fig. 35).

The plots can be enlarged/shrinked using the Ctrl + keys and the Ctrl – keys respectively (first press Ctrl and then repeatedly one of the + or – keys).

Enable/disable Auto plot here



Use Strg + or Strg – to enlarge/shrink the plots

Fig. 34 The lamp control plots in state „Auto plot enabled“

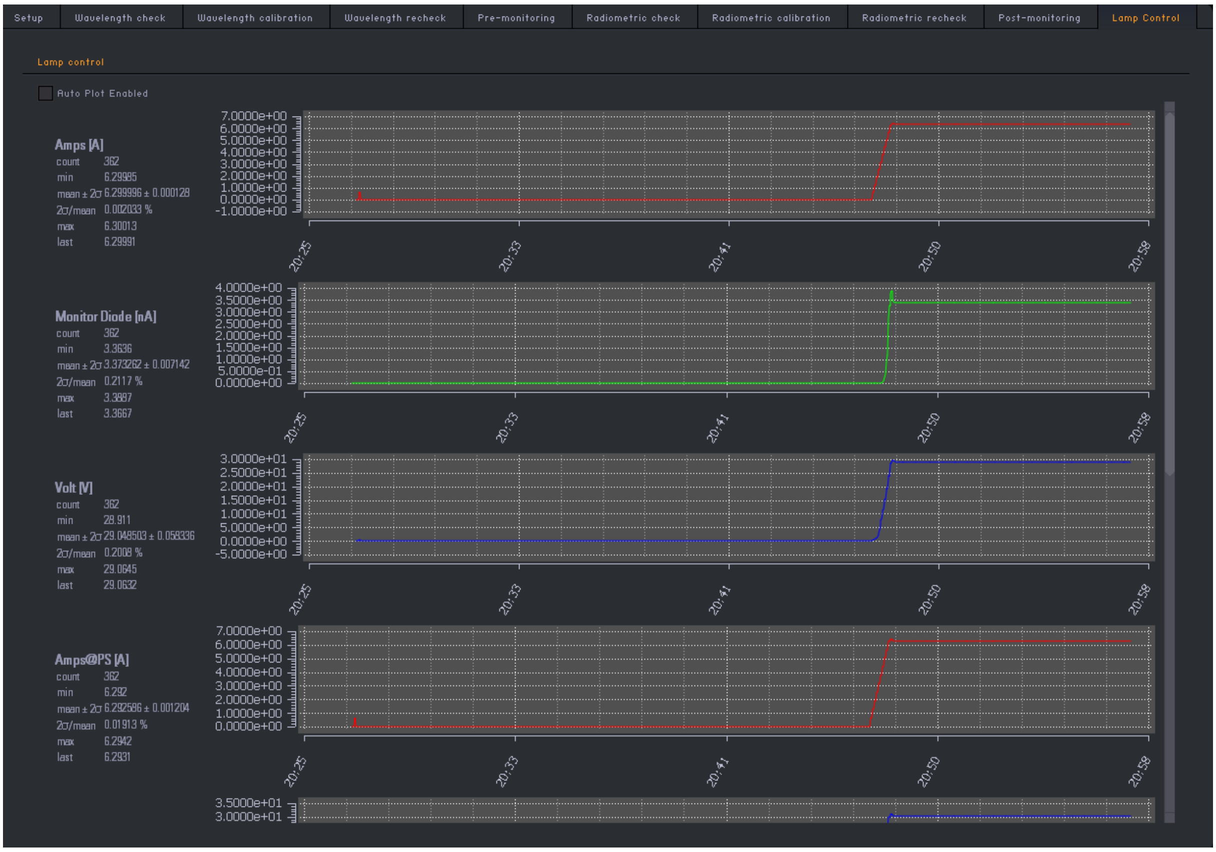


Fig. 35 The lamp control plots in state "Auto plot disabled" also showing the ramp up part

3.3.2.15 Premonitoring

Task description

The Premonitoring step does three things:

- Check the stability of the spectrometer at a constant wavelength
- Check the stability of the signal measured by the monitor diode
- Make sure the lamp is running for a minimum amount of time before checking the radiometric calibration

Starting the task

Premonitoring should be started after the lamp control state "adjusting" (see Fig. 33) is active.

For starting Premonitoring press Advance and then Start in the task panel. The Start knob then turns into the Abort knob (see Fig. 36):

Fig. 36 Premonitoring started – the Start knob has turned into the Abort knob

Result window

The Premonitoring result window shows two time series plots. In the upper part it shows the spectrometer signal at constant wavelength and in the lower part it shows the signal of the monitor diode. In addition, the measured standard deviation and the tolerance for the measured standard deviation are shown in the result table. The status of the Premonitoring is shown in the lower right corner (see Fig. 37):

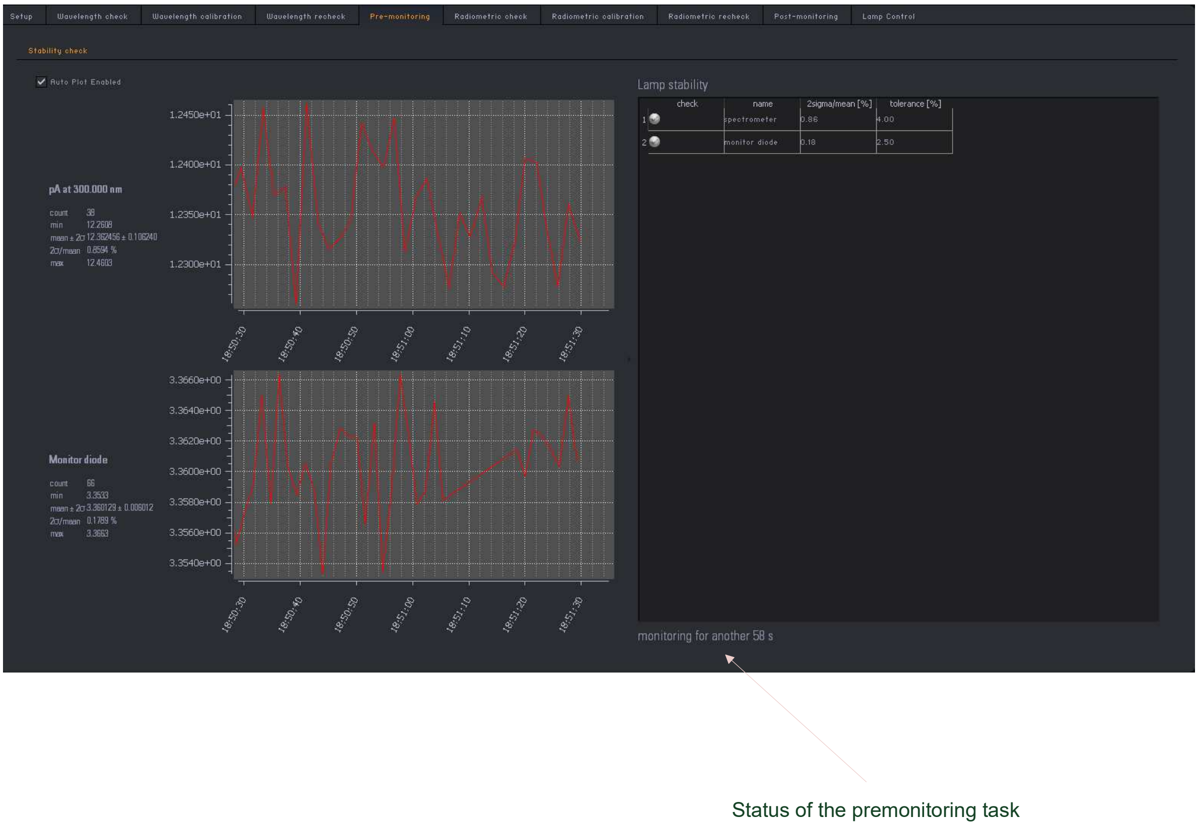


Fig. 37 The Premonitoring result window in the state "Monitoring". The current status of the premonitoring step is shown in the lower right corner

As soon as the required time for settling the lamp has elapsed and both signals (spectrometer and monitor diode) are stable (i.e. the standard deviation is within the tolerance) the two icons in the result table get green and the status of the monitoring switches to “Lamp is stable” (see Fig. 38):

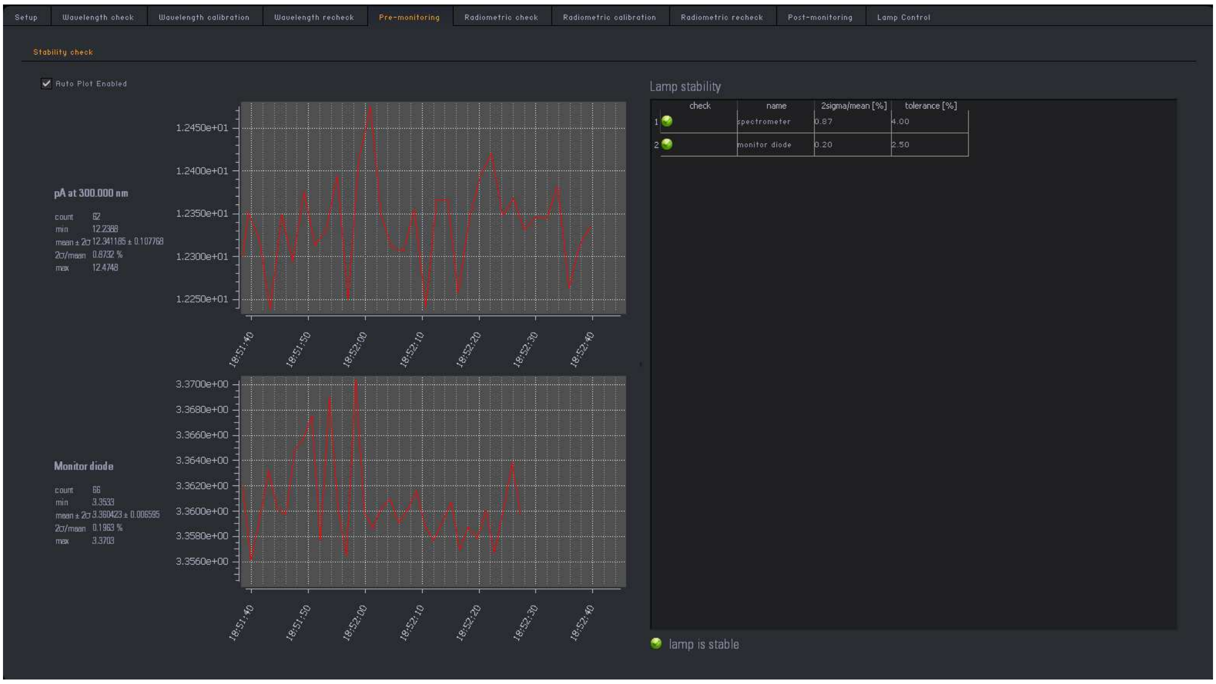


Fig. 38 The Premonitoring result window in the state “lamp is stable”

3.3.2.16 Radiometric check

Task description

The radiometric check measures the calibration lamp using the old wavelength calibration. The result is compared to the data of the calibration lamp. If the ratio of the two measurements is within predefined limits at the constant wavelength and in the UV-B and UV-A the check passes, otherwise it fails.

Starting the task

For starting the radiometric check press Advance and then Start in the task panel. The Start knob then turns into white (see Fig. 39):

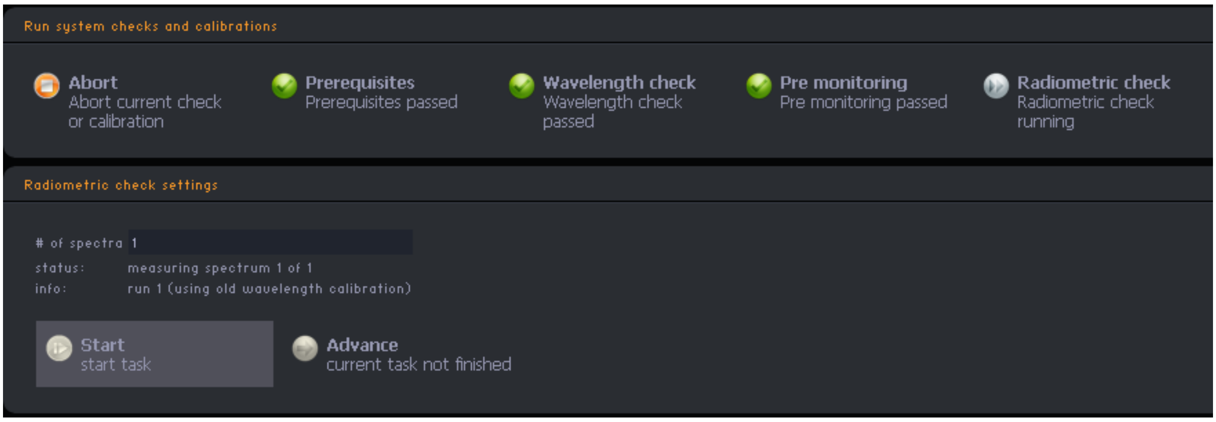


Fig. 39 The task panel after the Radiometric check has started

Result window

The result window of the radiometric check shows the calibrated spectrum and a table where the deviation of this measured spectrum from the nominal spectrum of the calibration lamp is shown. The deviation is shown for the constant wavelength, the UVB and the UVA (see Fig. 40):

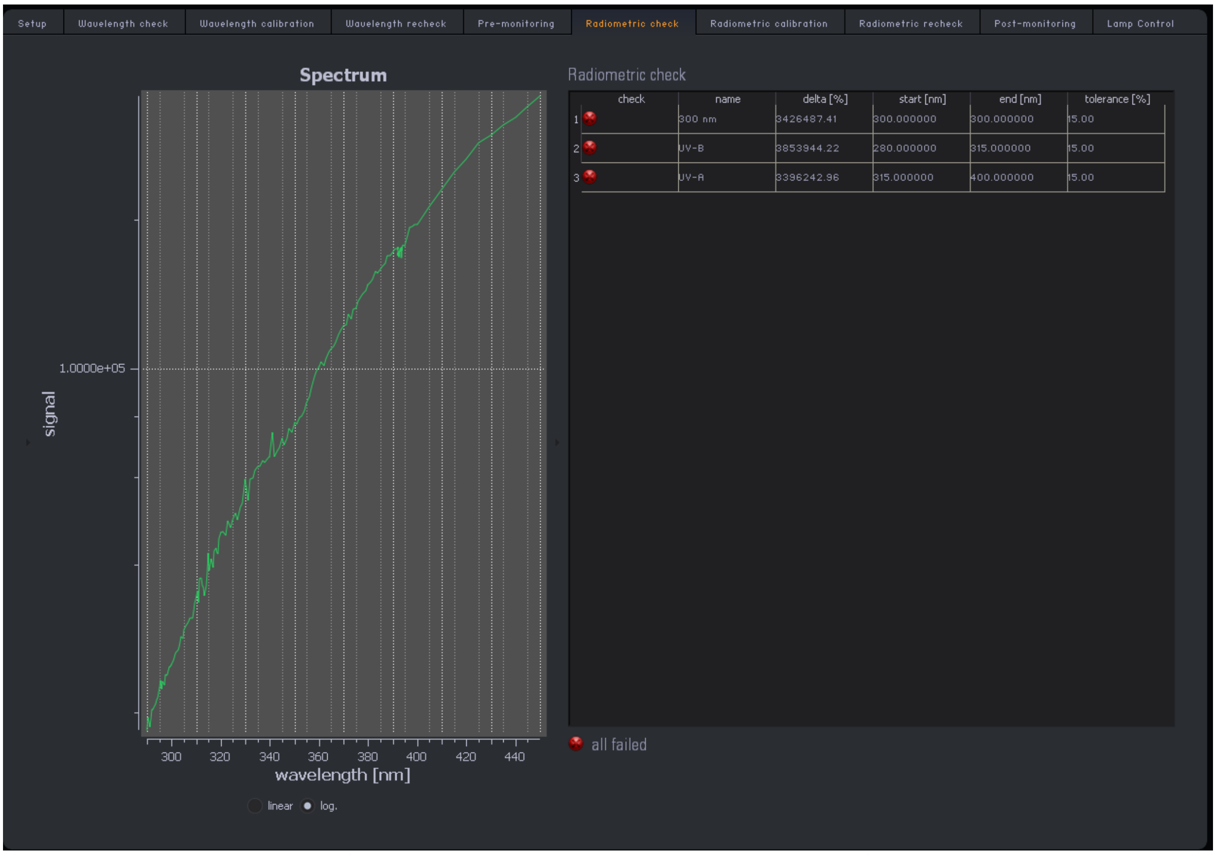


Fig. 40 The result window of the radiometric check as it could look like if the check fails. The example shows the window in a zoomed state

3.3.2.17 Radiometric calibration

Task description

The radiometric calibration measures an uncalibrated spectrum of the calibration lamp. In case a new wavelength calibration has been generated before, it uses this new wavelength calibration.

Starting the task

For starting the radiometric calibration press Advance and then Start in the task panel. The Start knob then turns into white (see Fig. 41):

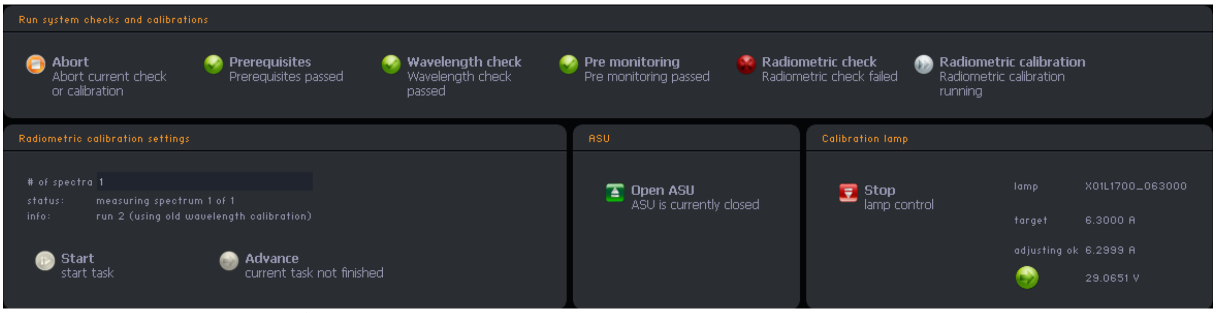


Fig. 41 The task panel after the Radiometric calibration has started

Result window

The result window of the radiometric calibration just shows the uncalibrated spectrum. It is the only result window without a tolerance table (see Fig. 42):

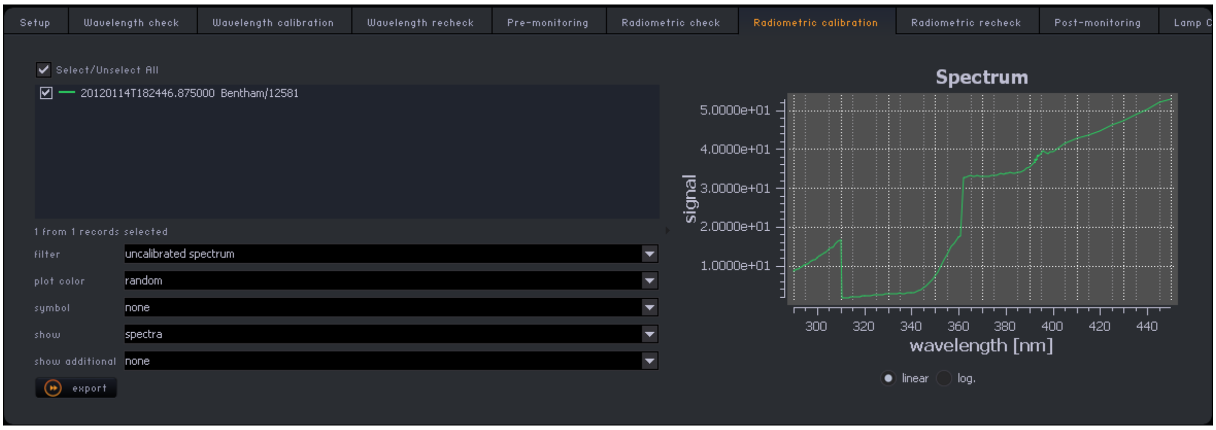


Fig. 42 The result window of the radiometric calibration showing an uncalibrated spectrum of the calibration lamp

3.3.2.18 Radiometric recheck

Task description

The radiometric recheck works much like the radiometric check. It measures the calibration lamp and compares the result to the data of the calibration lamp. If the ratio of the two measurements is within predefined limits at 300 nm and in the UV-B and UV-A, the check passes, otherwise it fails.

In contrast to the radiometric check, the radiometric recheck uses a new wavelength calibration and the radiometric calibration generated before.

Starting the task

For starting the radiometric recheck press Advance and then Start in the task panel. The Start knob then turns into white (see Fig. 43):

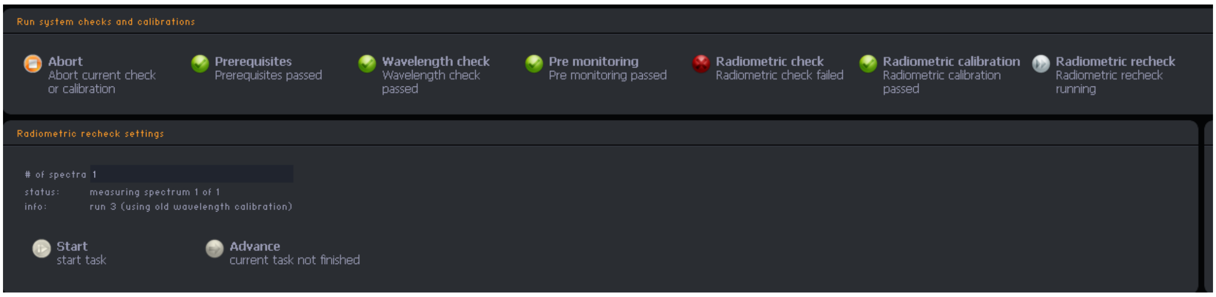
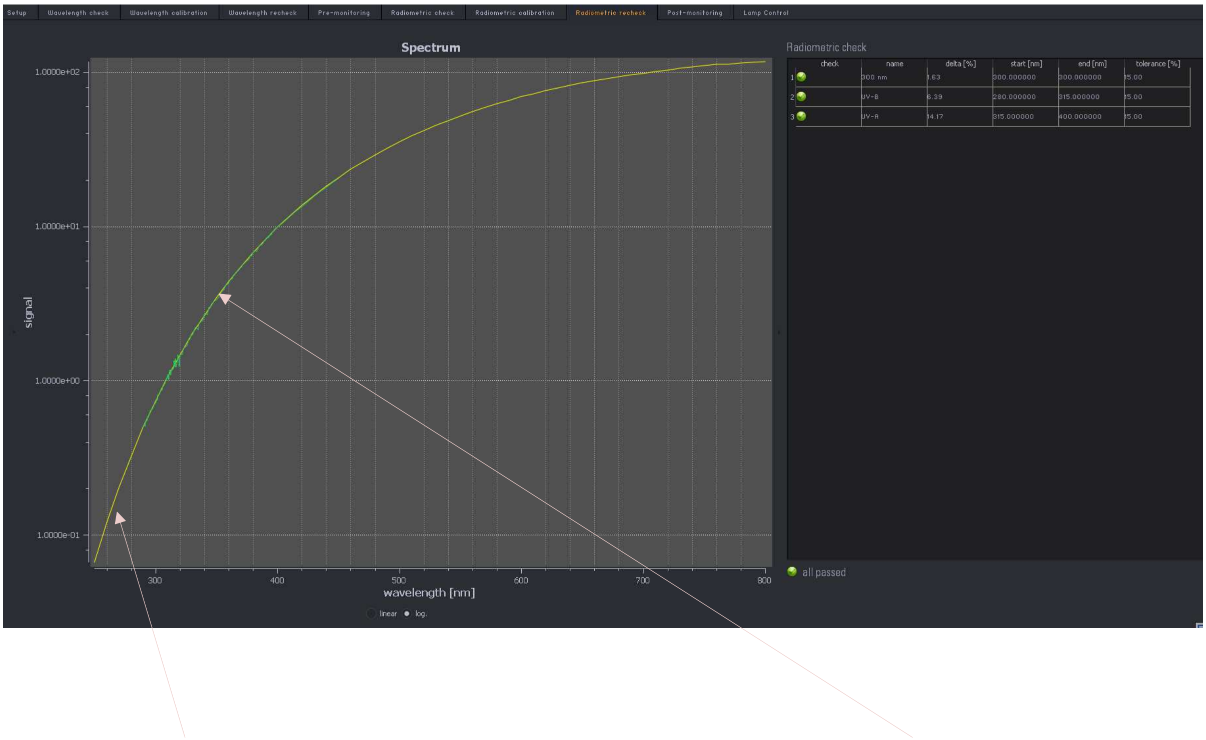


Fig. 43 The task panel after the Radiometric recheck has started

Result window

Similar to the result window of the radiometric check the result window of the radiometric recheck shows the calibrated spectrum and a table where the deviation of this measured spectrum from the nominal spectrum of the calibration lamp is shown. The deviation is shown for the constant wavelength, the UV-B and the UV-A (see Fig. 44):



Nominal data of the calibration lamp

Measured calibration lamp

Fig. 44 The result window of the radiometric recheck showing the measured lamp data and the nominal lamp data. In the case shown the nominal lamp data (yellow curve) cover a wider wavelength region than the measured lamp data (green curve).

3.3.2.19 Postmonitoring

Task description

Postmonitoring is done to prove that the calibration lamp is still in a stable state after having done the steps for the radiometric check and calibration and recheck.

It does:

- Check the stability of the spectrometer at a constant wavelength
- Check the stability of the signal measured by the monitor diode

Note that there is a separate configuration file for the Postmonitoring step because in contrast to the Premonitoring, the Postmonitoring typically is configured for observing just a time span of 2 min. This is enough, because the calibration lamp has been burning for quite a while at this point in the calibration flow.

Starting the task

Postmonitoring must be started after either a successful radiometric check has been done or a successful radiometric calibration and recheck has been done. For starting press the Advance and Start button in the task panel (see Fig. 45). The button will then turn into the Abort button.

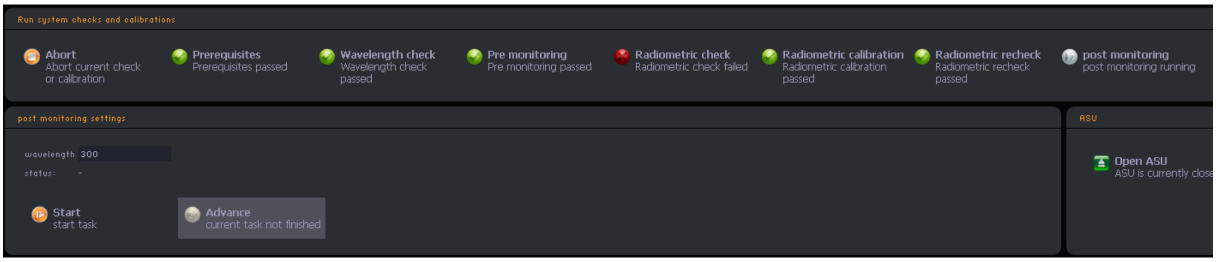


Fig. 45 The task panel before Postmonitoring has been started. Press Start for starting it. The Start button the turns into the Abort button.

Result window

The Postmonitoring result window shows two time series plots. In the upper part it shows the spectrometer signal at constant wavelength and in the lower part it shows the signal of the monitor diode. In addition, the measured standard deviation and the tolerance for the measured standard deviation are shown in the result table. The status of the Postmonitoring is shown in the lower right corner (see Fig. 46):

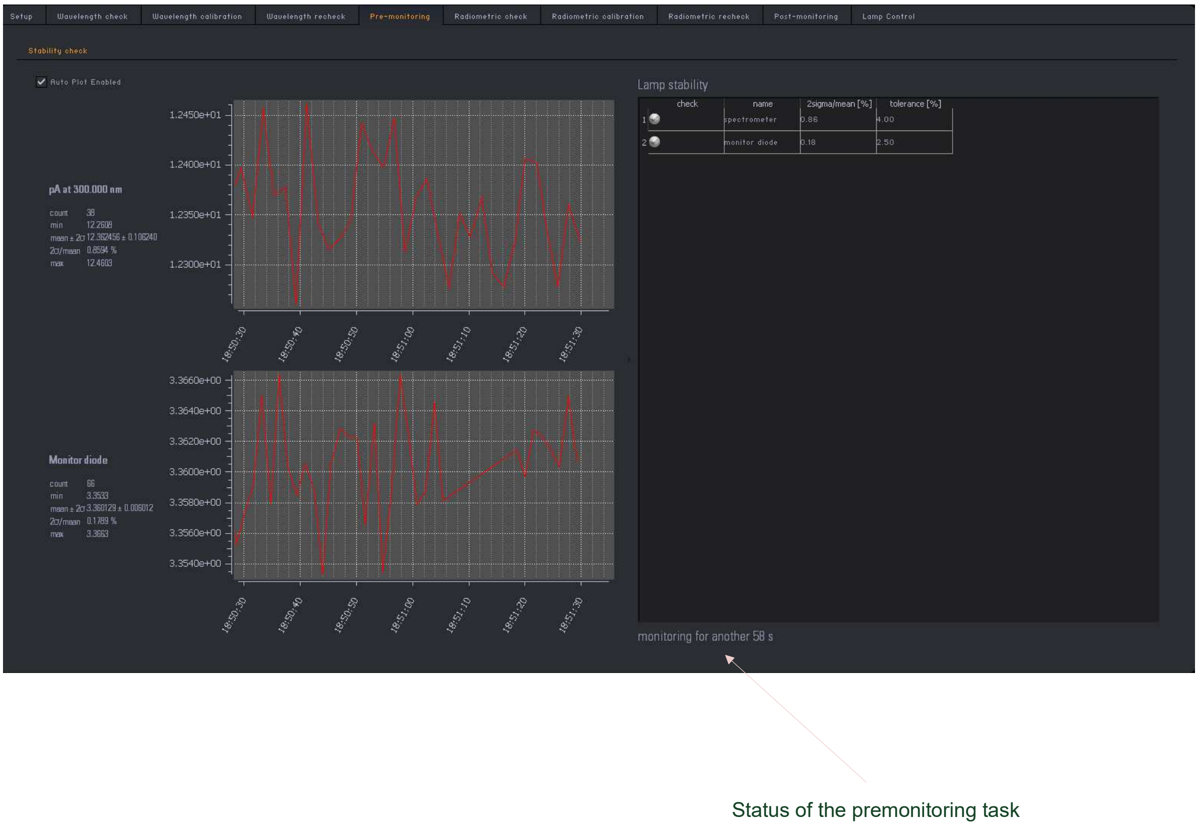


Fig. 46 The Postmonitoring result window in the state "Monitoring"

As soon as the required time for settling the lamp has elapsed and both signals (spectrometer and monitor diode) are stable (i.e. the standard deviation is within the desired tolerance) the two "LED" icons in the result table get green and the status of the monitoring switches to "lamp is stable" (see Fig. 47):

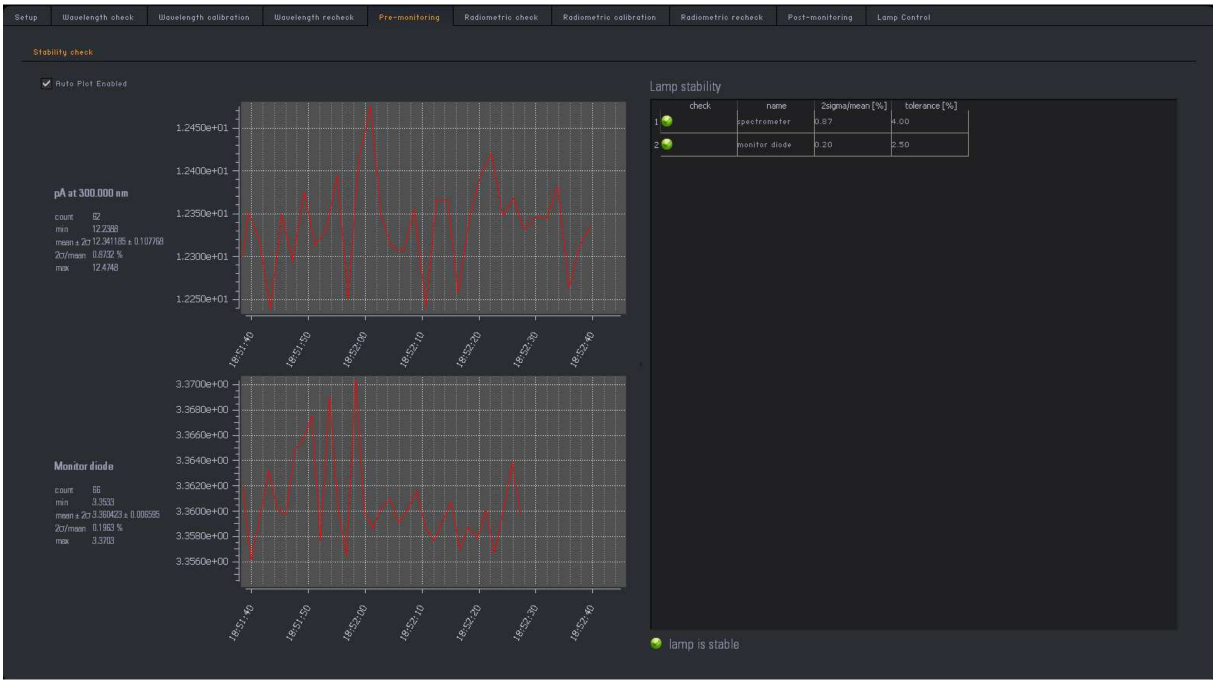


Fig. 47 The Premonitoring result window in the state "lamp is stable"

3.3.2.20 The calibration report

After the Postmonitoring has successfully finished and Advance has been pressed for the last time, the system brings up a report window (see Fig. 48 and Fig. 49):

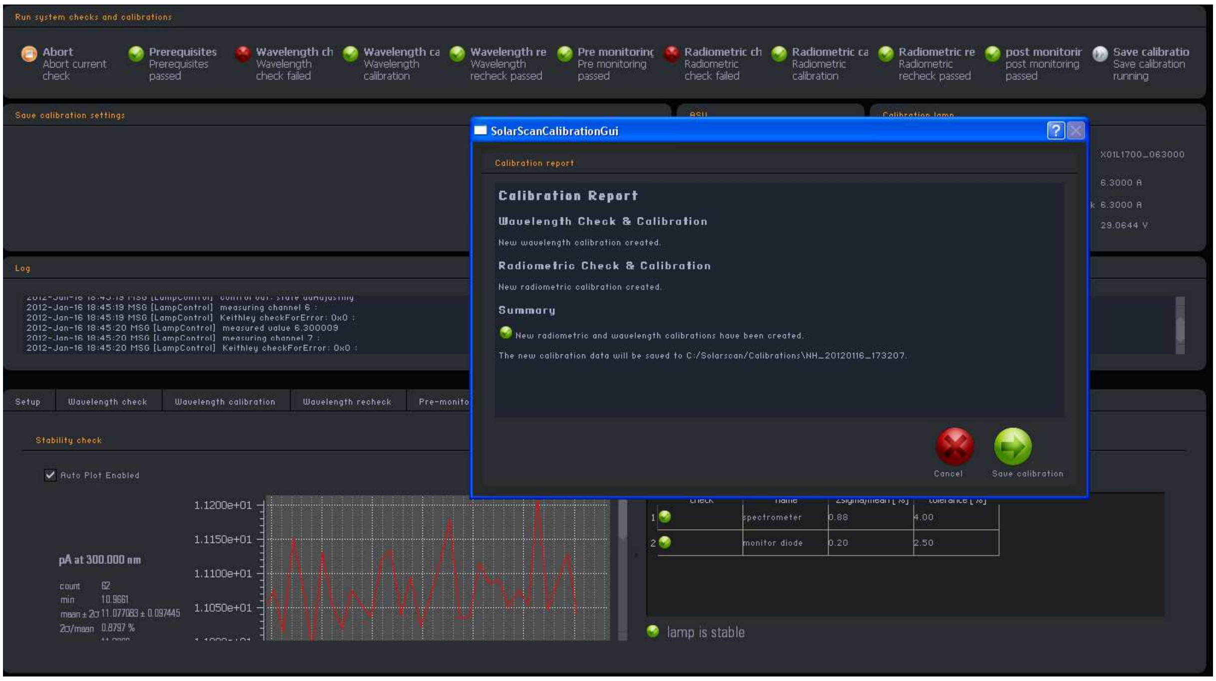


Fig. 48 The calibration report window is shown after the complete calibration flow has been successfully done. The example shows the maximum number of steps in the calibration flow.

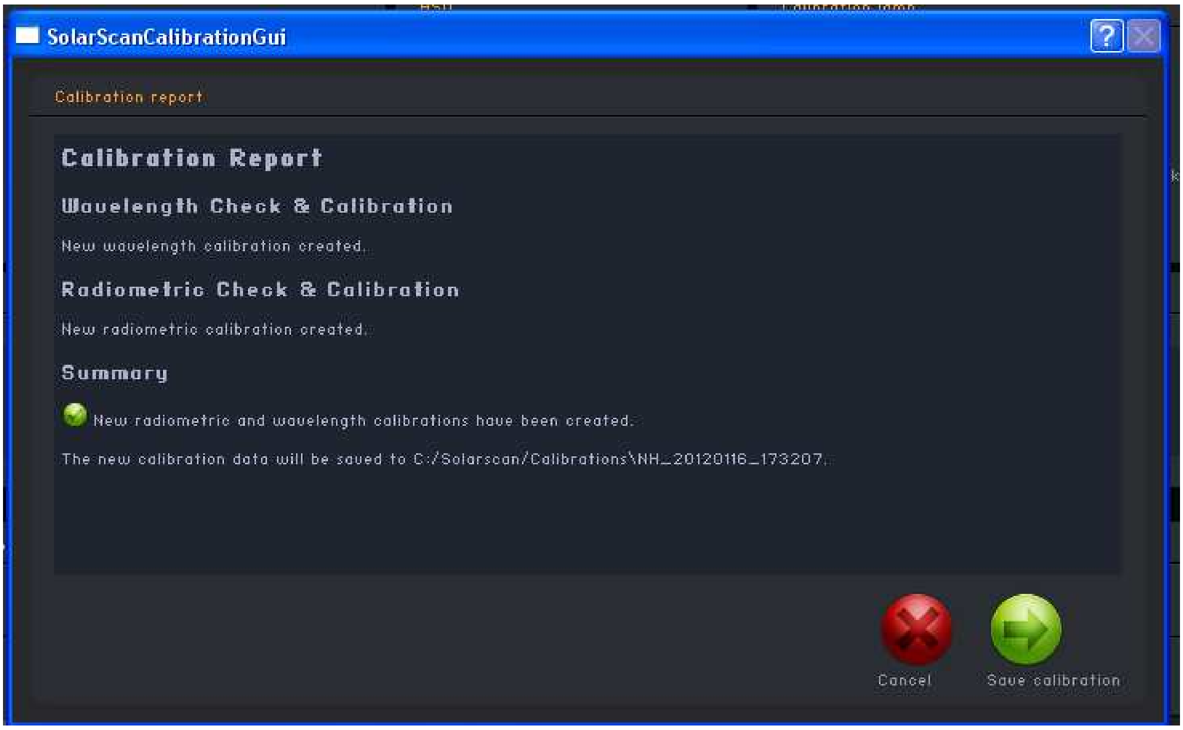


Fig. 49 The calibration report window after successfully calibrating the system for wavelength and radiometric accuracy. It also shows the folder where the calibration is written to.

After successful saving the calibration, a final window comes up showing the success of saving and activating the results of the calibration run (see Fig. 50):

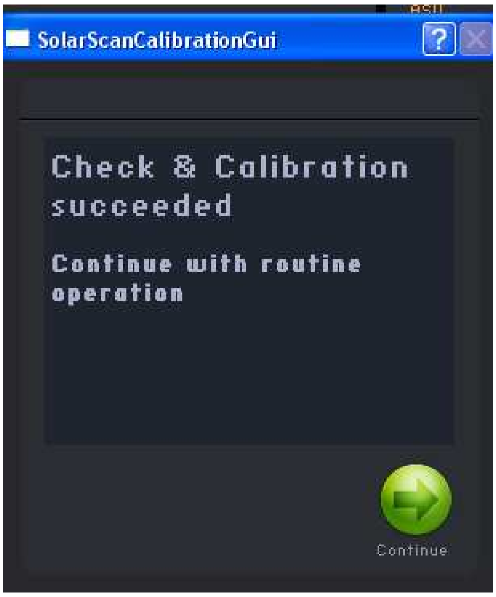


Fig. 50 The window showing the success of writing the calibration to disk

The spectrometer is calibrated now.

With the Continue button the SolarscanCalibrationGUI deletes the calibration flow entries.

3.3.2.21 Stop lamp control

The next step to do is to drive down the calibration lamp. For this press the Stop button in the lamp control panel (see Fig. 51) and wait until the calibration lamp is in state idle again (see Fig. 52).

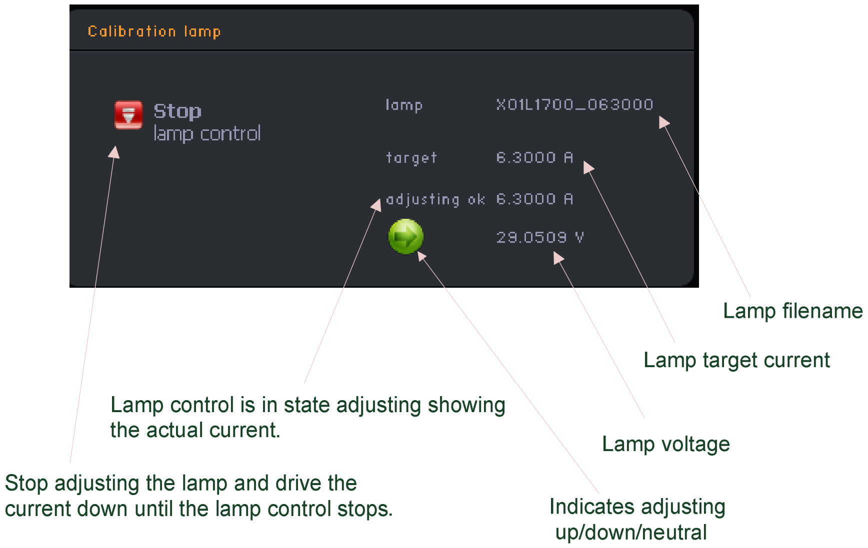


Fig. 51 Press Stop (red button) in the lamp control window for driving the lamp control down. Wait until the lamp control is in state ídle.

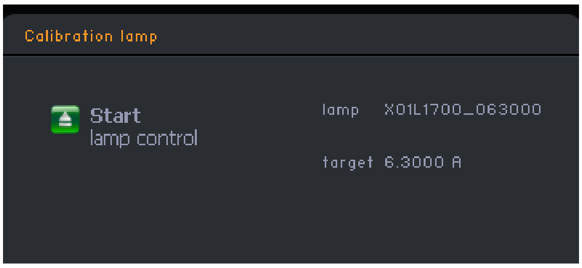


Fig. 52 Lamp control in state idle. Only the calibration lamp filename and the lamp target current are shown.

3.3.2.22 Stop the SolarscanCalibration GUI

For this, just close the window of the SolarscanCalibrationGUI application.

3.3.2.23 Rearrange hardware setup for normal operation

Remove the calibration hardware.

3.3.2.24 Restart the Scheduler

Start the start_scheduler.bat file.

3.3.3 Configuration files

All configuration files are listed in the Setup panel (see Fig. 53):

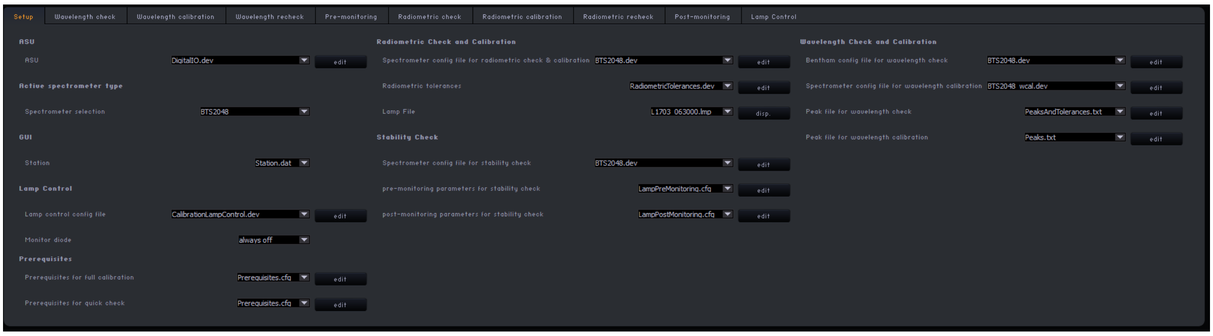


Fig. 53 The setup panel manages entries for all configuration files used in the SolarscanCalibration-GUI. The content of most of the files can be inspected by using the disp buttons right to the entries.

3.3.3.1 ASU

This entry is for defining a parameter file for a digital IO controlling the so called motorized “ASU cover”. The digital control is a separate instrument connected via USB.

3.3.3.2 Active spectrometer type

The entry “Spectrometer selection” must be set to BTS2048 when using this device as the spectrometer. When using the spectrometers manufactured by BENTHAM either “Bentham via Bentham DLL” or “Bentham via AUTOSCAN DLL” must be selected.

3.3.3.3 Prerequisites

The configuration file for the Prerequisites box is listed in the Prerequisites section (see Fig. 54):

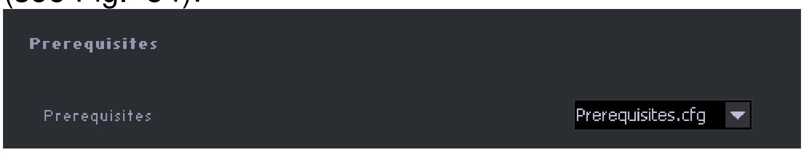


Fig. 54 The configuration file for the prerequisites box is listed under “Prerequisites”

3.3.3.4 Wavelength check

The task file for the wavelength check is listed in the Wavelength Check and Calibration section (see Fig. 55):

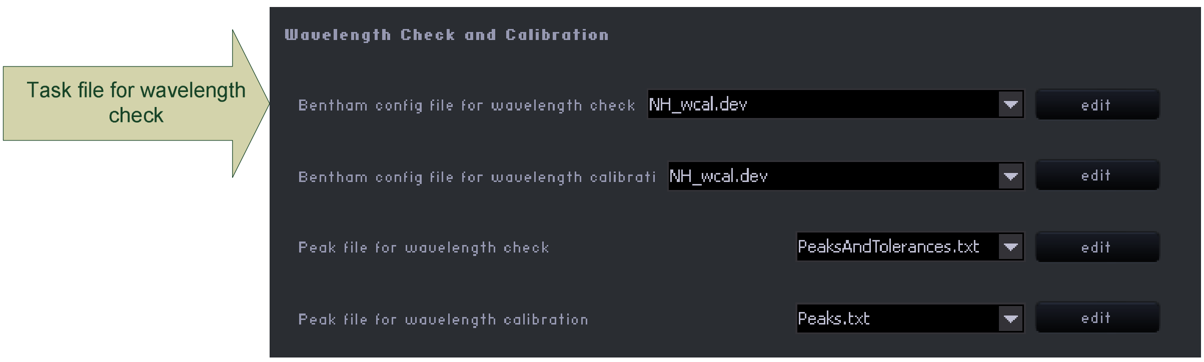


Fig. 55 The task file for configuring the Spectrometer for wavelength checking is listed under “Wavelength Check and Calibration”.

The configuration file for the peaks and tolerances of the peak positions to be measured is also listed in the Wavelength Check and Calibration section (see Fig. 56):

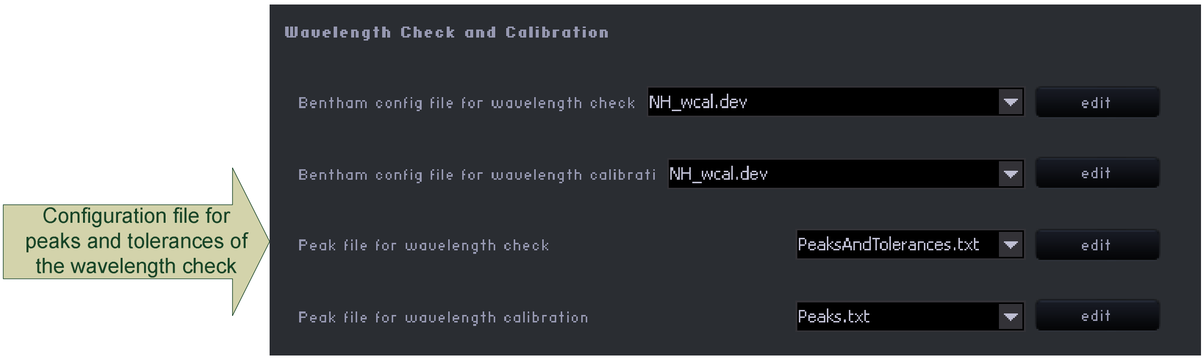


Fig. 56 The configuration file for defining the peak positions and tolerances when checking the wavelength is listed under “Wavelength Check and Calibration” in the setup panel.

3.3.3.5 Wavelength calibration

The configuration file for wavelength calibration is listed in the Wavelength check and Calibration section (see Fig. 57):

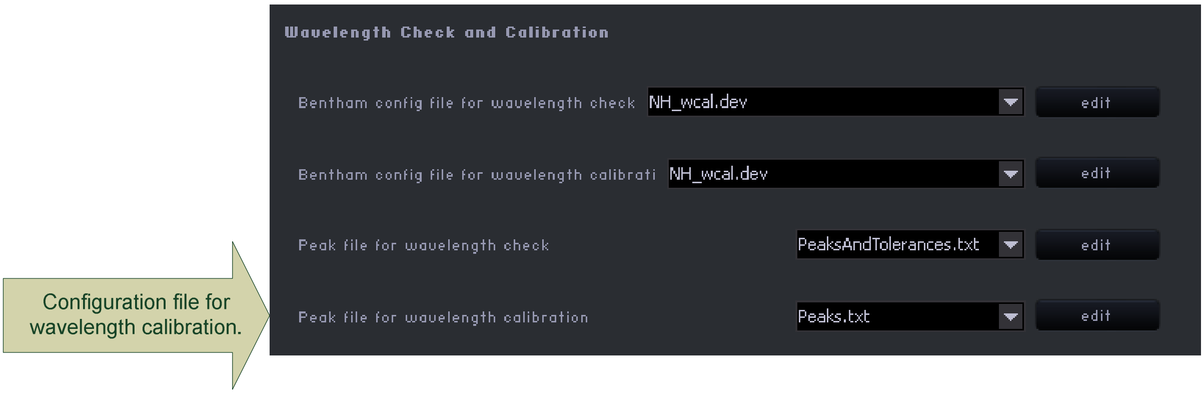


Fig. 57 The configuration file for wavelength calibration is listed in the Wavelength check and Calibration section

3.3.3.6 Wavelength recheck

The wavelength recheck uses the same configuration file as the wavelength check - see 3.3.3.4.

3.3.3.7 Starting and stopping the lamp control

The filename of the task file for lamp control is listed under “Lamp Control” (see Fig. 58):

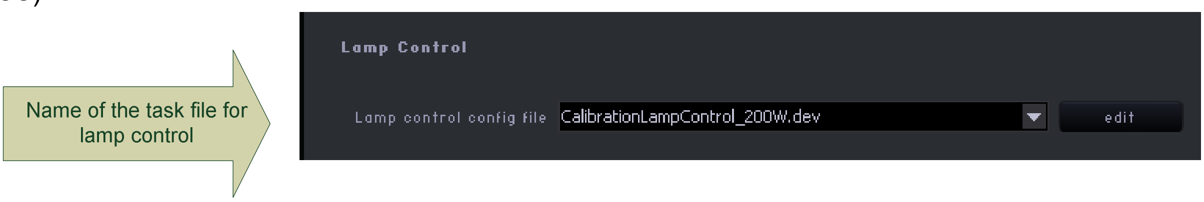


Fig. 58 The filename of the task file for lamp control is listed under “Lamp Control“

The filename holding the calibration data of the lamp used for radiometric calibration is also listed under “Lamp Control” (see Fig. 59):

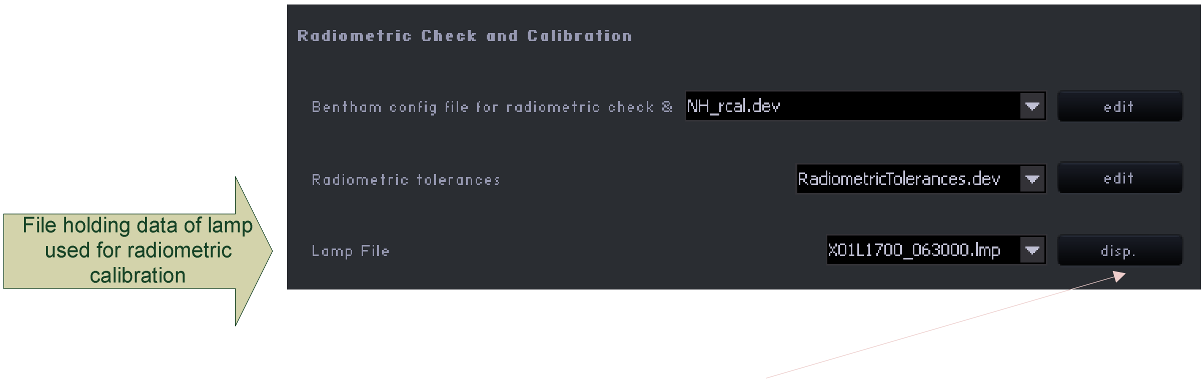


Fig. 59 The configuration file with the calibration data of the lamp.

For security reasons the target current for the lamp control is part of the name of the file holding the calibration data of the lamp.

The filename holding the calibration data of the lamp must follow the rule: <name>_######.lmp. I.e. <name>_063000 stands for target current of 6.3 amps.

Please note that only filenames according to the naming convention <name>_######.lmp are listed.

The data can be displayed graphically by using the disp button right to the filename (see Fig. 60):

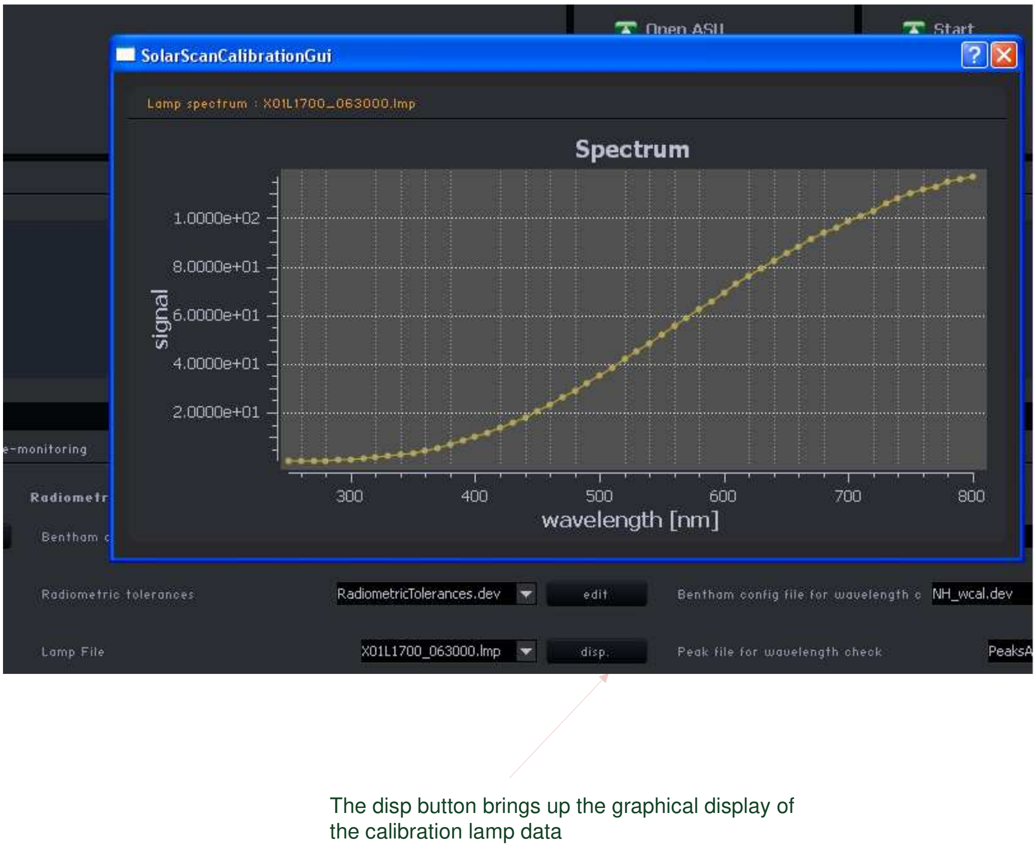


Fig. 60 The graphical display of the lamp calibration data is initiated by pressing the disp button right to the “Lamp File” list box.

3.3.3.8 Premonitoring

The spectrometer task file for Pre and Postmonitoring is listed under “Stability Check” in the Setup panel (see Fig. 61):

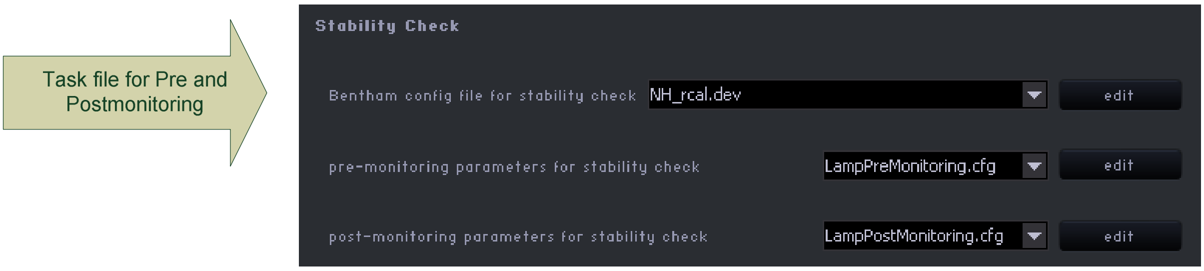


Fig. 61 The spectrometer task file for pre and postmonitoring is listed under “Stability check”.

The configuration file for Premonitoring is also listed under “Stability check” (see Fig. 62):

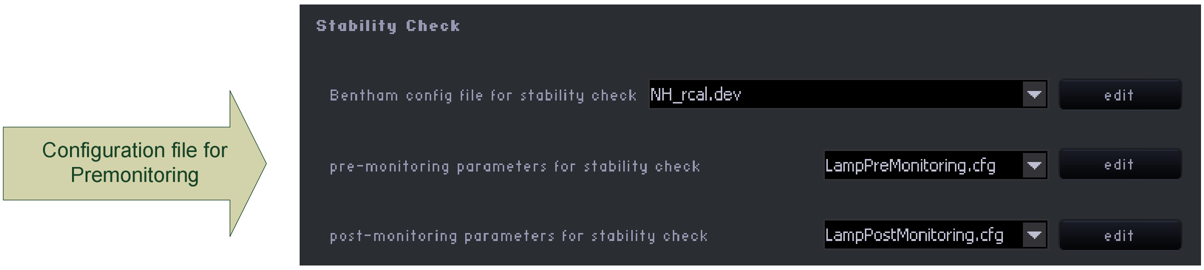


Fig. 62 The configuration file for Premonitoring is listed under “Stability check”

3.3.3.9 Radiometric check

The spectrometer task file for Radiometric check is listed under “Radiometric check and Calibration” in the Setup panel (see Fig. 63). The same file is used for the radiometric calibration and the radiometric recheck.

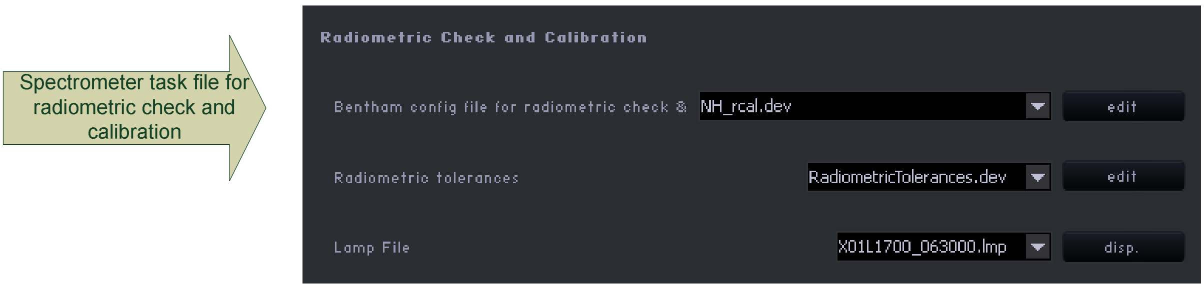


Fig. 63 The spectrometer task file for radiometric check and calibration

The configuration file for checking the deviation of the measured and nominal lamp data is also listed under “Radiometric check and calibration” (see Fig. 64):

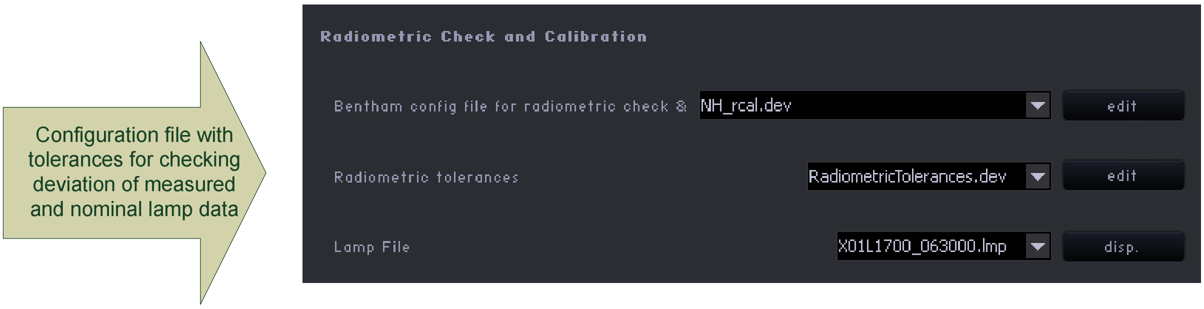


Fig. 64 The configuration file for checking deviation of measured and nominal lamp data.

3.3.3.10 Radiometric calibration

The only relevant configuration file is the task file for the spectrometer. It is listed under "Radiometric Check and Calibration" in the Setup panel (see Fig. 65). The task file is the same as it is used for radiometric check and radiometric recheck.

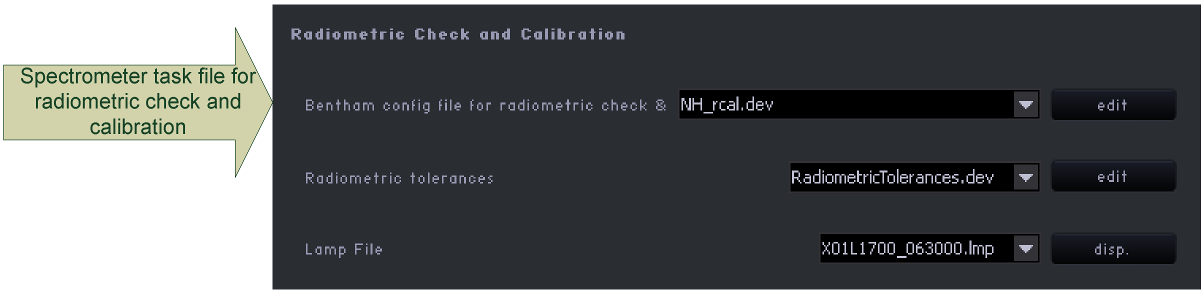


Fig. 65 The spectrometer task file for radiometric check and calibration

3.3.3.11 Radiometric recheck

The spectrometer task file for Radiometric recheck is listed under "Radiometric check and Calibration" (see Fig. 66):

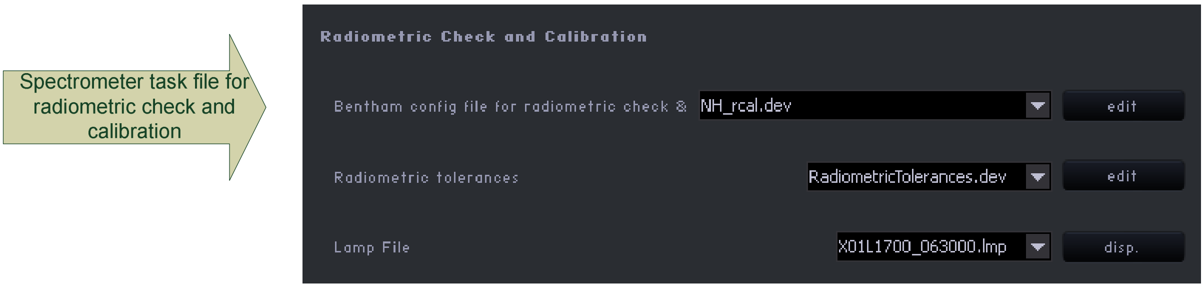


Fig. 66 The spectrometer task file for radiometric check and calibration

The configuration file for checking the deviation of the measured and nominal lamp data is also listed under "Radiometric check and calibration" (see Fig. 67):

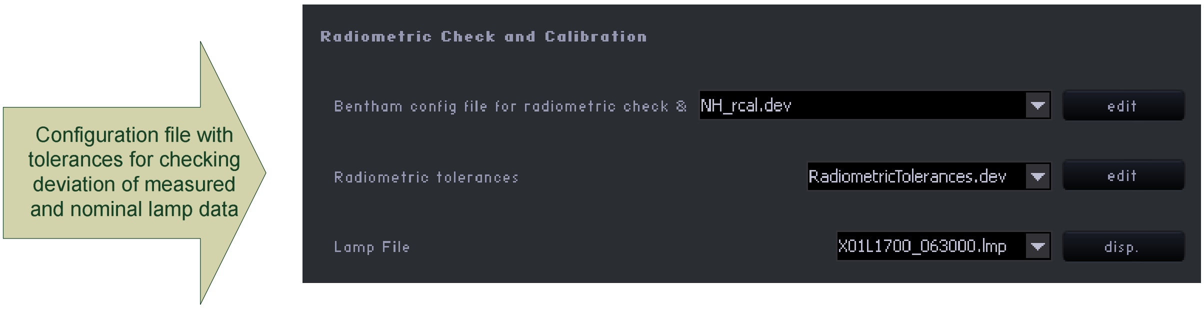


Fig. 67 The configuration file for checking deviation of measured and nominal lamp data in the Setup panel.

3.3.3.12 Postmonitoring

The spectrometer task file for Pre and Postmonitoring is listed under "Stability Check" (see Fig. 68):

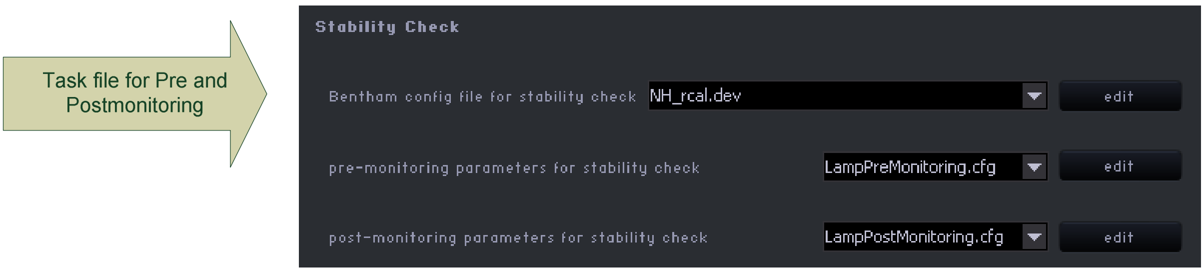


Fig. 68 The spectrometer task file for pre and postmonitoring is listed under "Stability check".

The configuration file for Postmonitoring also is listed under "Stability check" in the Setup panel (see Fig. 69):

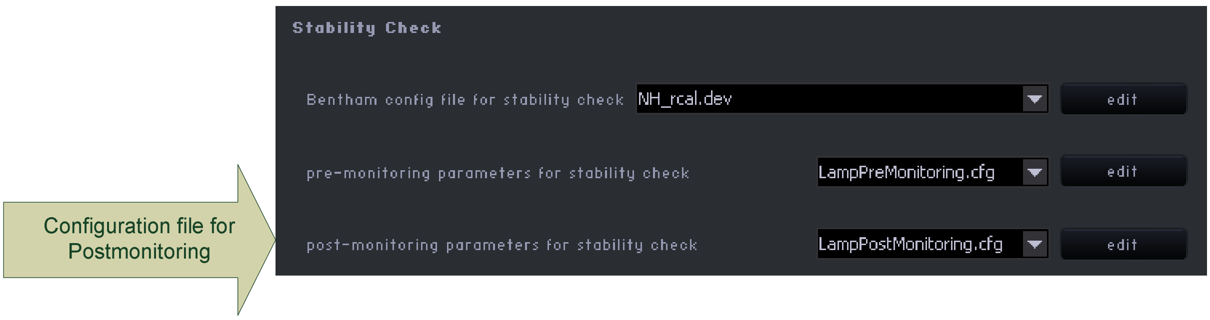


Fig. 69 The configuration file for Postmonitoring is listed under "Stability check"