L-DALI™

DALI Light Controller

User Manual

LOYTEC electronics GmbH

Contact

LOYTEC
Blumengasse 35
A-1170 Vienna
AUSTRIA/EUROPE
support@loytec.com
http://www.loytec.com

Version 5.2

Document 88077110

Contents

Abbreviations

100BaseT ....100 Mbps Ethernet network with RJ-45 plug

Aggregation.........................Collection of several CEA-709 packets into a single CEA-852 packet

AST.....................................Alarming, Scheduling, Trending

BACnet ....Building Automation and Control Network

BBMD.................................BACnet Broadcast Management Device

BDT ....Broadcast Distribution Table

CA......................................Certification Authority

CC ....Configuration Client, also known as CN/IP Device

CEA-709 ....Protocol standard for LONWORKS networks

CEA-852 ....Protocol standard for tunneling CEA-709 packets over IP channels

CN......................................Control Network

CN/IP ....Control Network over IP

CN/IP Channel....................logical IP channels that tunnels CEA-709 packets according CEA-852

CN/IP packet.......................IP packet that tunnels one or multiple CEA-709 packet(s)

COV ....change-of-value

CR ....Channel Routing

CS.......................................Configuration Server that manages CEA-852 IP devices

DA......................................Data Access

DALI..................................Digital Addressable Lighting Interface, see IEC 62386

DHCP.................................Dynamic Host Configuration Protocol, RFC 2131, RFC 2132

DL ....Data Logger, Datenlogger (Webservice)

DNS ....Domain Name Server, RFC 1034

DST....................................Daylight Saving Time

GMT...................................Greenwich Mean Time

IP...Internet Protocol

LSD Tool ....LOYTEC System Diagnostics Tool

IP-852................................Logical IP-Channel, for tunneling CEA-709 packets according to CEA-852

LAN ....Local Area Network

LSD Tool ....LOYTEC System Diagnostics Tool

MAC ....Media Access Control

MD5...................................Message Digest 5, a secure hash function, see Internet RFC 1321

MS/TP................................Master/Slave Token Passing (this is a BACnet data link layer)

NAT ....Network Address Translation, see Internet RFC 1631

NV......................................Network Variable

OPC....................................Open Process Control

OPC UA.............................OPC Unified Architecture

PEM ....Privacy Enhanced Mail

RNI.....................................Remote Network Interface

RTT .... Round-Trip Time
RTU.................................... Remote Terminal Unit
SL .... Send List
SMTP .... Simple Mail Transfer Protocol
SNTP.................................. Simple Network Time Protocol
SNVT .... Standard Network Variable Type
SSH..................................... Secure Shell
SSL .... Secure Socket Layer
TLS..................................... Transport Layer Security
UI.. User Interface
UNVT................................. User-Defined Network Variable Type
UTC.................................... Universal Time Coordinated
WLAN .... Wireless LAN
XML .... eXtensible Markup Language

1 Introduction

1.1 Overview

The L-DALI controllers for LONMARK and BACnet systems are DALI gateways with built-in light controller functionality.

DALI (Digital Addressable Lighting Interface) is defined in the international standard IEC 60929 Annex E and the newer IEC 62386. It is used to dim and switch luminaries from most leading manufacturers. DALI also supports devices like multi-sensors (e.g. for brightness, occupancy, etc.) and intelligent switches.

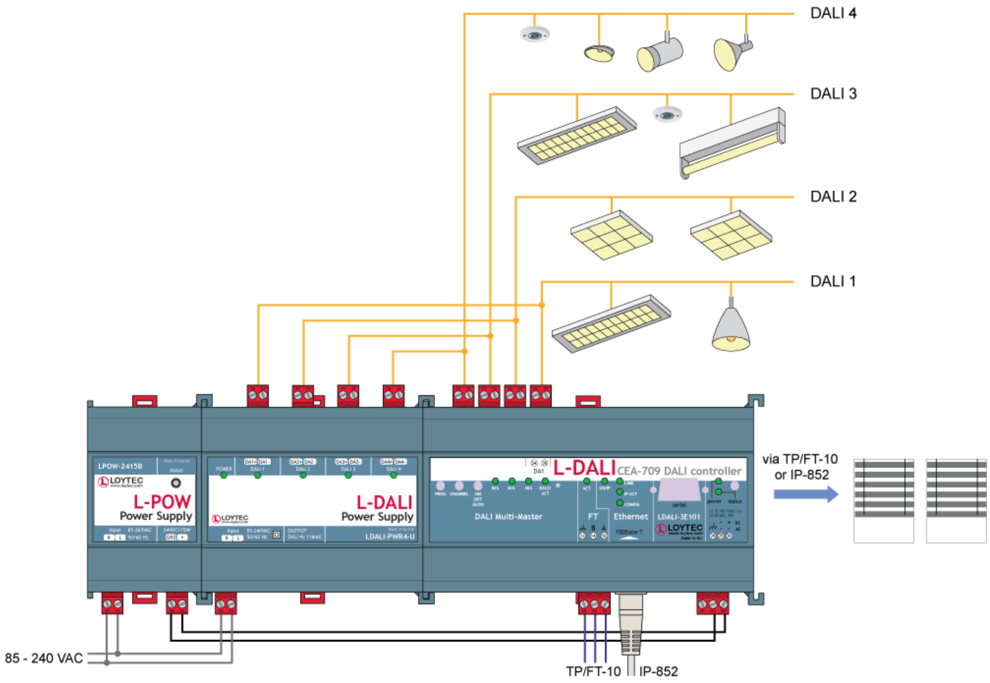


Figure 1: L-DALI supports up to four DALI channels.

The L-DALI lineup features 1, 2, or 4 independent DALI channels. Up to 64 DALI-based luminaries per DALI channel can be controlled individually or via 16 groups. All luminaries are monitored for lamp defect. L-DALI can provide this information to the Building Management System (BMS) through its NV or BACnet interface respectively. For CEA-709 communication IP-852 (Ethernet/IP) and TP/FT-10 are supported, for BACnet communication BACnet/IP and BACnet MS/TP are supported.

LDALI-3101-U, LDALI-E101-U, LDALI-3E101, LDALI-3E102, and LDALI-3E104 are gateways connecting a DALI network to a CEA-709 network, LDALI-ME204 and LDALI-E201-U are gateways connecting a DALI network to a BACnet network. On the DALI network the L-DALI controller represents a DALI master controller. On the CEA-709 or BACnet network the L-DALI controller offers a NV interface or a BACnet server objects respectively to control the ballasts and the sensors connected for the DALI network.

The constant light controller (LONMARK functional profile #3050, BACnet Loop object) allows controlling DALI ballasts with lighting nodes on the CEA-709 or BACnet network. Occupancy and lux-level sensors for constant light control and buttons for manual operation can be integrated via the CEA-709/BACnet interface or directly on the DALI side.

On the LONMARK version a built-in sunblind controller (LONMARK functional profile #6111) interacts with other nodes on the CEA-709 network. Interaction between the sunblind and constant light controller applications allows adjusting sunblinds depending on the constant light controller output, e.g. for energy saving concepts.

L-DALI supports automation functions such as alarming, scheduling, and trending. The L-DALI controllers offer local scheduling services including the possibility to configure several local and remote 24 hour schedulers through the Web UI. Alarming includes the functionality to generate, deliver, acknowledge, and display alarm conditions and logs regardless whether the condition comes from the DALI or the CEA-709/BACnet network. The trending capability includes periodic and event triggered data logging of values and time stamps. Alarms and trend data are stored on the device accessible via an FTP connection as CSV files.

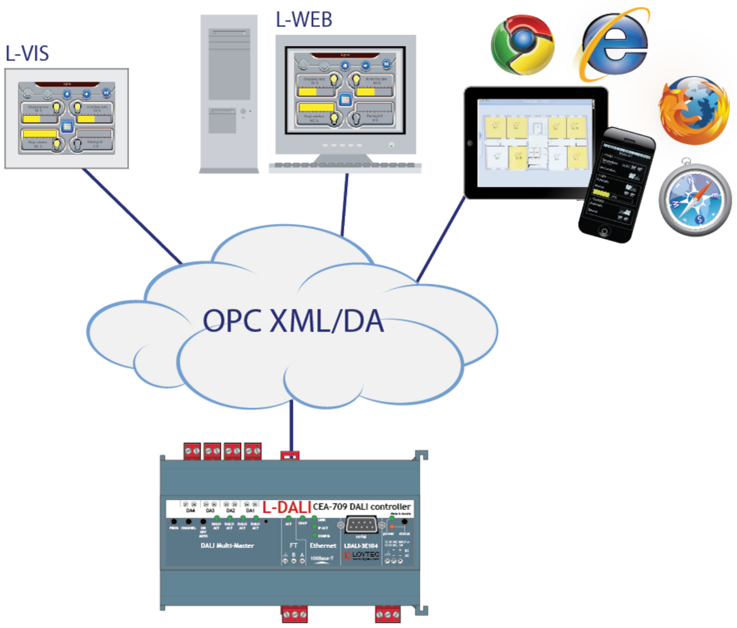


Figure 2: Using L-DALI together with the L-WEB product family.

L-DALI devices can be integrated with LWEB-802/803 visualization and LWEB-900 using OPC XML-DA. LWEB-802/803 and LWEB-900 can be used for visualization of an L-DALI based lighting control system. LWEB-900 offers additional functionality like saving trend and log files for long term storage, easy managing of remote schedulers and calendars, data analysis and report generation. L-DALI supports event-driven e-mail notification as a result of a predefined action triggered by a specific status or an exceeded high limit. e-mail notification can also be used to forward trend and log files to central SQL databases for long term storage (see LWEB-900 for details).

L-DALI supports common mathematical operations and functions, as well as Boolean expressions.

Some lighting controller applications do not need a connection to the CEA-709 or BACnet network. Therefore the L-DALI controller can control a DALI network as a standalone device with the aid of the supported automation functions (alarming, scheduling, and trending). Connected to an IP network, remote access allows to set parameters and to read the system status.

The complete configuration can be done through the built-in Web server. The commissioning and maintenance of the DALI system can be done using a standard Web browser on a PC. The configuration can also be done via an PC based configuration software (LONMARK/CEA-709 models only).

The L-DALI offers the following features:

- DALI gateway and controller
- Supports 1, 2, or 4 DALI channels (dependent on model)
- Direct control of up to 64 DALI devices per DALI channel
- Direct control of up to 16 DALI groups per DALI channel
- Scene control for up to 16 groups and one broadcast scene per DALI channel
- Detect lamp and ballast failure on DALI luminaries and signals
- Simple replacement of (broken) DALI devices (no configuration tool required)
- DALI Multi-Master capable
- Support for DALI sensors and buttons
- Built-in DALI protocol analyzer
- Supports Alarming, Scheduling, and Trending (AST™)
- Supports common mathematical operations and functions as well as Boolean expressions
- Supports event-driven e-mail notification
- Supports periodic testing of emergency lights
- Supports lamp burn-in mode
- Support calculation of energy consumption and run-hours
- OPC XML/DA (LWEB integration)
- Configuration via Web interface
- Firmware update via serial, CEA-709, or Ethernet port
- Operating Voltage: 12-35 V DC or 12-24 V AC
- 157 x 86 x 60 (L x B x H in mm) or 9 TE

- DIN rail mountable

LONMARK/CEA-709 models only:

- Fully compliant with CEA-709, CEA-852, and EN 14908 standard
- Supports LONMARK TP/FT-10 or CEA-852 Ethernet (IP-852) channels (selectable)
- Control of DALI capable ballasts via NVs
- Retrieve information from DALI capable sensors via NVs
- Supports LONMARK Functional Profiles:
    - Lamp Actuator #3040
    - Light Sensor #1010
    - Occupancy Sensor #1060
    - Constant Light Controller #3050
    - Sunblind Controller #6111
    - Scheduler #0007
    - Calendar #0006

BACnet models only:

- Fully compliant with BACnet standard ANSI/ASHRAE 135-2008
- Supports BACnet/IP or BACnet MS/TP (selectable)
- Control of DALI capable ballasts and sensors via BACnet server objects
- Retrieve information from DALI capable sensors via BACnet server objects
- BACnet client functionality (configurable)
- Supports the following BACnet server objects:
    - Analog Output objects to control DALI ballasts, groups, and channels
    - Multi-State Output objects for scene control of DALI groups and channels
    - Analog Input objects providing feedback from DALI ballast, groups, and channels
    - Analog Input objects providing status information from DALI groups and channels
    - Accumulator objects providing estimated energy usage of DALI groups and channels

- Analog Input object providing lux level information from supported DALI sensors
- Binary Input objects providing occupancy information from supported DALI sensors
- Loop objects providing constant light controller functionality

1.2 L-DALI Models

This Section provides an overview of the different L-DALI models in Table 1. This table identifies the different features of the L-DALI models. Models that possess a certain feature have a check mark (✓) in the respective column. If a feature is not available in the particular model, the column is left blank.

Feature \ Model	3E101	3E102	3E104	3101-U	E101-U	ME204	E201-U
DALI Power Supply				✓	✓		✓
DALI Channels	**1**	**2**	**4**	**1**	**1**	**4**	**1**
DALI Emergency Lights Test	✓	✓	✓	✓	✓	✓	✓
Constant Light Controller	✓	✓	✓	✓	✓	✓	✓
Sunblind Controller	✓	✓	✓	✓	✓		
Display with Jog Dial				✓	✓		✓
LON TP/FT-10	✓	✓	✓	✓			
LON IP-852 (Ethernet/IP)	✓	✓	✓		✓		
BACnet MS/TP						✓	
BACnet/IP (Ethernet/IP)						✓	✓
Configuration via WEB UI	✓	✓	✓		✓	✓	✓
Configuration via Software	✓	✓	✓	✓	✓	✓	✓
Console connector	✓	✓	✓			✓	

Table 1: Available features in different L-DALI models.

Important: ***In the following text models with BACnet interface are referred to as LDALI-20X and models with LONMARK/CEA-709 interface are referred to as LDALI-10X.***

1.3 Scope

This document covers L-DALI devices with firmware version 5.2 and the L-INX Configuration Software version 5.2.

2 What's New in L-DALI

2.1 New in L-DALI 5.2

LINX Configurator replaces LDALI Configurator

The L-DALI models are now supported by the LINX Configurator PC software. The LDALI Configurator software therefore is no longer available.

LINX Configurator support for LDALI-20X models

The LINX Configurator now supports LDALI-20X models, including DALI offline configuration workflow and the configuration of alarming, scheduling and trend log objects, e-mail templates, math objects, data point connections etc. Custom user registers and BACnet objects can be created and data points belonging to the static L-DALI application interface can be OPC exposed. Further, the BACnet network can be scanned and client maps can be created from local BACnet objects to remote BACnet objects.

Support for LOYTEC LDALI-MS1, LDALI-BM1 and LDALI-RM1

The L-DALI now supports the LDALI-MS1 multi-sensor, the LDALI-BM1 push-button coupler and the LDALI-RM1 relay module. These devices are an optimal addition to the L-DALI controller and allow designing cost effective and flexible DALI lighting systems. For more information on these devices see the corresponding datasheet and Section 8.4.4.

DALI Button Functions

For DALI button devices supporting configurable button functions these functions can be configured using the Web-UI (online) and the LINX Configurator (online and offline). The following functions are available to control DALI groups (see Section 5.3.4.7 and 7.6):

- Dim up, dim down,
- Off,
- On with last dimming value,
- Scene recall: 1–15,
- Dim to a specified value in %,
- Color temperature warmer/colder.

On LDALI-10X models the state of a button device input can be forwarded to a network variable as an alternative to directly control DALI lights (see Section 8.2.11.1).

DALI Scene Configuration

A new page was added to the L-DALI Web-UI allowing configuration of DALI scenes (see Section 5.3.5). Scene configuration also supports colour scenes with colour temperature and changeable light colour.

DALI Installation Wizards

A DALI search wizard was added to the DALI installation Web-UI to quickly locate and assign DALI devices. When preconfiguring DALI devices in an offline workflow (name, group assignment, DALI parameters etc.) the wizard assists in locating the corresponding physical device once online (see Section 5.3.4.2).

On the Constant Light Controller Web-UI a sensor search wizard was added. Once groups were configured to a CLC instance this wizard allows to automatically search for corresponding DALI sensors in the areas lit by these groups (see Section 5.3.2).

Identification of DALI Devices using Physical Selection

Devices which can be physically selected can be identified by selection during the installation and commissioning process of the DALI network. A DALI button can be selected by pressing one of the buttons, an occupancy sensor can be selected by triggering occupancy. Now the last physically selected device is marked in the DALI scan results on the DALI Installation Web-UI and can be therefore identified.

Online Commissioning

The Web interface of the device now provides an online commissioning tool for the respective networking technologies. Using this tool, data point configurations can be created based on placeholder devices, which are marked to be commissioned later. The necessary addressing information can be assigned later on the Web interface by scanning for devices online or be entered manually. Device replacement is also possible in the commissioning Web interface without the need to edit devices in the data point configuration. For more information on the commissioning Web UI please refer to Section 5.3.

Web Interface

The Web interface of the device offers a number of new features:

- A new device info page provides a quick overview of all relevant operational parameters, such as CPU load, active protocols, time synchronization and many more.

- The trend log configuration on the Web UI now also provides a preview tab, which shows a chart of the trend log data. The trend chart allows zoom, scrolling and hiding specific data curves, as shown in Figure 3.

Figure 3: New trend chart on the Web UI

- An improved DALI sensor calibration Web-UI allows to dim DALI groups directly from the sensor calibration Web-UI page making it easier to adjust the artificial light near the calibrated sensor. Further the calibration page can be accessed directly from the CLC Bindings page.

Scheduler

The scheduler objects have been extended by the following new features:

- Color support in BACnet and generic schedulers allows consistent assignment and display of preset colors in L-WEB, L-VIS and the Configurator scheduler UIs.

- Event auto-prune removes passed events, if the capacity of exception events in a scheduler gets low.

- The scheduler default for LONMARK and generic schedulers is extended by a “silent” mode. In this mode the scheduler becomes inactive as soon as the last event is withdrawn. This mode can be used in event-driven scheduler models.

- Generic schedulers allow specifying an existing value preset as the schedule default. LONMARK and BACnet schedulers try identifying a matching preset name from the schedule default value.

Format Strings in E-Mails

Data point variables used in e-mail templates can now use format strings to specify their numeric appearance in the e-mail text.

Storing Project Documentation directly on the Device

A new page on the L-DALI Web-UI allows uploading project documentation and setting-up links to external documentation files (see Section 5.2.20). Storing documentation directly on the device ensures its availability (e.g. for a service technician) later on.

2.2 New in L-DALI 3.2

This section describes the major changes and new features. For a full list of changes refer to the Readme file.

New Constant Light Controller Modes

The constant light controller application offers some new Modes:

- The modes CONTROL, REGULATOR and PRESENCE are now available in variations with the extension AUTO. In these modes any active override will be automatically relinquished when the room enters “unoccupied” state. This allows to reactivate automatic mode when the room is unoccupied. The light will be switched off and – if necessary – switched on when someone enters the room.

- As a special variation of the AUTO modes, modes with the extension BEDROOM are available. In this mode the manual override is only relinquished when the lights are switched on. This ensures lights are switched off when the room is unoccupied, but stay off when commanded to off manually.

- The mode AMBIENT allows switching on and off lights depending on the measured lux level. It is therefore suitable for decorative lighting (facade illumination, shop window lighting, floor lighting, etc.).

The modes CONTROL and REGULATOR (incl. all variations) now consider the configured Unoccupied Level when switching off, similar to the mode PRESENCE. This allows dimming down to a configured level when the room is unoccupied instead of switching off. Typical applications are a reception area or similar areas where the light is never turned off completely.

Note: *If the an LDALI-10X device was not shipped with firmware 3.2 the new modes are only available after the datapoint configuration was up- and downloaded again using the LINX Configurator.*

Further information on the constant light controller modes can be found in Sections 8.2.8.3 and 8.3.5.2 and for the behavior when switching off in Sections 8.2.8.8 and 8.3.5.7.

Overriding the Constant Light Control with DALI Buttons

When using internal manual bindings the constant light controller now detects when one of the groups it controls is dimmed by another DALI master. Such and override is interpreted similar to an override using the CEA-709/LONMARK or BACnet interface. Typically such a DALI master is a DALI push-button coupler or a similar device for manual control of DALI lights. As these devices directly control the DALI groups, they allow local operation of the lights independent of the L-DALI.

Constant Light Controller Bindings across Channels

DALI groups and DALI sensors can be linked to constant lights controller instances using internal bindings across DALI channels (see Sections 5.3.2 and 7.6).

Support for DALI ballasts with Colour Control

In the LDALI-10X models DALI ballasts of type “colour control” (device type 8) can be controlled. The LDALI-10X supports devices with changeable colour temperature and with coloured light. The latter can be used for effect lighting, while the previous can be used to simulate the colour temperature of the sun during the course of a day (warmer white during morning and evening, cooler white at noon).

Support for additional DALI Sensors

Support for the ThebenHTS PlanoSpot 360 DALI multi-sensor was added.

Change the DALI Short Address

If required the DALI short address, which was assigned to a DALI device by the L-DALI, can be changed via the Web-UI (see Section 5.3.4.9).

Generic Scheduler

Generic schedulers – like generic trends and alarms – can now be created, that are neither CEA-709 nor BACnet objects. Generic schedulers appear next to the generic alarm folder and are ready-to-use on any device. This is beneficial for creating technology-independent applications. Generic schedulers can write to any technology as well as data point favorites and are the ideal solution, if configured via LWEB-900 only. For more information on creating generic schedulers refer to Section 7.10.

Web Interface

The Web interface of the device offers a number of new features:

- Live update of values in the data point list. This allows monitoring values without repeated pressing of reload. Data point structures can be expanded or collapsed for better overview.

- Breadcrumb navigation has been added to the data point list. This gives faster access to sub-folders.

- A new firmware upgrade menu on the Web interface allows online checking for firmware updates and upgrading by selecting a local firmware file. All this is possible without starting the Configurator.

- The trend overview page displays current trend log states and provides controls for easy trend data upload in CSV format.

- The DALI group page allows switching groups on and off to test the group assignment.

Application Objects

Application objects such as math objects, e-mail templates, and alarm logs can now be organized in folders. Copy and paste of application objects between Configurators has been improved. Math objects now allow single constants and single variable assignments, such as “=5” or “=v1”. Input variables can be configured to trigger a new calculation or not.

3 Quick-Start Guide

This Chapter contains step-by-step instructions on how to configure the L-DALI for the simple project shown in Figure 4.

The project consists of a single room (Room 306) which is illuminated by four DALI lamps. Two of those lamps form a light band near the windows of the room and the other two lamps form a light band near the corridor. The room is equipped with a DALI multi-sensor which acts as both a light sensor and an occupancy sensor. The build-in constant light controller of the L-DALI device uses the input from the DALI multi-sensor and dims the DALI ballasts accordingly. For manual override a DALI push-button is installed. The sunblind controller is not used in this quick-start example.

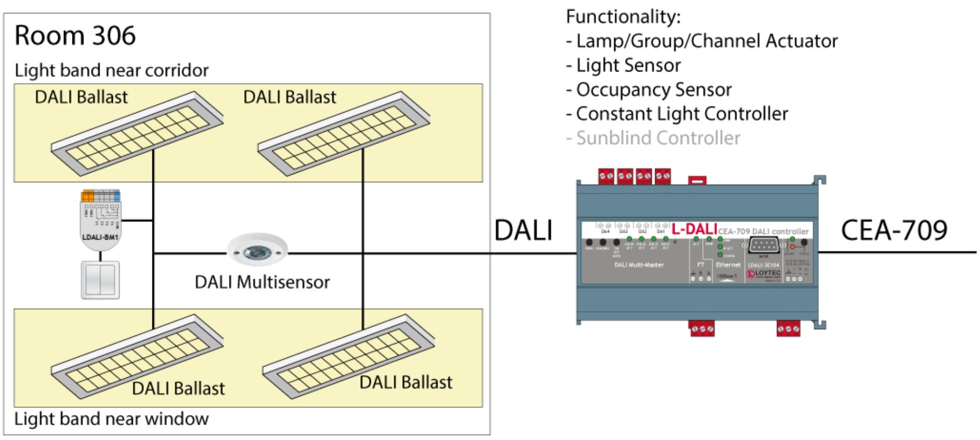


Figure 4: Quick-Start Example Project.

3.1 Hardware Installation

3.1.1 Models without built-in DALI Bus Power Supply

For models without built-in DALI bus power supply it is recommended to use the LDALI-PWR4-U or LDALI-PWR2-U power supply together with the L-DALI. Connect the L-DALI to the LDALI-PWR4-U and to the DALI network as shown in Figure 5. To allow for easy configuration it is recommended to always connect the L-DALI to the Ethernet network. More detailed instructions are given in Chapter 4.

Important: ***Do not connect terminal 26 to earth ground!***

After the DALI ballasts have been installed and connected to the DALI network, the installation can be tested by following these steps:

1. Check that the DALI LEDs ("DALI x ACT", where x is 1 to 4) do not light up red. If one of these LEDs is red, check the proper connection of the bus power supply for the corresponding channel and check the DALI wiring for short-circuits.

2. Press the DALI mode button ("ON/OFF/AUTO") on the front panel of the L-DALI once. Now all DALI ballasts should be switched on (maximum level) and the DALI LEDs on the L-DALI should light up green.

3. Press the DALI mode button again. Now all DALI ballasts should be switched off and the DALI LEDs on the L-DALI should light up orange.

4. Press the DALI mode button again. This should not change the state of the DALI ballasts but return the L-DALI to the auto-mode (control via CEA-709/BACnet interface).

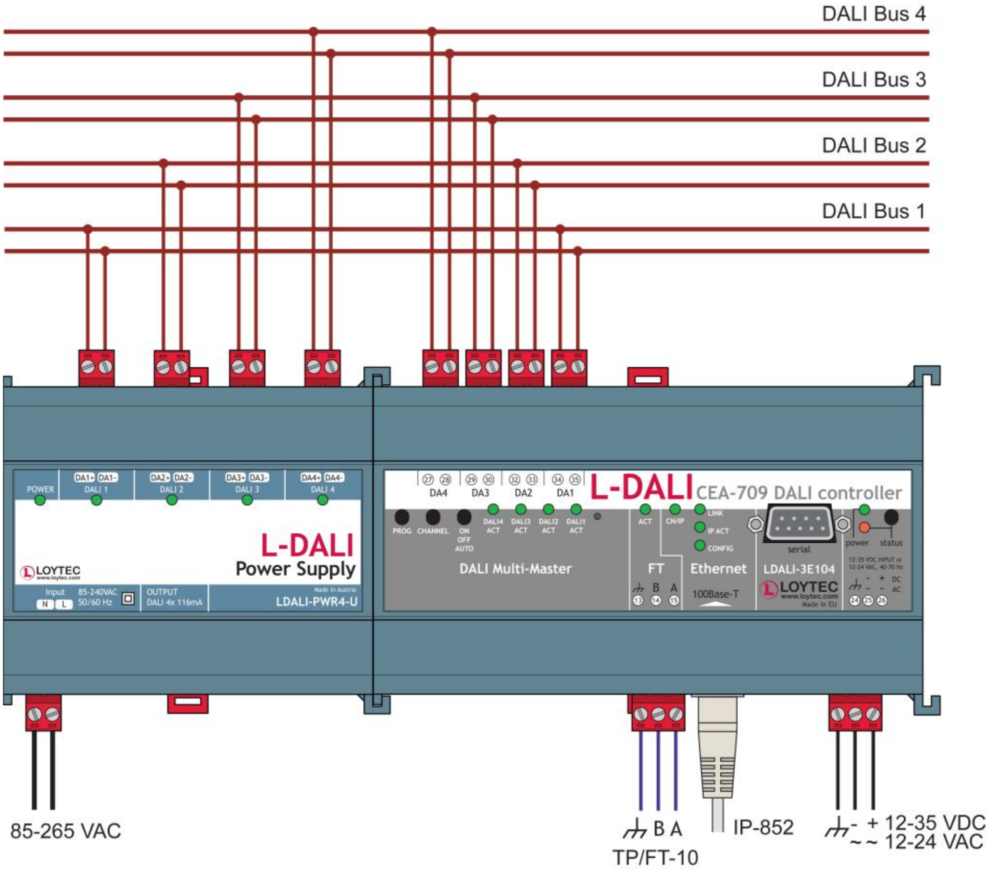


Figure 5: Basic Hardware Installation with external DALI bus power.

3.1.2 Models with built-in DALI Bus Power Supply

Connect the L-DALI to the power and to the DALI network as shown in Figure 5. To allow for easy configuration it is recommended to always connect the L-DALI to the Ethernet network (if available). More detailed instructions are given in Chapter 4.

After the DALI ballasts have been installed and connected to the DALI network, the installation can be tested by following these steps:

1. Check that the status LED ("status") does not light up red. If the LED is red, check the DALI wiring for short-circuits.

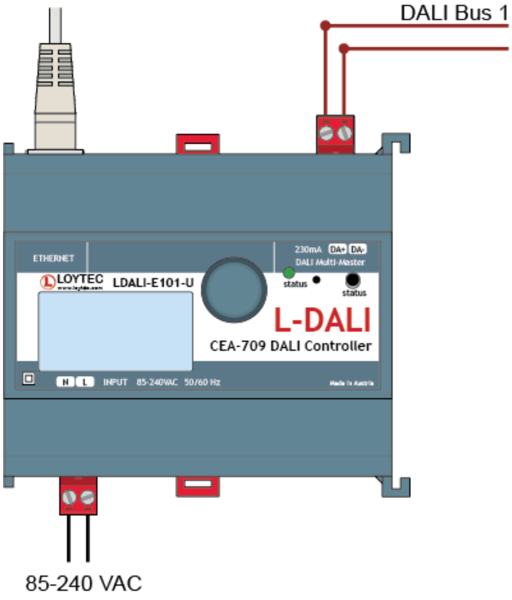


Figure 6: Basic Hardware Installation with integrated DALI bus power.

2. Turn the jog dial to navigate to the DALI menu item.

```
  LOYTEC LDALI-E101-U
    Unnamed Project
      192.168.3.51
# 28%
DALI »»       100FD
Datapoints »»
Device Settings »»
```

3. Press down to jog dial to enter the DALI menu.

```
          DALI
Manual Override: Auto
Assigned devices »»
Replace device
Rescan
Scan Results
Wink Duration:   5 s
```

4. Turn the jog dial to navigate to the menu item “Manual Override: Auto”.

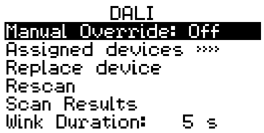
```
          DALI
Manual Override: Auto
Assigned devices »»
Replace device
Rescan
Scan Results
Wink Duration:   5 s
```

5. Press down the jog dial. Then turn it one step to the right. The manual override must change to “On”.

```
          DALI
Manual Override: On
Assigned devices »»
Replace device
Rescan
Scan Results
Wink Duration:   5 s
```

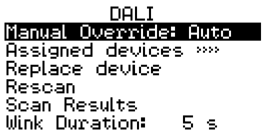
   Now all DALI ballasts should be switched on and the status LED on the L-DALI should light up green.

6. Turning the dial one more step right the manual override must change to “Off”.



Now all DALI ballasts should be switched off and the status LED on the L-DALI should light up orange.

7. Turn the dial one step right once more. This should not change the state of the DALI ballasts but return the manual override back to the auto-mode (control via CEA-709/BACnet interface).



3.2 Device configuration

The L-DALI can be configured via a console interface or via the Web interface. To configure the L-DALI, the following steps have to be performed:

1. Setup IP configuration (see Sections 3.2).

2. Setup the DALI network (see Section 3.4 or 3.5).

Note: *This setup procedure assumes the use of the IP interface.*

3.3 Configuration of IP Address

3.3.1 IP Configuration via Console

If your model is equipped with a console interface use a standard null-modem cable with full handshaking to connect COM1 of the PC to the Console on the device. Use a PC terminal program with the communication settings set to 38,400 bps / 8 data bits / no parity / 1 stop bit / no handshake. Power up the device or press **Return** if the device is already running. The following menu should appear on the terminal:

```
Device Main Menu
================

[1] Show device information
[2] Serial firmware upgrade
[3] System configuration
[4] DALI maintenance
[5] IP configuration
[6] CEA-852 device configuration
[7] CEA-709 configuration
[8] Reset configuration (factory defaults)
[9] Device statistics

[a] Data Points

[0] Reset device

Please choose:
```

Figure 7: Device Main Menu.

Select ‘5’ from the device main menu and enter the IP address, netmask, and gateway address. Note that you must use different IP addresses if you are using multiple IP devices in your setup.

```
IP Configuration Menu
============================================================
[1] DHCP          : disabled
[2] IP Address      : 192.168.1.254
[3] IP Netmask      : 255.255.255.0
[4] IP Gateway      : 192.168.1.1
[5] Hostname       : new
[6] Domainname      : <unset>
[7] DNS Servers      : <unset>
[9] MAC Address      : 00:0A:B0:01:0C:9F (factory default)
[0] NTP Servers      : <unset> (out-of-sync)
[b] Link Speed & Duplex  : Auto Detect

[q] Quit without saving
[x] Exit and save

Please choose:
```

Figure 8: Enter basic IP settings.

Press ‘x’ to save the IP settings and reset the device with the main menu item ‘0’ in order to let the new IP settings take effect.

Important!	***The default IP address 192.168.1.254 is only set for configuration access. It must be changed in order to make the device functional.***

3.3.2 IP Configuration via the Web Interface

As an alternative to the console interface the Web interface can be used to configure the device. In a Web browser enter the default IP address 192.168.1.254 of the L-DALI. Note that if your PC has an IP address in a subnet other than 192.168.1.xxx please open a command tool and enter the following route command to add a route to the L-DALI.

To Add a Route to the Device

1. Windows **START → Run**
2. Enter ‘cmd’ and click **OK**.
3. In the command window enter the command line

   `route add 192.168.1.254 %COMPUTERNAME%`

   In Windows7 replace %COMPUTERNAME% with the PC’s actual IP address.
4. Then open your Web browser and type in the default IP address 192.168.1.254.

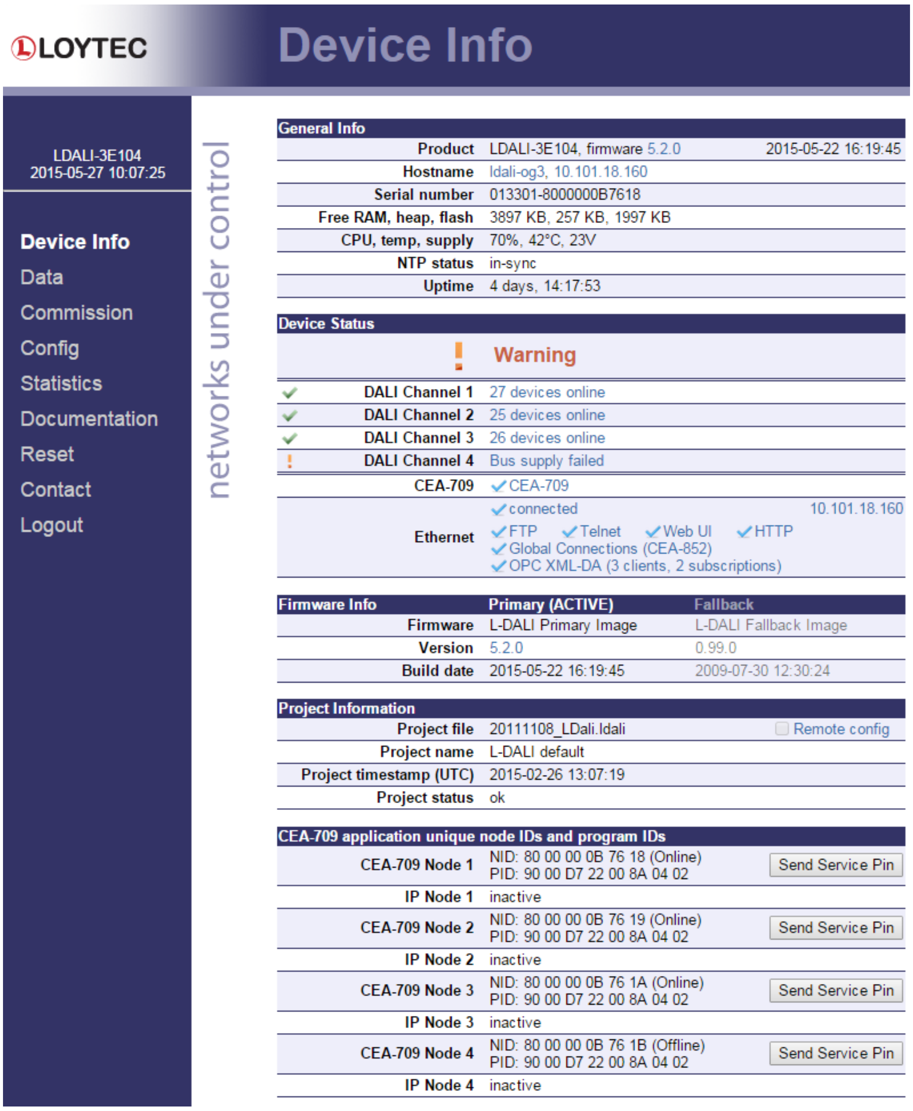


Figure 9: Example Start Screen.

5. Click on **Config** in the left menu. You will be asked to enter the administrator password in order to change the IP settings. Enter ‘loytec4u’ and select **Login**.

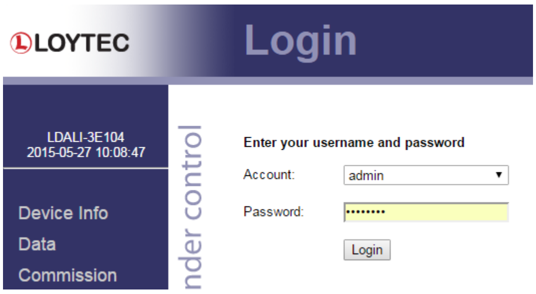


Figure 10: Enter ‘loytec4u’ as the default administrator password.

6. The Config menu opens. Click on **Port Config** in the Config menu and select **Ethernet** tab. Enter the IP address, the IP netmask, and IP gateway for this device as shown in Figure 11.

7. Press **Save Settings** and then reset the device by selecting **Reset** in the highlighted text. This changes the IP settings of the device.

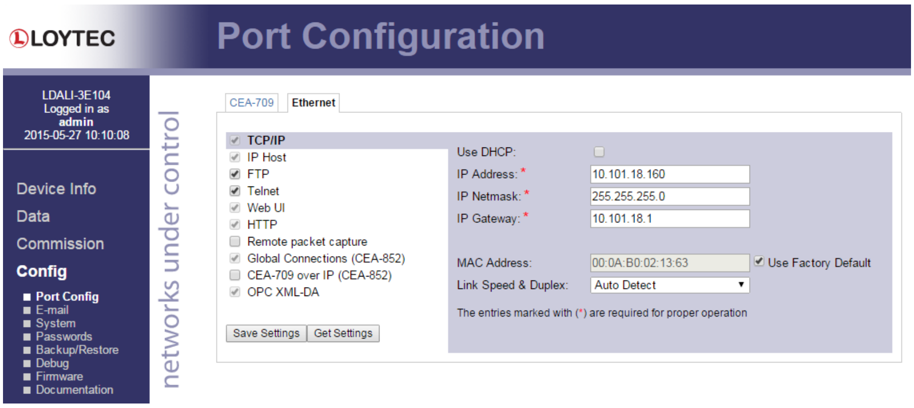


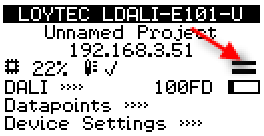
Figure 11: Enter IP address and gateway.

3.3.3 IP Configuration via the LCD Display

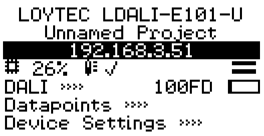
Device models with an LCD display can also be configured to their basic settings through jog dial navigation on the LCD UI. Turn the jog dial to navigate between menu items and press to enter a menu or go into selection mode. When in selection mode turn the jog dial to alter the value and press again to quit the selection. Some input fields provide acceleration. This means turning faster changes the value in larger increments.

To Set the IP Address on the LCD Display

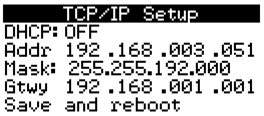
1. On the LCD main screen set the desired language. Navigate to the flag symbol, press the button and choose the desired language.



2. Navigate to the IP address on the main screen and press the button.



3. There navigate to the needed input fields, press and change the value. Press again to set the value. Continue to the next field.



4. Finally navigate to **Save and reboot** and press.

5. Acknowledge the reboot and the device reboots with the new IP address.

3.4 Configuration with PC Software

For the LDALI product family a PC based configuration software – the LINX Configurator – is available. As an alternative the web interface can be used for configuration (see Section 3.5).

Install the LINX Configurator software from the setup.exe. This file can be downloaded from www.loytec.com. This tool can be used as a stand-alone tool or as LNS plug-in. In this example we will use the LINX Configurator software as a stand-alone tool.

A detailed description of the LINX Configurator configuration software can be found in Chapter 7.

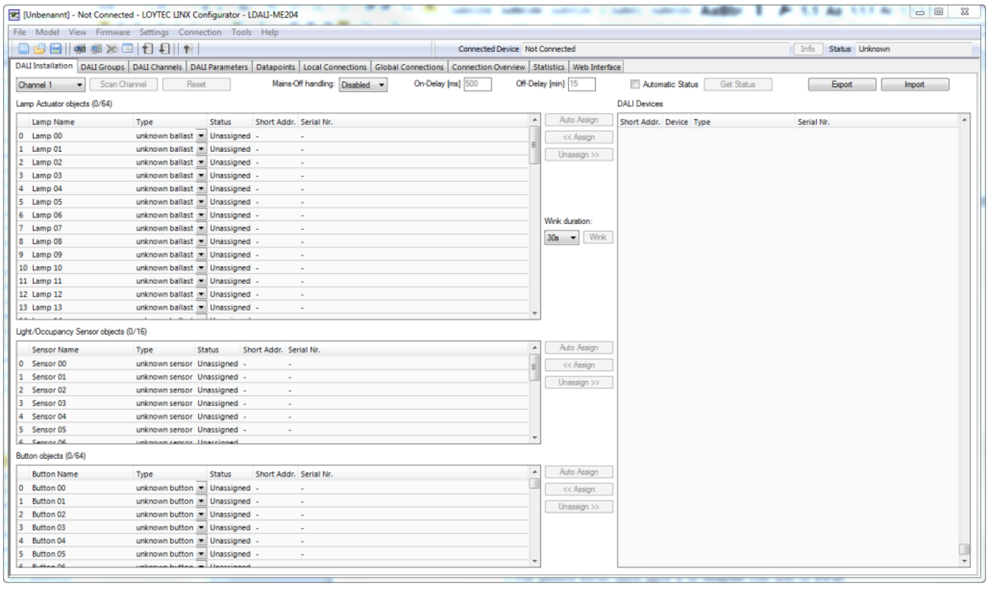


Figure 12: LINX Configurator software, Start.

3.4.1 Connect to Device in Stand-Alone Mode

1. Select the IP connection method by clicking on the **Connect to device** button in the tool bar as shown in Figure 13.

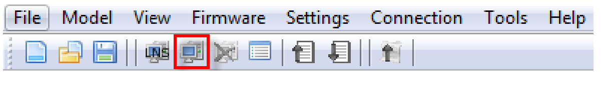


Figure 13: LINX Configurator software, Connect to device

2. In the Connection dialog (see Figure 14) choose connection type "Web Service", enter the IP address or hostname of the L-DALI and the devices "admin" password. The default "admin" password is 'loytec4u' (older firmware versions used 'admin').

3. Optionally, click on **New** and enter a user-defined name for this connection. That name can be selected later to connect. Click on **Save** to store that connection.

4. Click on **Connect**. This establishes the connection to the device.

5. The LINX Configurator software asks if you want to upload the current configuration of the device. You can cancel this dialog because in this quick-start we configure the device from scratch.

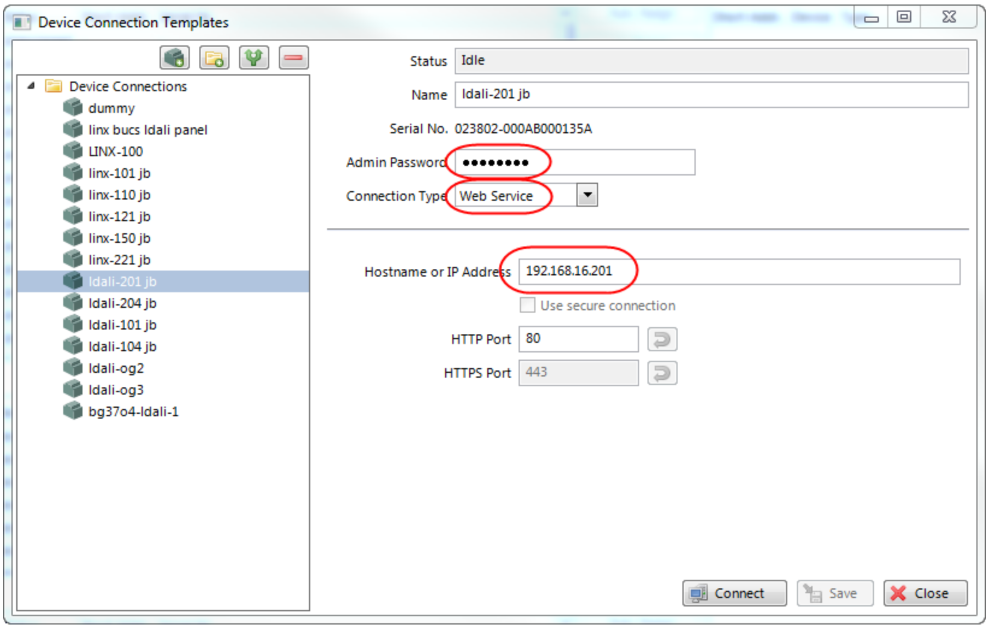


Figure 14: LINX Configurator software, Connection Dialog.

3.4.2 Scan DALI Channel

1. Change to the **DALI Installation** tab.

2. Select the DALI channel and click on the **Scan Channel** button shown in Figure 15.

Figure 15: LINX Configurator software, scan DALI Channel.

3. The L-DALI device scans the selected channel and displays all detected devices as shown in Figure 16. In case an error occurs see Section 11.4 for a description of the error codes and possible reasons.

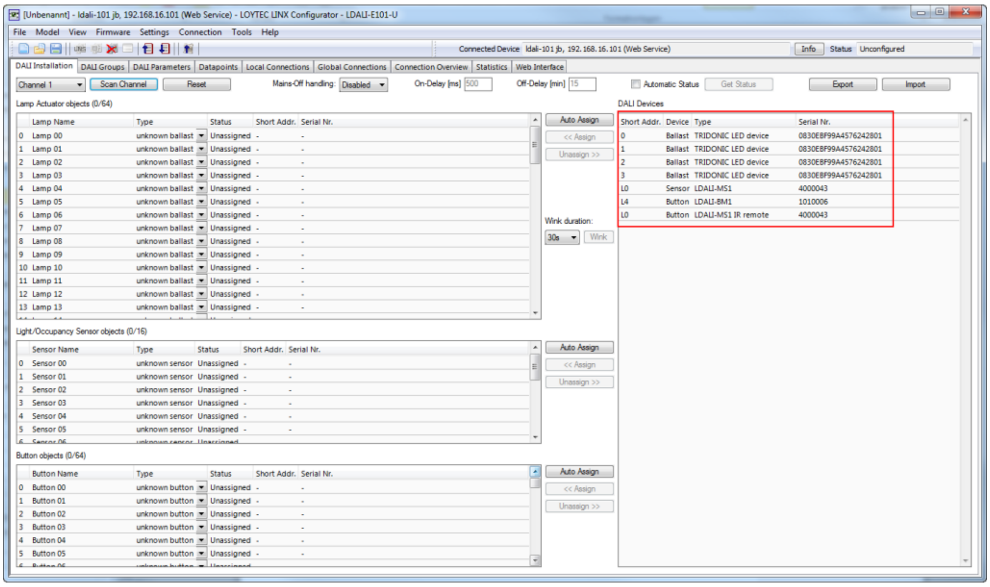


Figure 16: LINX Configurator software, detected DALI devices.

3.4.3 Assign Lamps, Sensor and Button to LONMARK/BACnet Objects

1. To identity which of the four detected DALI ballasts is which physical lamp, select one and click the **Wink** button. The corresponding lamp blinks for the configured wink duration.

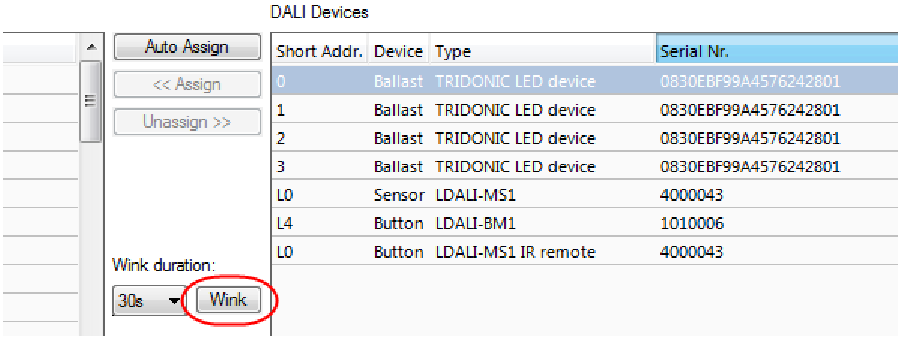


Figure 17: LINX Configurator software, Wink button.

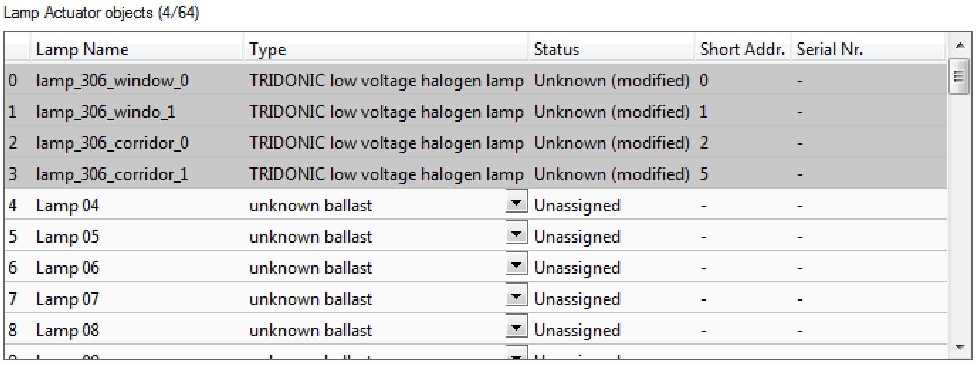


Figure 18: LINX Configurator software, assigned Lamp Actuator objects.

2. To assign an DALI ballast to an object, select the ballast in the list of **DALI Devices** and drag it to the desired position in the **Lamp Actuator objects** list on the left side. You can change the names of the lamp actuator objects. After you have assigned all lamps and changed the names the configuration should look as shown in Figure 18.

3. To assign the DALI multi-sensor to an object, select the sensor in the list of **DALI Devices** and drag it to the desired position in the **Light/Occupancy Sensor objects** list on the lower left side. After you have changed the names of the sensor object the configuration should look as shown in Figure 19.

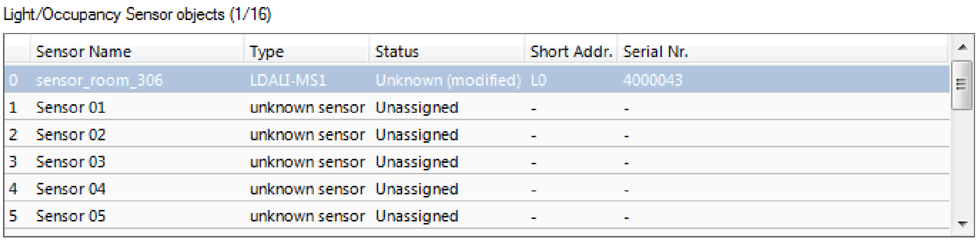


Figure 19: LINX Configurator software, Assigned Light/Occupancy Sensor Object.

4. Similar, assign the DALI button by dragging it to the desired position in the **Button objects** list.

3.4.4 Grouping Lamps

Lamps which are assigned to a group can be controlled together by the corresponding group object. We will create three groups: One for the lamps near the window, one for the lamps near the corridor and one for all lamps in the room.

1. In the LINX Configurator software switch to the **DALI Groups** tab.

2. Add the two lamps near the window to group 0, the two lamps near the corridor to group 1 and add all lamps in the room to group 2. Assign names to the groups as shown in Figure 20.

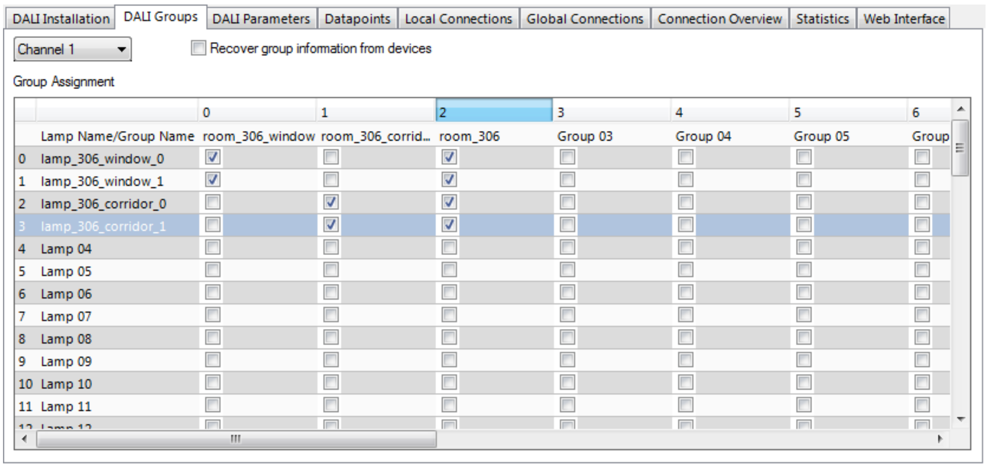


Figure 20: LINX Configurator software, DALI group configuration.

3.4.5 Parameterize the Constant Light Controller

1. In the LINX Configurator software switch to the **DALI Parameters** tab.

2. Select the constant light controller with index 0 as shown in Figure 21. The parameters of the selected constant light controller are displayed in the middle of the window (1).

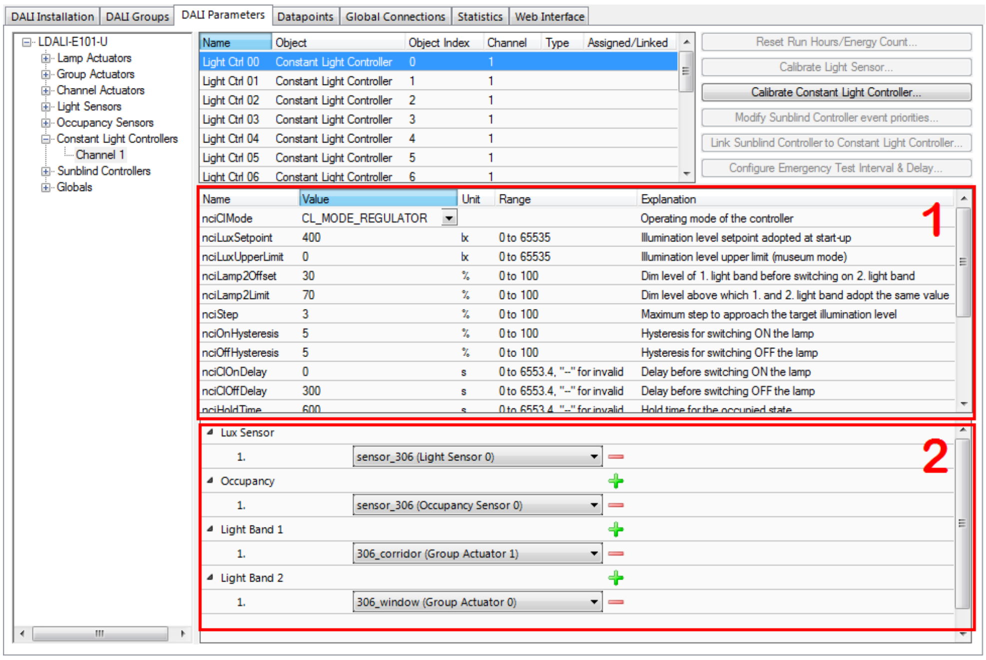


Figure 21: Parameterize Constant Light Controller.

3. Change the parameter *nciClMode* to CL_MODE_REGULATOR. This parameter selects the operating mode. The REGULATOR mode is used if a light/occupancy sensor is installed which measures the indoor illumination.

4. Change the setpoint of the constant light controller (*nciLuxSetpoint*) to the desired light level (e.g. 400 lux).

5. You can leave the default values for the remaining parameters. For a description of the constant light controller functionality and the parameters refer to Section 8.2.8.

6. Determine which sensors (occupancy & lux) are used as inputs to the constant light controller instance and which groups are controlled in the lower part of the window (2).

7. Click on the  to add an input or output. Use the drop down box to select a sensor (input) or a light group (output). In our example we use "sensor_306" for lux and occupancy sensor inputs, group "306_corridor" as output of the first (brighter) light band and group "306_window" as output for the second (darker) light band.

3.4.6 Parameterize the DALI button

1. Stay on the **DALI Parameters** tab and select the button object to which you have assigned your DALI button as shown in Figure 22.

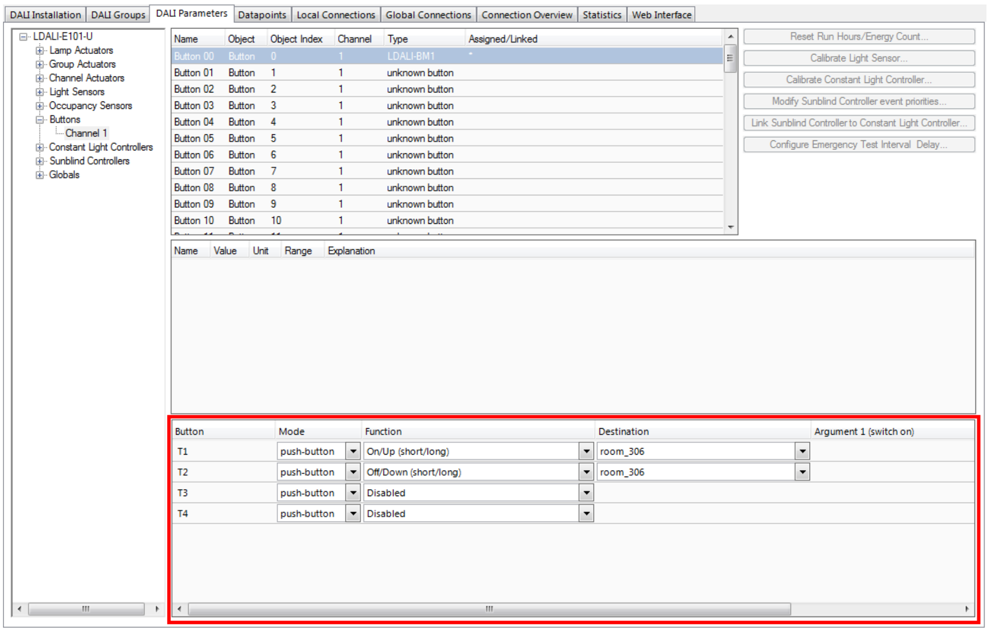


Figure 22: Parameterize DALI button functions.

2. Determine which function is executed and which group is controlled by which button input in the lower part of the window. In our example we use button T1 for "On/Up" and T2 for "Off/Down", both controlling group "room_306". This will allow us to control all lights in the room together.

3.4.7 Download Configuration

1. To download the configuration into the L-DALI device click on the **Download Configuration** button in the toolbar.

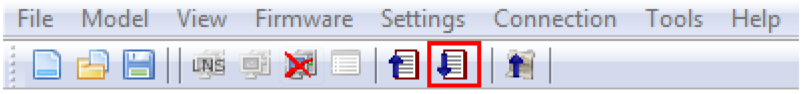


Figure 23: LINX Configurator software, Download Configuration.

2. Because we have changed only the DALI configuration and the parameters it is sufficient to check only DALI Configuration and Parameters in the following dialog. This speeds up the download process.

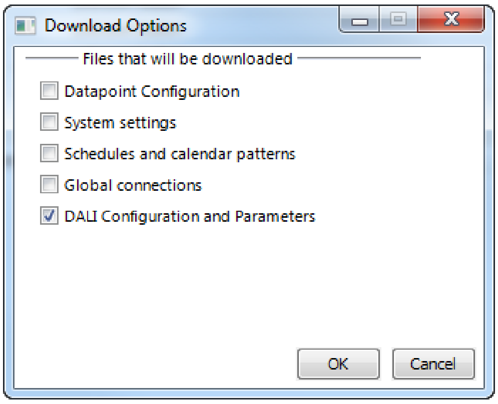


Figure 24: LINX Configurator software, Download DALI Configuration and Parameters.

3. After the download is complete, go back to the DALI Installation tab and press the **Get Status** button. Now the assigned DALI devices are displayed on a green background (see Figure 25). The green background color indicates that the communication with the device is OK. If there is a communication error, the device will be displayed with a red background color.

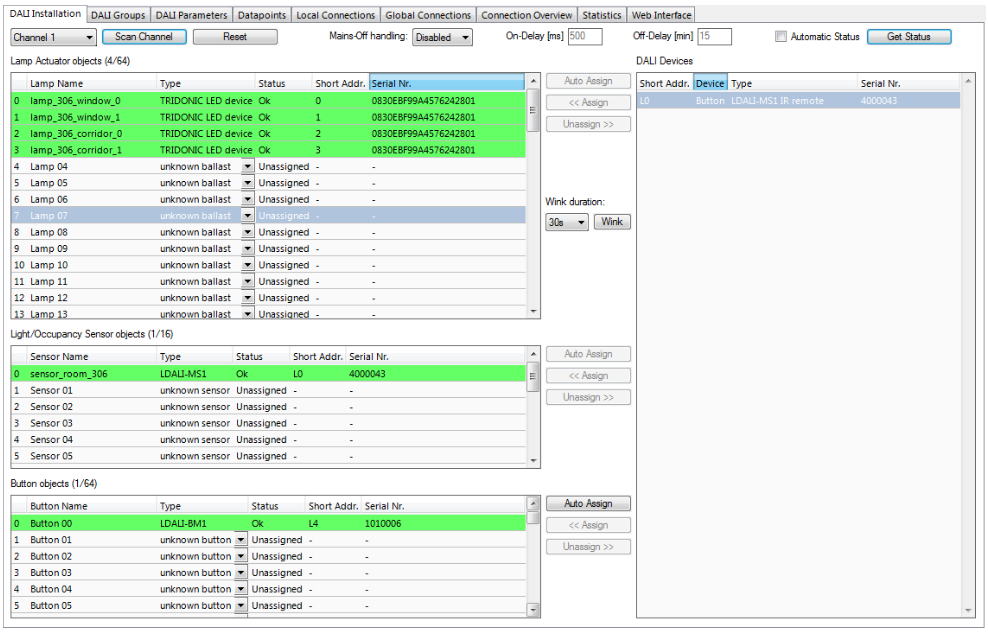


Figure 25: LINX Configurator software, After Configuration Download.

3.4.8 Calibrate Light Sensor

The L-DALI device allows calibrating the light sensor under up to seven different light conditions to counter any non-linearity of the sensor. However, in many cases it is enough to calibrate the sensor with a single light condition which is near the setpoint. In this quick-start only one lux measurement is performed.

1. In the LINX Configurator software switch to the **Parameters** tab.
2. Select the light sensor index 0 and click on the **Calibrate Light Sensor…** button as shown in Figure 26. The light sensor calibration window as shown in Figure 27 in displayed.

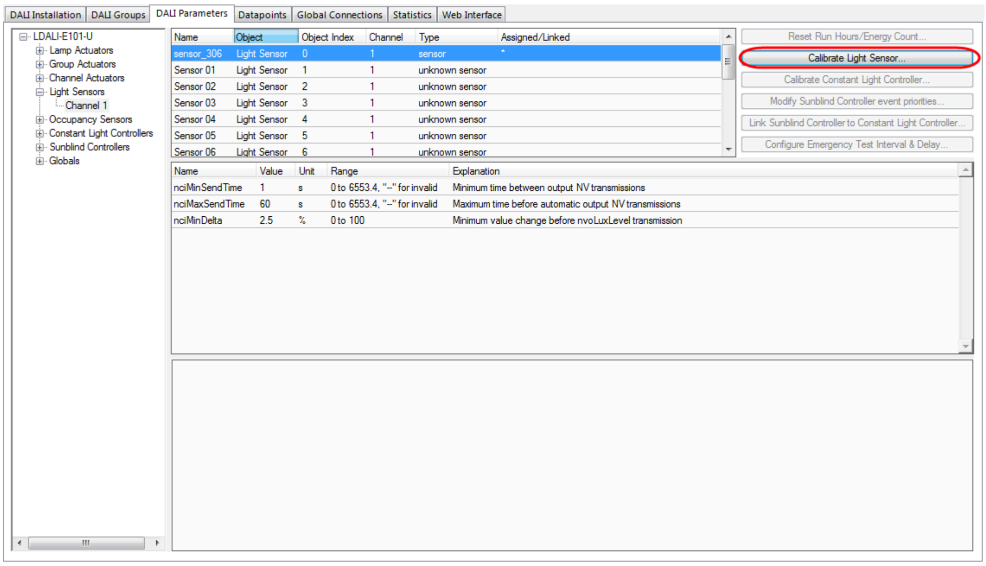


Figure 26: Parameterize Light Sensor.

3. Measure the current lux level at the reference area (e.g. desk) using a luxmeter.
4. Enter the measured lux level in the input field and select the index 0.
5. Press the **Calibrate** button.
6. Close the dialog by pressing the **Done** button.

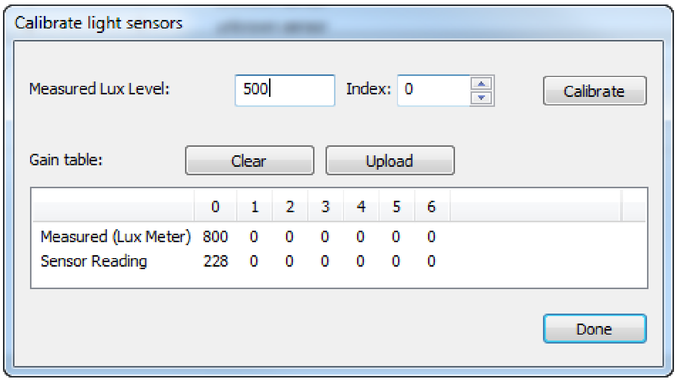


Figure 27: Calibrate Light Sensor.

3.5 Configuration using Web Interface

As an alternative to the LINX Configurator PC Software the Web interface of the L-DALI can be used. In a Web browser enter the IP address as set up in Section 3.3.

A detailed description of the L-DALI Web interface can be found in Chapter 5.

3.5.1 Scan DALI Channel

1. In the L-DALI Web interface click on **Commission** in the left menu. If not already logged in you will be asked to enter the administrator password. Enter 'loytec4u' and select **Login**.

2. The Commission menu opens. Click on **DALI Installation** in the Commission menu. The DALI installation page opens as shown in Figure 28.

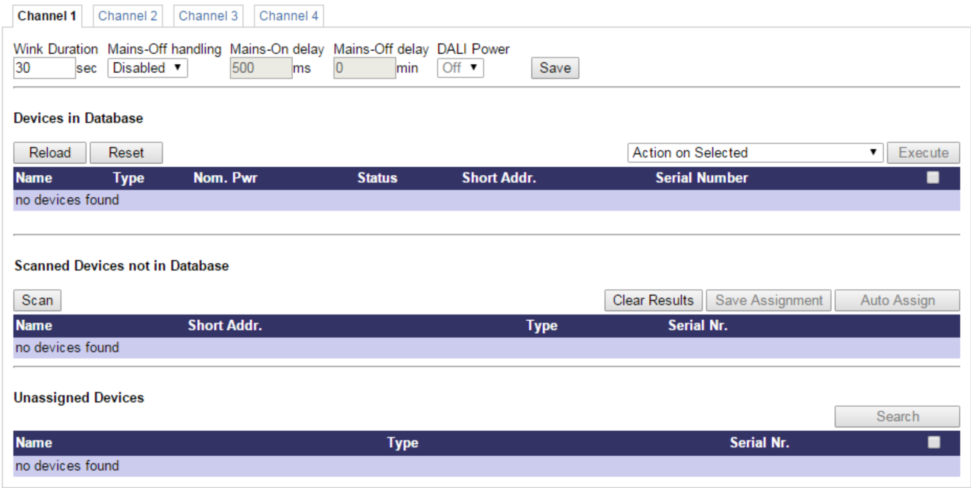


Figure 28: DALI Installation Web Interface: Initial view.

3. If the device supports more than one DALI channel select the DALI channel by clicking on the different tabs at the top of the page labeled **Channel 1**, **Channel 2**, etc. and press the **Scan** button.

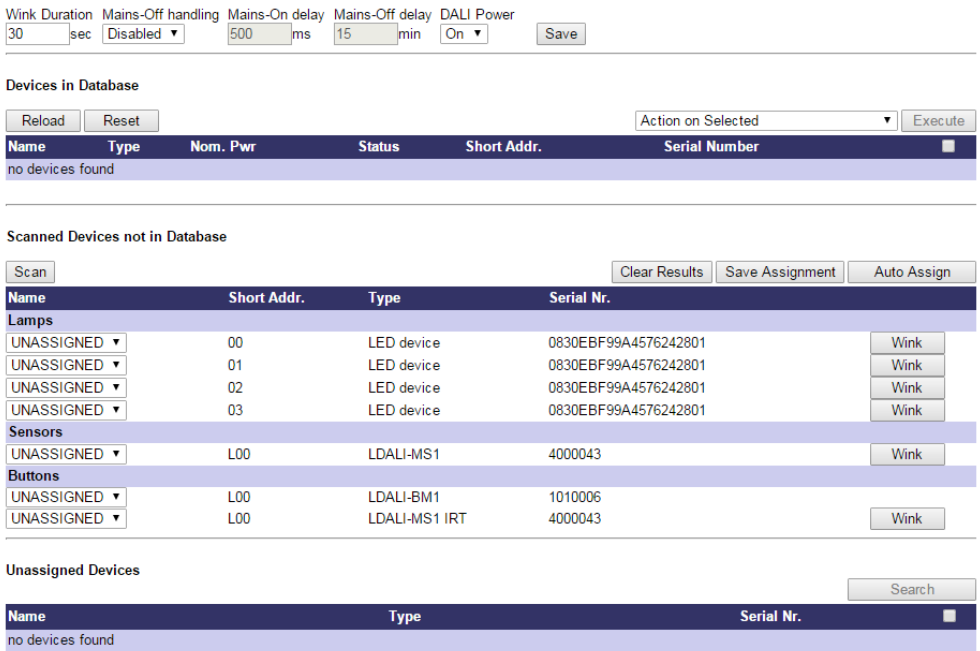


Figure 29: DALI Installation Web Interface: Detected DALI devices.

4. The L-DALI scans the DALI channel and lists the detected devices under **Scanned Devices not in Database** in the lower halve of the Web interface (see Figure 29). In case an error occurs see Section 11.4 for a description of the error codes and possible reasons.

3.5.2 Assign Lamps, Sensor and Button to LONMARK/BACnet Objects

1. To identity which of the four detected DALI ballasts is which physical lamp, select one and click the **Wink** button. The corresponding lamp blinks for the configured wink duration.

2. To assign a DALI device to a LONMARK or BACnet object, use the drop-down list available for each device in the **Scanned Devices not in Database** section (left column). This list contains all available fieldbus objects. When you are done, press the button **Save Assignment**. Alternatively to manual assignment press the button **Auto Assign** for random assignment. After you have assigned all lamps the configuration should look as shown in Figure 30.

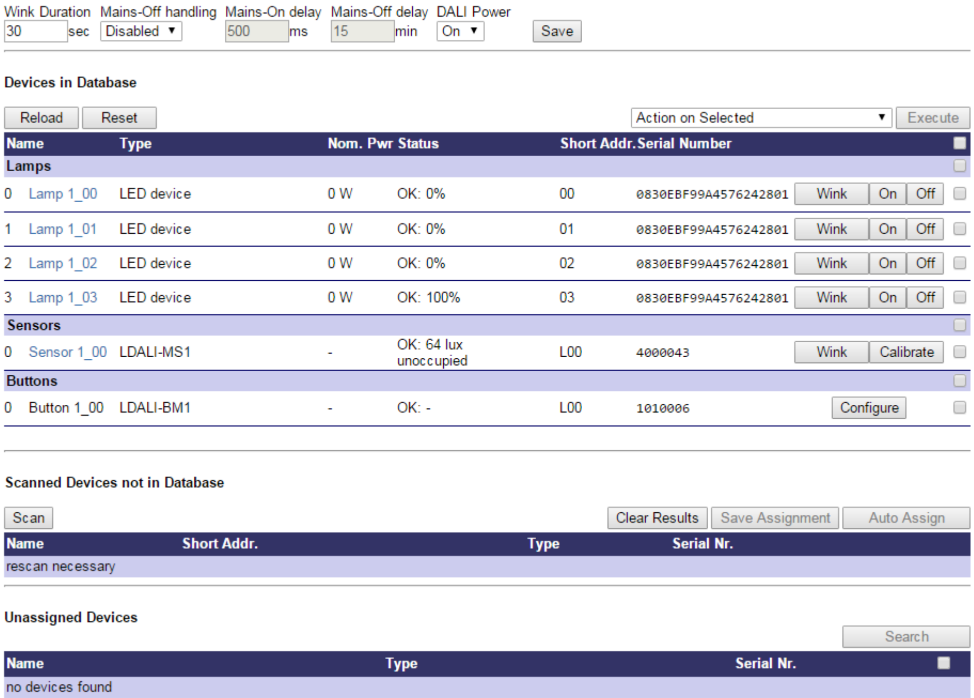


Figure 30: DALI Installation Web Interface: Device assignment done.

3.5.3 Grouping Lamps

Lamps which are assigned to a group can be controlled together by the corresponding LONMARK or BACnet group object. We will create two groups: One for the lamps near the window and one for the lamps near the corridor.

1. Click on **DALI Groups** in the Config menu. The DALI Groups page opens.

2. Again, selected the DALI channel by clicking on the different tabs at the top of the page labeled **Channel 1**, **Channel 2**, etc.

3. Add the two lamps near the window to group 0, the two lamps near the corridor to group 1 and all lamps to group 2 by checking the corresponding check boxes. Press the **Save** button. The result will look like in Figure 31.

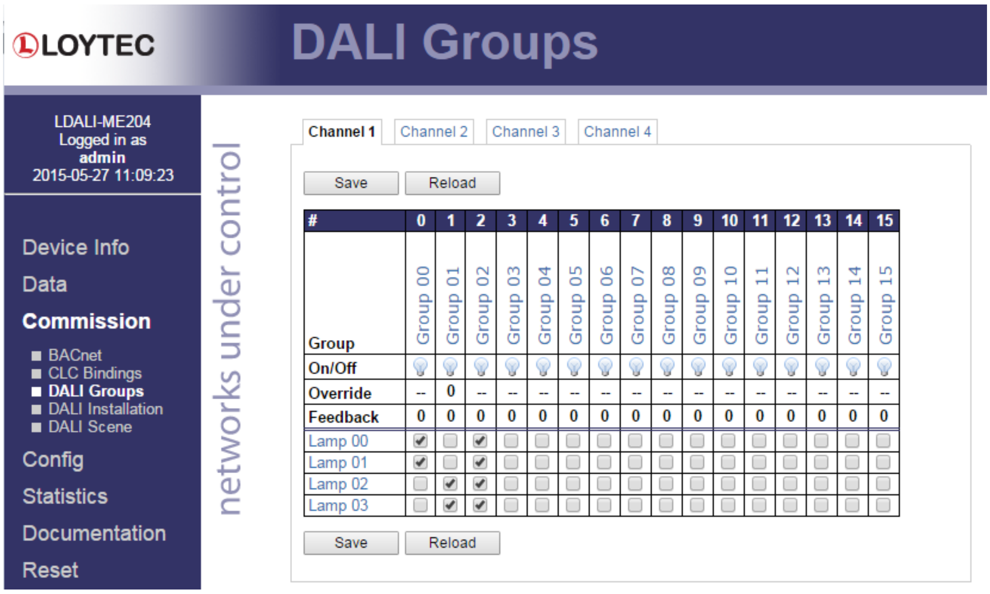


Figure 31: DALI Installation Web Interface: Group Configuration.

4. To test the group assignment, lights can be switched on and off by clicking on the bulb below the group name. Similar entering a value in the override row will dim the corresponding group. Entering – in the field will relinquish an active override.

3.5.4 Calibrate Light Sensor

The L-DALI device allows calibrating the light sensor under up to seven different light conditions to counter any non-linearity of the sensor. However, in many cases it is enough to calibrate the sensor with a single light condition which is near the setpoint. In this quick-start only one lux measurement is performed.

1. Go to the DALI Installation page and press the **Calibrate** button next to the sensor that is to be calibrated. The DALI sensor calibration page is shown in Figure 32.

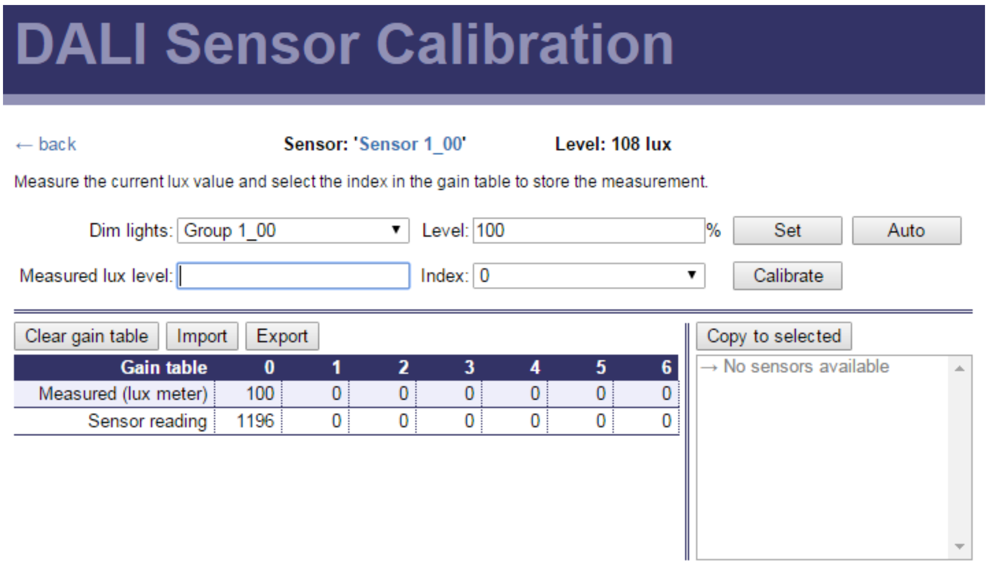


Figure 32: DALI Installation Web Interface: Sensor Calibration.

2. Measure the current lux level at the reference area (e.g. desk) using a luxmeter.

3. If possible dim artificial light in room until luxmeter show desired lux setpoint. To dim the light select the rooms DALI group in the **Dim lights** drop down box, enter a desired dim value in the field **Level** and press the **Set** button.

4. Enter the measured lux level in the input field and select index 0.

5. Press the **Calibrate** button.

3.5.5 Parameterize the DALI button

1. On the DALI Installation page click on the **Configure** button next to the button device that is to be configured. The dialog shown in Figure 33 appears.

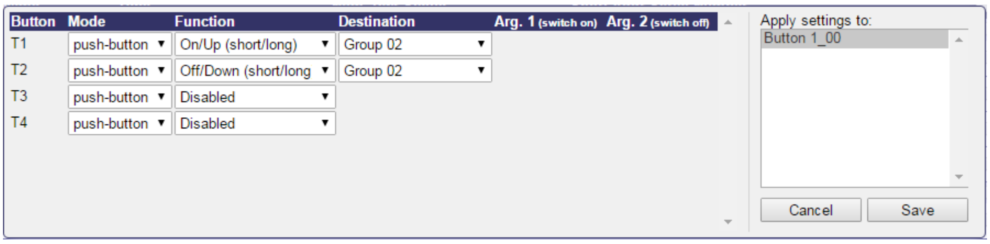


Figure 33: Configure DALI button functions.

2. Use the drop-down boxes to determine which function is executed and which group is controlled by which button input. In our example we use button T1 for “On/Up” and T2 for “Off/Down”, both controlling group 2. This will allow us to control all lights in the room together.

3. Click on **Save** to write the configuration to the button device.

3.5.6 Parameterize the Constant Light Controller

1. Go to the **Data Points** page and select the data point path of constant light controller application instance 0.

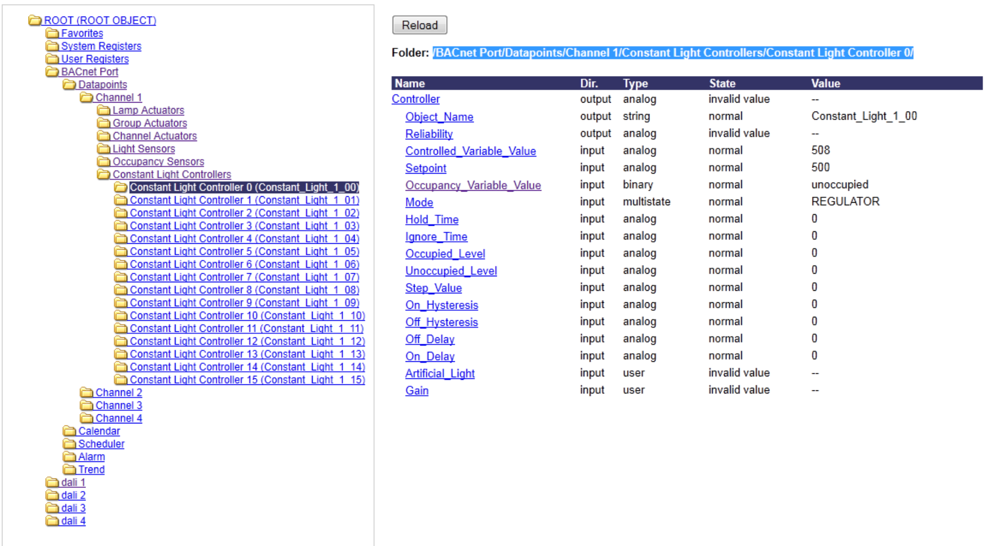


Figure 34: DALI Installation Web Interface: Parameters.

BACnet L-DALI: /BACnet Port/Datapoints/Channel 1/Constant Light Controllers/Constant Light Controller 0/

CEA-709 L-DALI: /CEA709 DALI Channel 1/Datapoints/Constant Light Controllers/Constant Light Controller 0/

The page will look like in Figure 34.

2. The parameters of the selected constant light controller are displayed on the right side. Change the constant light controller mode (parameter *nciClMode* or *Mode*) to REGULATOR. This parameter selects the operating mode. The REGULATOR mode is used if a light/occupancy sensor is installed which measures the indoor illumination.

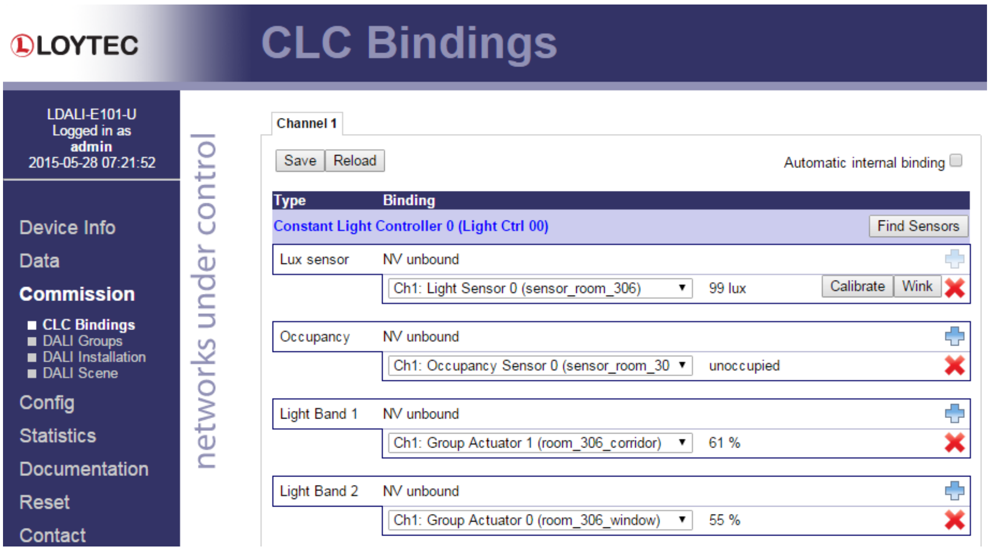


Figure 35: DALI Installation Web Interface: CLC Bindings.

3. Change the setpoint of the constant light controller (parameter *nciLuxSetpoint* or *Setpoint*) to the desired light level (e.g. 400 lux).

4. You can leave the default values for the remaining parameters. For a description of the constant light controller functionality and the parameters refer to Section 8.2.8 (CEA-709) and 8.3.5 (BACnet).

5. To determine which sensors (occupancy & lux) are used as inputs to and which light groups are controlled by a constant light controller instance go to the **CLC Bindings** page (see Figure 35).

6. Again, selected the DALI channel by clicking on the different tabs at the top of the page labeled **Channel 1**, **Channel 2**, etc.

7. Click on the **Add** to add an input or output. Use the drop down box to select a sensor (input) or a light group (output). In our example we use “sensor_306” for lux and occupancy sensor inputs, group “306_corridor” as output of the first (brighter) light band and group “306_window” as output for the second (darker) light band.

3.6 Configuration of BACnet Interface (LDALI-20X only)

3.6.1 Configure BACnet Interface

To allow integrating the L-DALI to a BACnet network a network wide unique device ID and device name must be configured. This is best done using the web interface:

1. Similar to the configuration of the IP address connect to the L-DALI using your Internet browser.

2. Click on **Config** and then **BACnet Config** in the left menu.

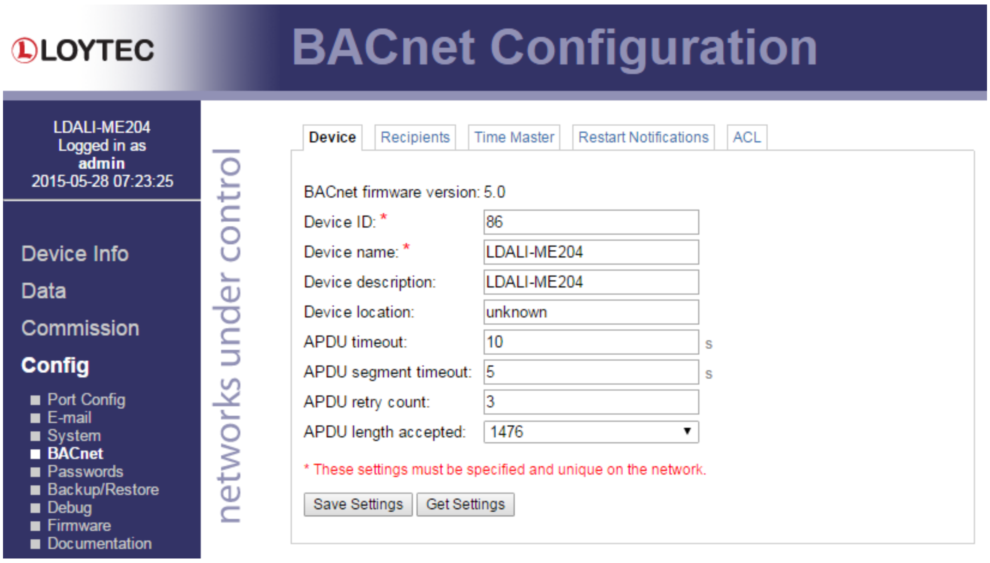


Figure 36: BACnet Device Configuration.

3. Enter a unique device ID and device name as shown in Figure 36.

For further details see Section 5.2.12.

4 Hardware Installation

4.1 Enclosure

4.1.1 LDALI-3E10X and LDALI-ME204

The LDALI-3E10X and LDALI-ME204 enclosure is 159 mm wide for DIN rail mounting, following DIN 43 880 (see Figure 37).

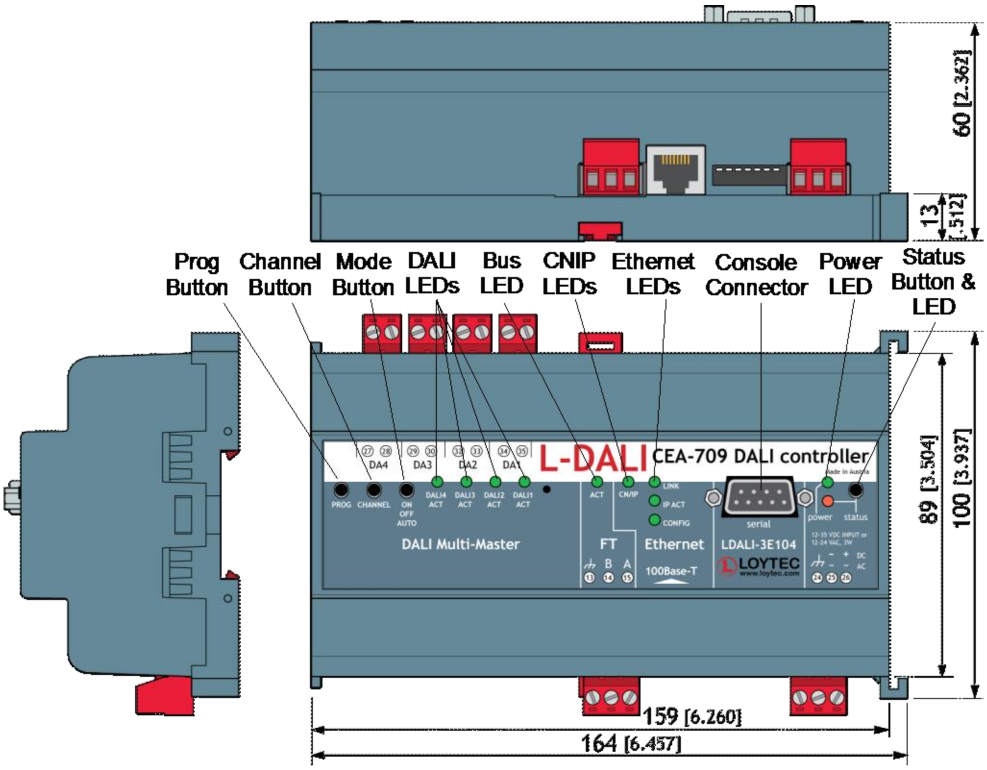


Figure 37: LDALI-3E10X and LDALI-ME204 Enclosure (dimensions in mm [inch]).

4.1.2 LDALI-E101-U, LDALI-3101-U and LDALI-E101-U

The LDALI E101 U, LDALI 3101 U and LDALI E101 U enclosure is 107 mm wide for DIN rail mounting, following DIN 43 880 (see Figure 38).

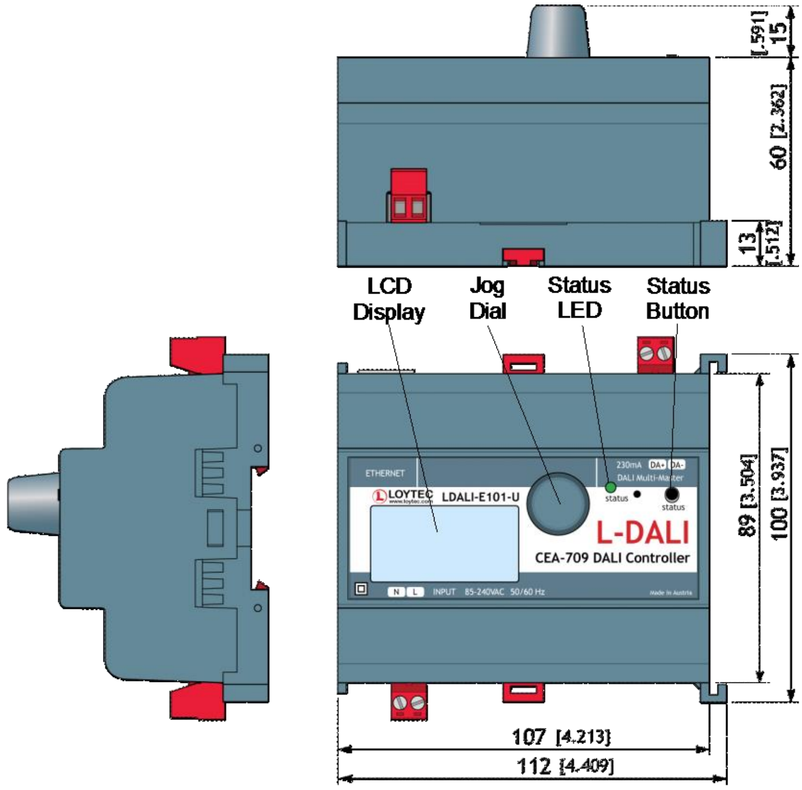


Figure 38: LDALI-E101-U, LDALI-3101-U and LDALI-E101-U Enclosure (dimensions in mm [inch]).

4.2 Product Label

The product label on the side of the L-DALI contains the following information:

- Order number with bar-code (e.g. LDALI-3E104),
- serial number with bar-code (Ser#),
- LDALI-10X only: unique node ID and virtual ID with bar-code for each DALI channel (NID1, VID1, NID2, VID2, etc.),
- Ethernet MAC ID with bar-code (MAC1).

Unless stated otherwise, all bar codes are encoded using “Code 128”. An additional label is also supplied with the L-DALI for documentation purposes. A virtual ID (VID) is a Node ID on the IP channel.

4.3 Mounting

The device comes prepared for mounting on DIN rails following DIN EN 50 022. The device can be mounted in any position. However, an installation place with proper airflow must be selected to ensure that the temperature of the L-DALI device does not exceed the specified range (see Chapter 13).

4.4 LED signals

4.4.1 LDALI-3E10X and LDALI-ME204

4.4.1.1 Power LED

The L-DALI power LED lights up green when power is supplied to terminals 24, 25, and 26.

4.4.1.2 Status LED

The L-DALI is equipped with a red status LED (see Figure 37). This LED is normally off. During boot-up the status LED is used to signal error conditions (red). If the fall-back image is executed the status LED flashes red once every second.

4.4.1.3 FT Activity LED (LDALI-10X only)

The FT port on the LDALI-3E10X has a three-color LED (green, red, and orange, see Figure 37). Table 2 shows different LED patterns of the port and their meaning.

Behavior	Description	Comment
GREEN flashing fast	Traffic	
GREEN flashing at 1Hz	L-DALI is unconfigured	
RED permanent	Port damaged	
RED flashing fast	Traffic with high amount of errors	
RED flashing at 1 Hz (all ports)	Firmware image corrupt Please upload new firmware	
ORANGE permanent	Port disabled	e.g. using LSD Tool

Table 2: CEA-709 Activity LED Patterns.

4.4.1.4 MSTP Activity LED (LDALI-20X only)

The MS/TP port on the LDALI-ME20X has a three-color MSTP Activity LED (see Figure 37). Table 3 shows the different LED patterns of the port and their meaning. A permanent color reflects a state. Flicker is for 25 ms when there is activity on the MS/TP data link layer.

Behavior	Description	Comment
GREEN permanently, flicker off	Multi-Master, token ok, flicker when traffic	Normal condition on a multi-master MS/TP network.
ORANGE flicker	Sole master, flicker when traffic	Normal condition on a single-master MS/TP network.
RED permanent, flicker GREEN	Token lost state, flicker when transmit attempt	Cable might be broken.
RED flash fast	Transmission or receive errors	This indicates bad cabling.

Table 3: MS/TP Activity LED Patterns.

4.4.1.5 Ethernet Link LED

The Ethernet Link LED lights up green whenever an Ethernet cable is plugged-in and a physical connection with a switch, hub, or PC can be established.

4.4.1.6 Ethernet Activity LED

The Ethernet Activity LED lights up green for 6 ms whenever a packet is transmitted or received or when a collision is detected on the network cable.

4.4.1.7 Ethernet Config LED

Currently the Ethernet Config LED has no function.

4.4.1.8 CN/IP LED

On the LDALI-3E10X the CNIP LED is a three color LED that indicates different operating states of the L-DALI's CEA-852 device.

Green: The CEA-852 device is fully functional and all CEA-852 configuration data (channel routing info, channel membership list, send list) are up-to-date.

Green flicker: If a valid CEA-709 packet is received or transmitted over the IP channel, the CNIP LED turns off for 50 ms. Only valid CEA-709 IP packets sent to the IP address of the L-DALI can be seen. Stale packets or packets not addressed to the L-DALI are not seen.

Yellow: The CEA-852 device is functional but some configuration data is not up-to-date (device cannot contact configuration server but has configuration data saved in Flash memory)

Red: The CEA-852 device is non-functional because it was rejected from the CEA-852 IP channel or shut-down itself due to an internal error condition.

Off: The CEA-852 device is non-functional because it has not been started. This can be the case if the L-DALI uses DHCP and it has not received a valid IP configuration (address) from the DHCP server.

Flashing Red at 1 Hz: The CEA-852 device is non-functional because it is started but has not been configured. Please add the device to a CEA-852 IP channel (register in configuration server).

Flashing green or orange at 1 Hz: The L-DALI's CEA-709 side of the gateway has not been commissioned yet. The color indicates the CEA-852 IP channel status as described above.

On the LDALI-ME20X the CNIP LED reflects the status of the BACnet/IP communication. It flashes green for 25 ms when BACnet packets are transmitted or received over the BACnet/IP interface.

4.4.1.9 DALI Activity LEDs

Each DALI interface on the L-DALI has a three color LED (green, red and orange). Table 4 shows different LED patterns and their meaning.

Behavior	Description
GREEN flashing	Traffic
RED flashing fast	Traffic with errors
RED permanent	No bus power-supply/bus-power supply failed
ORANGE permanent	Manual override to off or interface is selected
GREEN permanent	Manual override to on

Table 4: DALI Activity LED patterns.

4.4.2 LDALI-E101-U, LDALI-3101-U and LDALI-E101-U

4.4.2.1 Status LED

The L-DALI has a three color status LED (green, red and orange). Table 5 shows different LED patterns and their meaning.

Behavior	Description
GREEN flashing	DALI Traffic
RED flashing fast	DALI Traffic with errors
RED permanent	DALI bus-power supply failed/short
ORANGE permanent	Manual override to off
GREEN permanent	Manual override to on

Table 5: Status LED patterns.

4.4.3 Wink Action

If the L-DALI receives a wink command on any of its network ports, it shows a blink pattern on the

- CEA-709 or CNIP activity LEDs and the DALI activity LEDs (LDALI-3E10X and LDALI-ME204) or

- Status LED (LDALI-E101-U, LDALI-3101-U and LDALI-E101-U).

The LEDs turn green/orange/red (each 0.15 s). This pattern is repeated six times. After that, the activity LED of the DALI channel corresponding to the CEA-709 node on which the wink was received flashes orange six times. After that the L-DALI LEDs resume their normal behavior.

4.5 Buttons

4.5.1 Status Button

The L-DALI is equipped with a status button (see Figure 37). When pressing the status button shortly during normal operation of the L-DALI, it sends a “Service Pin Message” on every active CEA-709 node or a BACnet “I Am” message on all active BACnet data link layers respectively.

LDALI-10X only: Note that there is one CEA-709 node for each DALI channel and each has its own unique node ID (“Neuron ID”). Pressing the status button longer than 2 seconds will allow you to select the node to send out the “Service Pin Message” message: The DALI port LED of the currently selected node will light up orange. After 2 seconds the next available node will be selected. When the status button is released the “Service Pin Message” is sent out on the currently selected node.

As an alternative to pressing the status button, a service pin message can be sent via the Web interface (see Section 5.1).

The status button can also be used to switch the device back to factory default state. Press the service button and power-cycle the device. Keep the button pressed until the LEDs illuminate orange permanently. Release the button within five seconds from that time on to reset the device to factory defaults. Alternatively, the device can be switched back to factory defaults over the console UI (see Section 11.2.2).

4.5.2 DALI Mode Button

L-DALI without LCD display are equipped with a DALI mode button (“ON/OFF/AUTO”, see Figure 37). It is used to manually override the dim values of the attached DALI devices and constant light controller instances. Press it once and all DALI devices on the selected channels are switched on (“on mode”), press it again and all DALI devices on the selected channels are switched off (“off mode”), press it a third time and the selected channels go back to “auto mode”.

In the “on mode” and “off mode” the dim levels of the DALI devices and constant light controller instance cannot be changed via the CEA-709 or BACnet interface (“manual override”). In the “auto mode” the dim level of the DALI devices is controlled via the CEA-709 or BACnet interface respectively.

Channels are selected via the DALI Channel button (see Section 4.5.3). The current state of a DALI channel can be determined based on the corresponding DALI Activity LED (see Section 4.4.1.9).

The main purpose of the mode button is to test the wiring during installation of the DALI system.

4.5.3 DALI Channel Button

L-DALI without LCD display are equipped with a DALI Channel button (“CHANNEL”, see Figure 37). It is used to select a specific channel. All other functions which can be performed via the DALI button interface (e.g. select DALI mode, see Section 4.5.2) are applied only to the selected DALI channel(s).

By default all DALI channels are selected. If the DALI Channel button is pressed once the first channel is selected and the corresponding DALI Activity LED lights up orange. Now each time the button is pressed the next channel is selected. If the last DALI channel is selected and the button is pressed once again, all DALI channels are selected. If no button is pressed for more than 15 seconds, the current selection is canceled.

4.5.4 DALI Program Button

L-DALI without LCD display are equipped with a DALI Program button (“PROG”, see Figure 37). It is used to replace a broken ballast. When the button is pressed, the L-DALI scans the selected DALI channel for missing and unconfigured ballasts. If exactly one missing ballast and one unconfigured ballast are found on a channel, the unconfigured ballast is used to replace the missing ballast. That is, the unconfigured ballast is configured with the address and the configuration parameters of the missing ballast. If multiple missing ballasts or multiple unconfigured ballasts are found, the Web UI or LINX Configurator software must be used to replace the missing ballast(s) (see Section 5.3.2 and 7.5.1).

During the replace operation, the DALI Activity LED of the corresponding channel lights up orange. If the operation was successful, the LED lights up green for 0.5 seconds, if it failed, the LED lights up red for 0.5 seconds.

Which channels are selected can be controlled via the DALI Channel button (see Section 4.5.3).

4.6 LCD Display and Jog Dial

Device models with an LCD display can also be configured to their basic settings through jog dial navigation on the LCD UI. The main page of the LCD UI is shown in Figure 39. It displays the device’s IP address, hostname and CPU load.

Below are menu items. Turn the jog dial to navigate between menu items and press to enter a menu or go into selection mode. When in selection mode turn the jog dial to alter the value and press again to quit the selection.

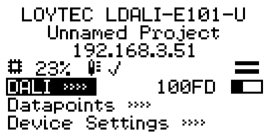


Figure 39: Main Screen of the LCD UI.

The **DALI »»** menu allows executing maintenance tasks on the DALI network. The corresponding sub-menu is shown in Figure 40.

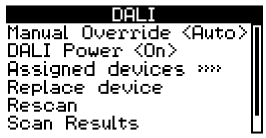


Figure 40: DALI Menu on LCD UI.

The menu item **Manual Override** allows to manually override the DALI lights to either **On** or **Off**. Ensure this is set to **Auto** for control of the DALI lights via one of the L-DALIs lighting applications or the BACnet/ LONMARK interface.

The menu item **DALI Power** allows disabling the internal DALI bus power supply.

The menu item **Assigned devices** lists all DALI devices configured on the DALI channel and allows to execute related maintenance functions (e.g. start/stop burn-in mode of selected lamps).

The menu item **Replace device** allows commissioning a new device, after it has been installed as a replacement for a broken DALI device. When using this function the complete DALI configuration of the replaced device (parameters, groups, etc.) is restored on the new device.

The **Datapoints »»** menu allows browsing through the data points on the device.

The **Device Settings »»** menu allows configuring basic device settings. Navigate to the **Device Mgmt »»** sub-menu, which is displayed in Figure 41.

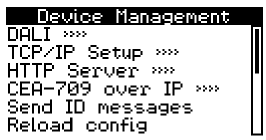


Figure 41: Device Management Menu on the LCD UI.

This menu gives you the following options for basic device configuration:

- **DALI:** This menu allows executing maintenance tasks on the DALI network (details see above).
- **TCP/IP Setup**: This menu allows configuring the device's IP address.
- **HTTP Server**: This menu allows to enable/disable the HTTP server and to configure its TCP port.

- **Send ID messages**: When selecting this menu, the device sends out service pin, BACnet I-Am, and identification broadcasts for finding the device in the L-Config tool on all applicable ports.

- **Reload config**: By choosing this menu, the device performs a quick restart by reloading its configuration only.

- **Reboot system**: By choosing this menu, the device performs a full reboot.

- **Clear DP config**: By choosing this menu, the user can clear the device's entire data point and DALI configuration. This is equivalent to the same Web UI function. The IP address as well as other settings needed to reach the device are not deleted.

- **Factory Defaults**: By choosing this menu, the user can reset the entire device to its factory default. All user configuration including IP addresses are cleared.

- **Remote Config**: When enabling this option, the LWEB-900 master device manager restores the last saved configuration to the discovered device, if it has no configuration yet. This feature is beneficial when replacing a device.

- **Change PIN**: Alter the default PIN to any 4-digit number to protect certain operations on the LCD UI. The user will be prompted to enter the PIN on protected areas.

- **Contrast**: Allows adjusting the LCD contrast.

- **Language**: Allow to select the language used for the LCD UI.

4.7 DIP Switch Settings

L-DALI models without LCD display are equipped with a DIP switch. The DIP switch assignment is shown in Table 6. Please leave all switches at default state.

DIP Switch #	Function	Factory Default
1	Must be OFF	OFF
2	Must be OFF	OFF
3	Must be OFF	OFF
4	Must be OFF	OFF
5	Must be OFF	OFF
6	Must be OFF	OFF
7	Must be OFF	OFF

Table 6: DIP Switch Settings for L-DALI.

4.8 Terminal Layout and Power Supply

The L-DALI provides screw terminals to connect to the network as well as to the power supply. The screw terminals can be used for wires of a maximum thickness of 1.5 mm^2/AWG12.

4.8.1 LDALI-3E10X

Terminal	Function
13	Earth Ground
14, 15	CEA-709 A, B of FT-10 Channel Port
24	Earth Ground
25, 26	Power Supply 12-35 VDC or 12-24 VAC ± 10% **Do not connect terminal 26 to earth ground!**
27, 28	DALI Channel 4
29, 30	DALI Channel 3
32, 33	DALI Channel 2
34, 35	DALI Channel 1

Table 7: LDALI-3E10X Terminals.

Note: *The LDALI-3E10X models do not provide the DALI bus power supply for any of the connected DALI channels. Thus, on each DALI channel a proper external DALI bus power supply must be provided. For this purpose LOYTEC recommends the use of LOYTEC's DALI bus power supply* ***LDALI-PWR4-U*** *or* ***LDALI-PWR2-U****, which is capable of providing the DALI bus power for the four or two DALI interfaces respectively. If some other DALI masters are connected to the DALI channel, these devices might generate the DALI bus power internally, in which case no additional bus power supply must be connected to this channel.*

4.8.2 LDALI-ME204

Terminal	Function
13	BACnet MS/TP / Modbus RS-485 Ground
14	BACnet MS/TP / Modbus RS-485 Non-Inverting Input
15	BACnet MS/TP / Modbus RS-485 Inverting Input
24	Earth Ground
25, 26	Power Supply 12-35 VDC or 12-24 VAC ± 10% **Do not connect terminal 26 to earth ground!**
27, 28	DALI Channel 4
29, 30	DALI Channel 3
32, 33	DALI Channel 2
34, 35	DALI Channel 1

Table 8: LDALI-ME204 Terminals.

Note: *The LDALI-ME204 models do not provide the DALI bus power supply for any of the connected DALI channels. Thus, on each DALI channel a proper external DALI bus power supply must be provided. For this purpose LOYTEC recommends the use of LOYTEC's DALI bus power supply* ***LDALI-PWR4-U*** *or* ***LDALI-PWR2-U****, which is capable of providing the DALI bus power for the four or two DALI interfaces respectively. If some other DALI masters are connected to the DALI channel, these devices might generate the DALI bus power internally, in which case no additional bus power supply must be connected to this channel.*

4.8.3 LDALI-E101-U and LDALI-E201-U

Terminal	Function
N, L	Power Supply 85-240 VAC 50/60 Hz
DA+, DA-	DALI Channel 1 (with internal DALI bus power 230mA)

Table 9: LDALI-E101-U and LDALI-E201-U Terminals.

Important: ***The LDALI-E101-U and LDALI-E201-U models come with an internal DALI bus power supply. Thus, no additional DALI bus power supply or any other device with an internal DALI bus power supply must be connected to the same DALI channel!***

4.8.4 LDALI-3101-U

Terminal	Function
A, B	CEA-709 A, B of FT-10 Channel Port
N, L	Power Supply 85-240 VAC 50/60 Hz
DA+, DA-	DALI Channel 1 (with internal DALI bus power 230mA)

Table 10: LDALI-3101-U Terminals.

Important: ***The LDALI-3101-U models come with an internal DALI bus power supply. Thus, no additional DALI bus power supply or any other device with an internal DALI bus power supply must be connected to the same DALI channel!***

4.9 Wiring

Connect the L-DALI to the DALI network as shown in Figure 5 and Figure 6. For easy configuration, it is recommended to always connect the L-DALI to the Ethernet network (if available).

4.9.1 LDALI-ME20X

If BACnet over MS/TP is enabled, the MS/TP network segment must be properly terminated with an LT-04 network terminator connected at each of the two ends of the segment media.

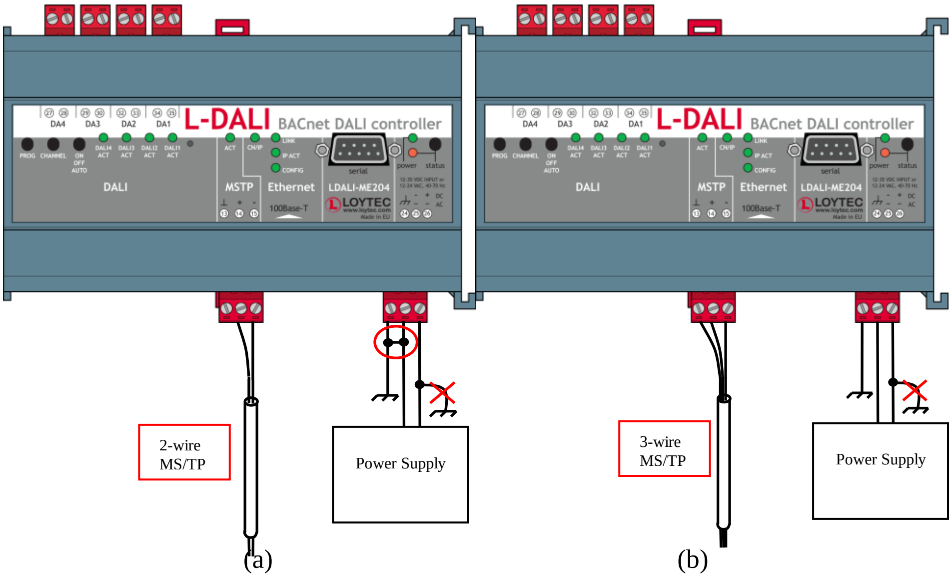


Figure 42: Connecting the LDALI-ME20X: (a) 2-wire MS/TP, (b) 3-wire MS/TP.

Important: ***When using 2-wire MS/TP, earth ground must be connected to both terminal 24 and 25 (see Figure 42a). Never connect terminal 26 to earth ground!***

4.9.2 LDALI-3E10X and LDALI-3101-U

Any CEA-709 FT-10 network segment connected to the L-DALI needs to be terminated according to the rules found in the specification of the transceiver (see Section 9.1).

Important: ***When using shielded network cables, only one side of the cable should be connected to earth ground. Thus, the shield must be connected to earth ground either at the L-DALI terminals or somewhere else in the network.***

5 Web Interface

L-DALI models with an Ethernet interface come with a built-in Web server and a Web interface to configure the device and extract statistics information. The Web interface allows configuring the IP settings, CEA-852 and CEA-709 settings, BACnet settings and managing the DALI devices. This interface is very simple to use and has an intuitive, self-explanatory user interface.

5.1 Device Information and Account Management

In a Web browser enter the default IP address 192.168.1.254 of the L-DALI. Note that if your PC has an IP address in a subnet other than 192.168.1.xxx you must open a command tool and enter the following route command to add a route to the L-DALI:

To Add a Route to the Device

1. Windows START → Run
2. Enter ‘cmd’ and click **Ok**.
3. In the command window enter the command line
   ```
   route add 192.168.1.254 %COMPUTERNAME%
   ```
4. Then open your Web browser and type in the default IP address 192.168.1.254.
5. The device information page should appear as shown in Figure 43.

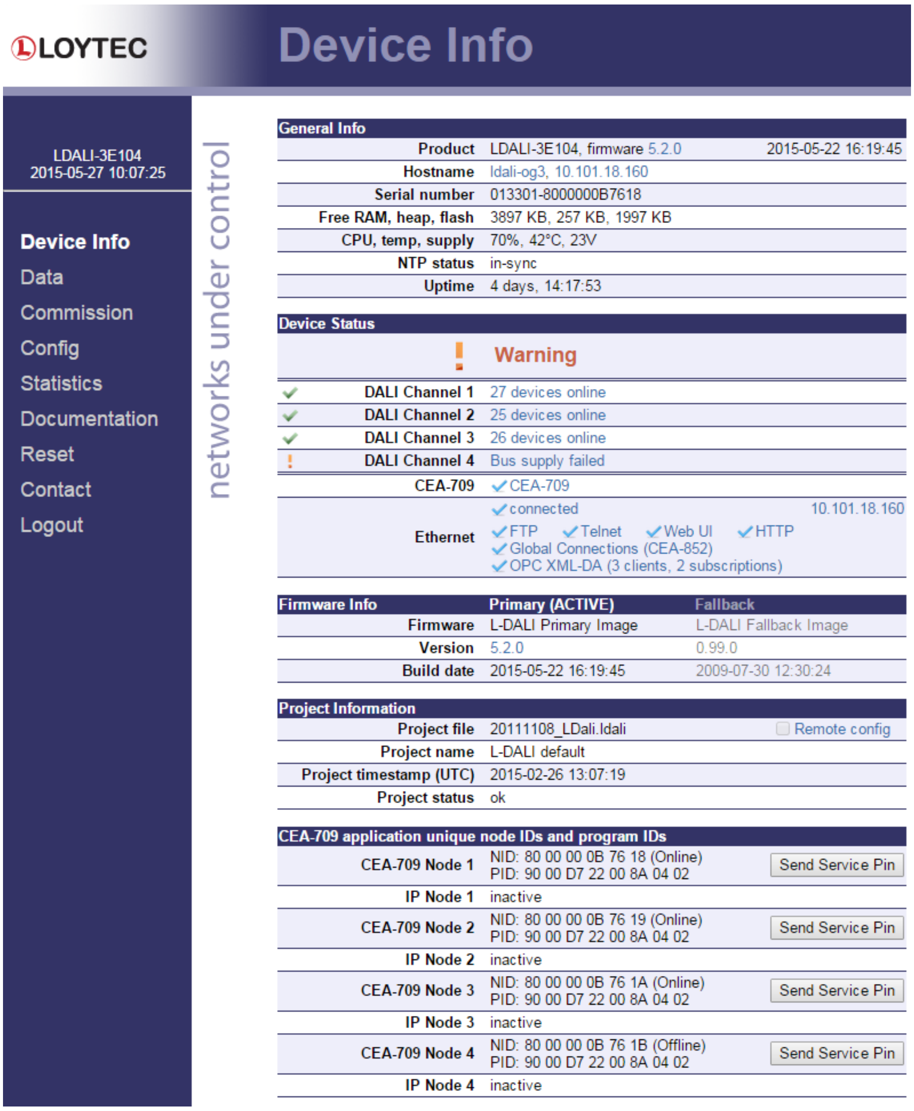


Figure 43: Device Information Page

The device information page shows some general information about the device in the **General Info** section. This includes the product model and the current firmware version. Below, it shows important operational parameters, such as free memory, CPU load, system temperature and supply voltage, time synchronization status and system uptime.

The **Device Status** section summarizes the status of the various ports and protocols on the device. The summary status is displayed as a green OK checkmark. If any of the interfaces, protocols or operational parameters are non-normal, a warning or error sign is shown instead. All items are links that lead directly to their configuration page.

Below the general status information more specific sections are displayed depending on the model. The **Firmware Info** provides version and build times of the primary and fallback firmware images installed on the device. The **Project Information** area shows details on the currently loaded data point configuration.

The **CEA-709 Application** section includes the unique node IDs ("Neuron IDs") of the CEA-709 network interfaces. This example shows the device information page for a LDALI-3E104 which has four DALI channels. The CEA-709 protocol is enabled and the CEA-852 (CEA-709 over IP) protocol is disabled. Therefore the web page shows the four applications **CEA-709 Node 1**, **2**, **3**, and **4** and displays their node IDs and program IDs. This page can also be used to send the CEA-709 service pin messages. This is a useful feature when commissioning the device, since it is not necessary to be on-site to press the status button.

Click through the menus on the left hand side to become familiar with the different screens. If you click on **Config** in the left menu you will be asked to enter the administrator password in order to make changes to the settings as shown in Figure 44. Enter the default administrator password ‘loytec4u’ and select **Login**.

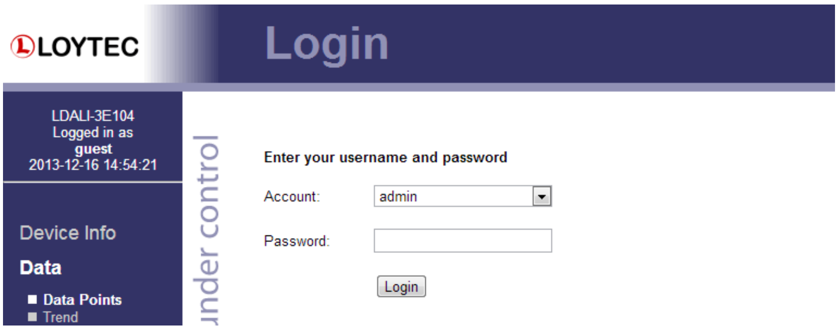


Figure 44: Enter ‘loytec4u’ as the default administrator password.

The Config menu opens. Click on **Passwords** in the Config menu, which opens the password configuration page as shown in Figure 45. The L-DALI has three user accounts: (1) **guest** allows the user to view certain information only, e.g., the device info page. By default the guest user has no password. (2) **operator** is able to read more sensible information such as calendar data. (3) **admin** has full access to the L-DALI and can make changes to its configuration. Note that the user accounts are also used to log on to the FTP and Telnet server.

Figure 45: Password Configuration Screen

Please change the administrator password in order to protect yourself from unwanted configuration changes by anyone else. To do so, select the **admin** account in the drop-down box and enter the new password. If the administrator password is left empty, password protection is turned off and everyone can access the L-DALI without entering a password. Click on **Change password** to activate the change.

5.2 Device Configuration

The device configuration pages allow viewing and changing the device settings of the L-DALI. Here are some general rules for setting IP addresses, port numbers, and time values:

- An empty IP address field disables the entry.

- An empty port number field sets the default port number.
- An empty time value field disables the time setting.

5.2.1 Port Configuration

This menu allows configuring the device's communications ports. For each communication port, which is available on the device and shown on the label (Port 1, CEA-709, Ethernet), a corresponding configuration tab is provided by the Web UI. An example is shown in Figure 46. Each port tab contains a selection of available communication protocols. By selecting a checkbox or radio button the various protocols can be enabled or disabled on the communication port. Some ports allow exclusive protocol activation only, other ports (e.g., the Ethernet port) allow multiple protocols bound to that port.

When selecting a protocol on a communication port, the protocol's communication parameters are displayed in a box on the right-hand side. To save the settings of the currently opened protocol, click the **Save Settings** button. Pressing **Get Settings** retrieves the current settings from the device.

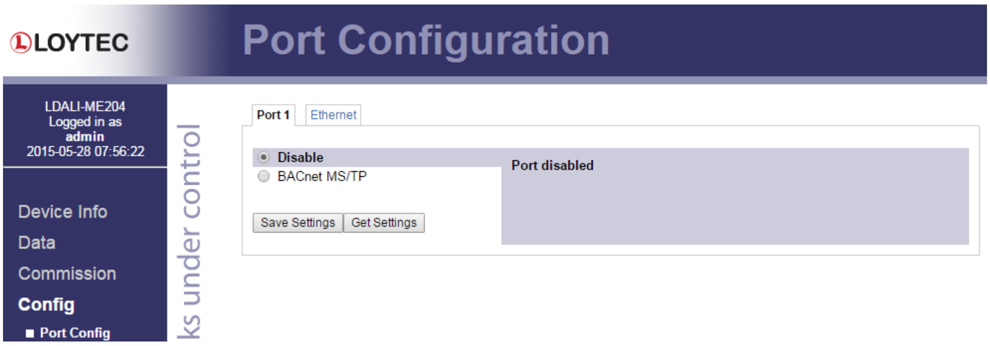


Figure 46: Port Configuration Page.

5.2.2 IP Configuration

The TCP/IP configuration is done under the Ethernet port tab as shown in Figure 47. The mandatory IP settings, which are needed to operate the device, are marked with a red asterisk (IP address, netmask, gateway). The **Enable DHCP** checkbox switches between manual entry of the IP address, netmask, and gateway address, and automatic configuration from a DHCP server.

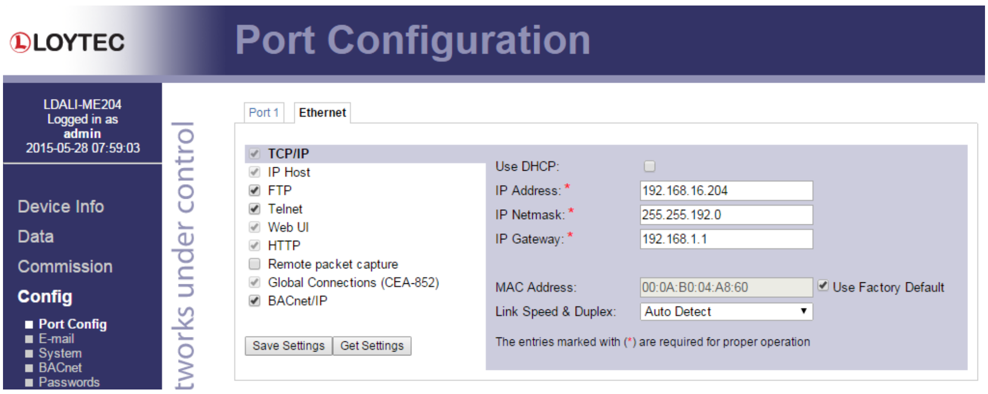


Figure 47: IP Configuration Page.

The device comes configured with a unique MAC address. This address can be changed in order to clone the MAC address of another device. Please contact your system administrator to avoid MAC address conflicts.

If the device is operated with a 10 Mbit/s-only hub, the link speed should be switched from **Auto Detect** to **10Mbps/Half-Duplex**. With modern 100/10 Mbit/s switches, this setting can be left at its default.

The settings for DNS and NTP servers should be made in the IP host settings (see Section 5.2.3). In case an IP interface runs DHCP, the DNS and NTP addresses supplied by DHCP can be seen here. Models with one Ethernet port only do not have these settings here.

Other standard protocols that are bound to the Ethernet interface are FTP, Telnet, and HTTP (Web server). By deselecting the checkbox, those protocols can be individually disabled. The standard UDP/TCP ports can be changed in the respective protocol settings. An example for the FTP server is shown for FTP in Figure 48. The FTP server is used for instance to update the firmware (see Section 10.1) or to upload a new data point configuration. Note that HTTP for the Web server can only be disabled on the console interface or by using the device configuration of the Configurator.

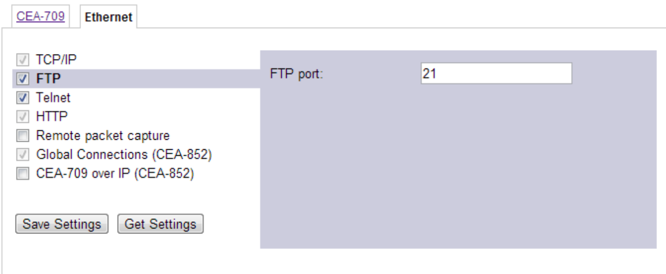


Figure 48: FTP server configuration on the Ethernet port.

5.2.3 IP Host Configuration

The L-DALI models, which provide more than one IP interface possess a separate **IP Host** tab for editing all common host settings. These settings affect all IP interfaces on the entire device. On models with a single Ethernet port, the IP Host settings appear directly on the Ethernet tab as shown in Figure 49.

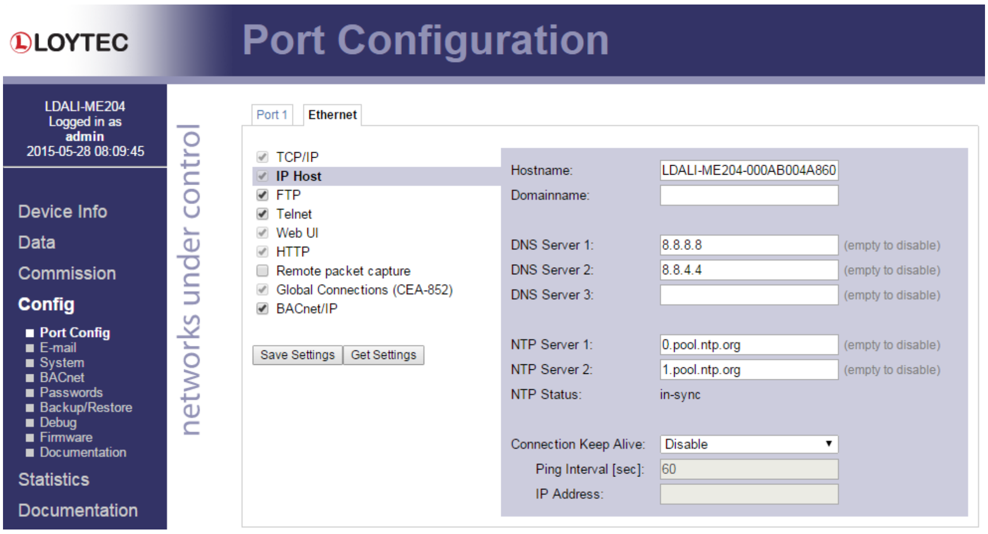


Figure 49: Setting on the IP Host settings.

Hostname and **Domainname** are optional entries and can be left empty. For some DHCP configurations it may be necessary to enter a hostname. Please contact your system administrator on how to configure DHCP to acquire an IP address.

If the device possesses more than one IP interface the **Default Gateway** setting defines the gateway of a given IP interface, which is going to route all non-local network traffic. One of the existing IP interfaces with a separate network must be selected here.

Up to three **DNS Servers** can be defined on this page. These DNS servers will be contacted by all services on any of the IP interfaces for name resolution. In case the DNS servers are supplied by DHCP running one of the IP interfaces, change the setting **Use DNS servers from** to point to that interface.

The device can be configured to synchronize its clock with NTP time. Enter the IP address of a primary and, optionally, a secondary NTP server. The device will use NTP as a time source if the time sync source in the system configuration page is set to **NTP** (see Section 5.2.11). The field **NTP status** below the NTP server settings displays the current NTP synchronization status (**out-of-sync**, or **in-sync**). The settings made here apply to all IP interfaces. In case the NTP servers are supplied by DHCP running one of the IP interfaces, change the setting **Use NTP servers from** to point to that interface.

The **Connection Keep Alive** feature allows the device to automatically ping other devices on the IP network in order to maintain an IP connection that might be automatically disconnected after a specific period of time (e.g. DSL routers automatically disconnect if no activity is detected). When enabled choose one of the options Auto IP or Custom IP.

If auto IP mode is selected and the device has a CEA-852 configuration server, a ping message is sent to all CEA-852 devices in the channel list of the configuration server. If the configuration server is disabled on this device a ping message is sent to the configuration server for the IP-852 channel, if one is known. If custom IP is selected, one specific IP address can be configured as the ping destination.

5.2.4 CEA-852 Device Configuration (LDALI-10X only)

The CEA-852 protocol is only available on the Ethernet port. To enable CEA-852 on the device, select the **CEA-852 (CEA-709 over IP)** checkbox on the **Ethernet** tab of the port configuration page. Note, that enabling the CEA-852 protocol will disable the CEA-709 protocol and vice versa.

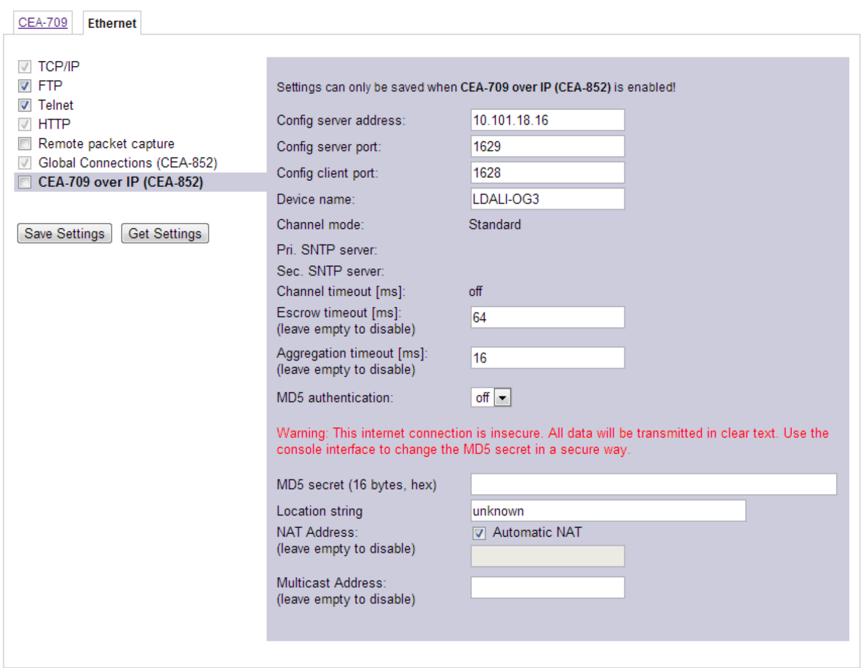


Figure 50: CEA-852 Device Configuration Page.

The CEA-852 protocol settings are displayed in the settings box on the right-hand side as shown in Figure 50. Typically, the device is added to an IP channel by entering the relevant information on a configuration server. The configuration server then contacts the CEA-852 device of the L-DALI and sends its configuration.

The field **Config server address** and **Config server port** display the IP address and port of the configuration server, which manages the L-DALI and the IP channel. The field **Config client port** represents the IP port of the L-DALI's CEA-852 device. This setting should be left at its default (1628) unless there are more than one CEA-852 devices operating behind a single NAT router. Please refer to the L-IP User Manual [1] to learn more about NAT configuration.

In the field **Device name** the user can enter a descriptive name for the L-DALI, which will appear in the IP channel to identify this device. You can enter a device name with up to 15 characters. It is recommended to use unique device names throughout the IP channel.

The **Channel mode** field reflects the current channel mode of the CEA-852 device. It is configured by the configuration server. If there are any two devices in the channel which use the same IP address but different ports (e.g., multiple L-DALI behind one NAT router) the channel switches to **Extended NAT mode**. Please refer to the L-IP User Manual [1] to learn more about configuring the Extended NAT mode in the configuration server.

The configuration server sets the **SNTP server** addresses and the **Channel timeout**.

The filed **Escrow timeout** defines how long the CEA-852 device on the L-DALI waits for out-of-sequence CEA-852 data packets before they are discarded. Please enter the time in ms or '0' to disable escrowing. The maximum time is 255 ms.

The field **Aggregation timeout** defines the time interval in which multiple CEA-709 packets are combined into a single CEA-852 data packet. Please enter the time in ms or '0' to disable aggregation. The maximum time is 255 ms. Note that disabling aggregation will negatively affect the performance of the CEA-852 device of the L-DALI.

The field **MD5 authentication** enables or disables MD5 authentication. Note that MD5 authentication cannot be used together with the Echelon's *i*.LON 1000 since the *i*.LON 1000 is not fully compliant with the CEA-852 authentication method. MD5 can be used with the *i*.LON 600. In the following field **MD5 secret** enter the 16-byte MD5 secret. Note that for security purposes the active MD5 secret is not displayed. You may enter the 16 bytes as one string or with spaces between each byte, e.g., 00 11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF.

Also note that entering the MD5 secret on the Web interface may pose a security risk. Since the information is transmitted over the network it can be subject for eavesdroppers on the line. It is recommended to use a cross-over cable.

In the field **Location string** the user can enter a descriptive test which identifies the physical location of the L-DALI. A location string can have a maximum length of 255 characters. This is optional and for informational purposes only.

If the CEA-852 device on the L-DALI is used behind a NAT router, the public IP address of the NAT router or firewall must be known. To automatically detect the NAT address leave the **Auto-NAT** checkmark enabled.

The **Multicast Address** field allows the user to add the CEA-852 device of the L-DALI into a multi-cast group for the CEA-852 IP channel. Enter the channel's IP multi-cast address here. Please contact your system administrator on how to obtain a valid multi-cast address. To learn when it is beneficial to use multi-cast addresses in your channel please refer to the L-IP User Manual [1].

5.2.5 Global Connections Configuration

The CEA-852 device used for global connections (see Section 6.3.4) can be configured on the Ethernet port. The global connections function is always enabled on the CEA-852 device. This is indicated by the checkbox **Global Connections (CEA-852)** on the **Ethernet** tab of the port configuration page as shown in Figure 51. The settings are shared with the **CEA-709 over IP** settings, if that protocol is enabled. Otherwise, the CEA-852 device is configured on this tab as described in Section 5.2.4.

If the user does not want to share the CEA-709 over IP channel with his global connections, the checkbox **Use separate IP channel for global connections** can be activated. In this case, a separate CEA-852 device is configured on this tab as described in Section 5.2.4. Note, that this CEA-852 device will need a different port number, e.g. 1630. In this case, also a separate configuration server must be used.

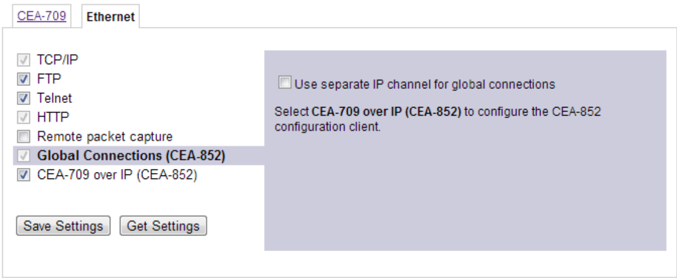


Figure 51: Global Connections Configuration Page.

5.2.6 BACnet/IP Configuration (LDALI-20X only)

The BACnet/IP protocol is available on the Ethernet port. To enable BACnet/IP on the device, select the BACnet/IP checkbox on the Ethernet tab of the port configuration page. Please note that the BACnet MS/TP protocol will be disabled.

The BACnet/IP protocol settings are displayed in the settings box on the right-hand side as shown in Figure 52. If the BACnet/IP network uses a non-default UDP port number other than 47808/0xBAC0, enter this port in the **BACnet/IP port** field. Enter '0' in this field for switching back to the default setting.

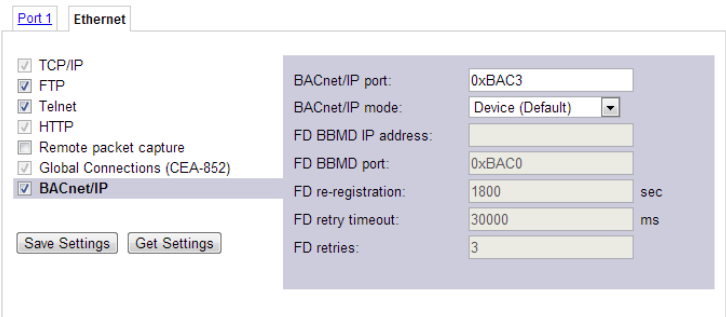


Figure 52: BACnet/IP Configuration.

In the field **BACnet/IP mode** the operation mode of the device is selected:

- **Device** (Default): In this mode the device operates as a regular BACnet/IP device on the local network without other advanced features.

- **Foreign Device** (FD): In this mode, the device registers at an existing BBMD in the BACnet/IP network as a foreign device. It is used, if the device is located as a single BACnet/IP device on a remote IP subnet or behind a NAT router. If operated as a foreign device behind a NAT router, port forwarding to the BACnet/IP port (UDP, default port 0xBAC0) and optionally to the Web server and FTP server port (TCP, default port 80 and 21) must be setup in the NAT router. If foreign device is selected, the following, additional settings must be made:

- **FD BBMD IP address** and **FD BBMD port**: IP address and port of the remote BBMD the device registers at as a foreign device.
- **FD re-registration**: A foreign device must periodically re-register at a BBMD. Here you can setup the corresponding interval. The default is 1800 seconds.
- **FD retry timeout** and **FD retries**: Here you can specify the behavior, if registration does not work instantly. These values should be left at default: 30000ms / 3 retries.

5.2.7 VNC Configuration

LOYTEC devices equipped with an LCD display also provide remote access over Ethernet to the LCD display. The VNC protocol is used for this purpose and the device implements a VNC server for exposing the display. The VNC server is by default disabled on the device. On the PC a VNC client needs to be installed. Using the default settings, the VNC client connects to port 5900 of the device. The password is ‘loytec4u’.

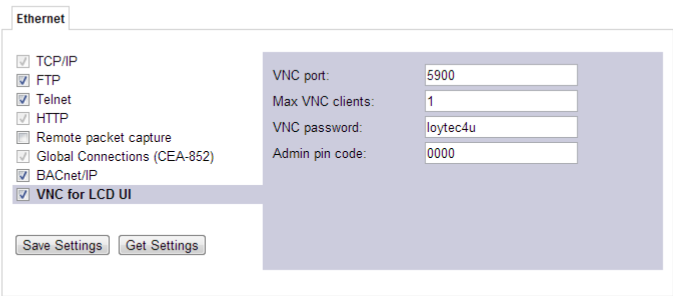


Figure 53: VNC Configuration.

The VNC server can be configured on the **Ethernet** tab of the port configuration. To turn on the VNC server, enable the **VNC for LCD UI** checkbox. The VNC protocol settings are displayed in the settings box on the right-hand side as shown in Figure 53. The **VNC port** and **VNC password** can be changed. As a default, only one VNC client may connect. This limit may be increased in **Max VNC clients**. In order to protect changes made on the LCD UI over VNC with a PIN code, the **Admin PIN code** can be configured. To disable PIN protection, enter ‘0000’.

5.2.8 CEA-709 Configuration (LDALI-10X only)

The CEA-709 protocol can be enabled/disabled as shown in Figure 54. The protocol settings box on the right-hand side displays the current transceiver settings. Note, that enabling the CEA-709 protocol will disable the CEA-852 protocol and vice versa.

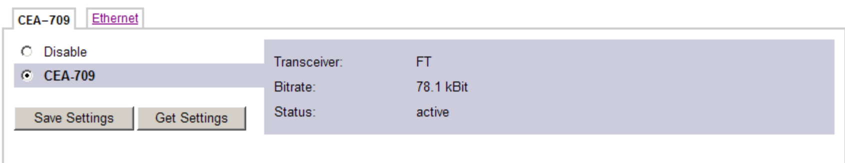


Figure 54: CEA-709 Configuration Page.

5.2.9 MS/TP Configuration (LDALI-ME204 only)

The BACnet MS/TP protocol can be enabled on the device's port Port1. To enable it, click the **BACnet MS/TP** radio button as shown in Figure 55. By default the BACnet MS/TP port is disabled.

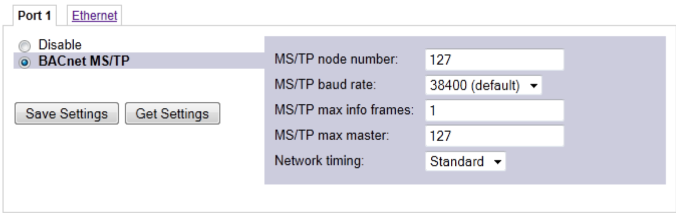


Figure 55: MS/TP Configuration.

The MS/TP protocol settings are displayed in the settings box on the right-hand side as shown in Figure 55. Mandatory settings are the **MS/TP node number** and the **MS/TP baud rate.** The MS/TP node number determines the physical address of the device on the MS/TP channel and must be in the range from '0' to the number configured with the **MS/TP max master** configuration option. It must be unique within the MS/TP channel. The Baud rate on the MS/TP channel can be set to 9600, 19200, and 38400 Baud.

Important: ***All masters on the MS/TP channel must have the same setting for MS/TP max master. Decreasing the default value 127 of MS/TP max master may reduce latency on the MS/TP bus.***

It is strongly recommended to leave the **MS/TP max info frames** and the **MS/TP max master** configuration options at their default settings. In any case the **MS/TP max master** number must be high enough to include the highest MS/TP node number of all masters on the channel. Slave devices may have a higher MS/TP node number than **MS/TP max master**.

To operate with slow devices on the MS/TP network set the **Network Timing** option to slow. This increases a number of timeouts, which is needed by some devices, but slows down network communication. If communication problems occur in standard mode, try setting the slow mode. For fine-tuning other parameters please refer to Section 9.2.

Note, that BACnet/IP will be disabled, when the MS/TP port is enabled.

5.2.10 E-Mail Configuration

The Web interface provides the e-mail configuration page to set up an e-mail account, which is used to send e-mails. The content and time when e-mails are sent is configured through the Configurator software (see Section 7.7.11). The e-mail configuration page is shown in Figure 56.

In the field for the outgoing e-mail server, enter the SMTP server of your Internet provider. Typically, the SMTP server port can be left at 25. In the field **Source e-mail address**, enter the e-mail address of the L-DALI's e-mail account. In the field **Source e-mail sender name** enter a name that the e-mail will display as the source name. Note, that only ASCII characters are allowed in the name. If replies shall be sent to another e-mail address, specify this in the **Reply e-mail address**.

If the provider's SMTP server requires authentication, enter the required user name and password. Note, that username/password is supported as well as SSL/TLS authentication (e.g., for using Hotmail, gmail, or Yahoo!). For older versions of secure connection check the SMTPS check box.

To verify the e-mail configuration, reboot the device to let the changes take effect and return to the e-mail configuration page. Then press one of the **Send Test E-Mail** buttons. Note, that a DNS server must be configured in the IP host settings (see Section 5.2.3) to resolve the e-mail server host name. The Web UI displays a warning message at the top of the page, if the DNS configuration is missing.

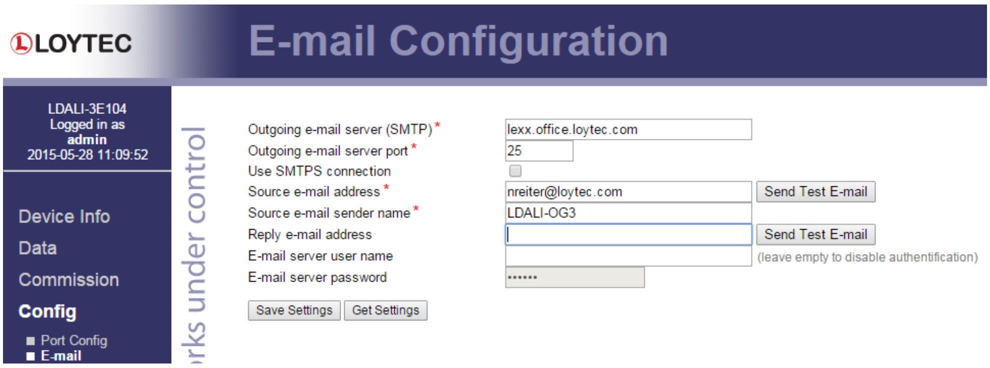


Figure 56: E-mail Configuration

5.2.11 System Configuration

The system configuration page is shown in Figure 57. This page allows configuring the device's system time. The time sync source can be set to **auto**, **manual**, **NTP**, or **LONMARK**. In the **auto** mode, the device switches to the first external time source that is discovered. Possible external time sources are NTP and LONMARK. The option **manual** allows setting the time manually in the fields **Local Time** and **Local Date**. In **manual** mode, the device does not switch to an external time source. Note, that if **NTP** is selected, the NTP servers have to be configured on the IP-Host configuration page (see Section 5.2.3).

In order to use BACnet as the time source, a BACnet device (time master) must be configured to distribute time synchronization. For doing so, the BACnet address of the devices, which shall be synchronized, must be added to the device object of the BACnet time master (see Section 8.3.2.7). The device synchronizes automatically as soon as it is contacted by the BACnet time master.

The time zone offset must be defined independently of the time source. It is specified as the offset to GMT in hours and minutes (e.g., Vienna/Austria is +01:00, New York/U.S.A. is -06:00). For setting the daylight saving time (DST) pre-defined choices are offered for Europe and U.S.A./Canada. DST can be switched off completely by choosing **none** or set manually for other regions. In that case, start and end date of DST must be entered in the fields below.

The next section on the page allows configuring the earth position of the L-DALI. This setting defines the longitude, latitude and elevation of the device. The latitude and longitude are entered as degrees, minutes, and seconds. The altitude (or elevation) is entered in meters from sea level. This setting is used for an astronomical clock. For fixed locations such as a building, the position can be entered on this page. For moving locations, this setting can be updated over the network using the network variable *nciEarthPos* (see Section 8.2.1).

The CSV delimiter specifies what character is used as delimiter when downloading a CSV file (e.g. trend log file) from the L-DALI.

In **Remote Configuration** it can be configured, whether a replaced device shall automatically request its configuration from an LWEB-900 server. This remote configuration request is sent only, if the device does not have a data point configuration.

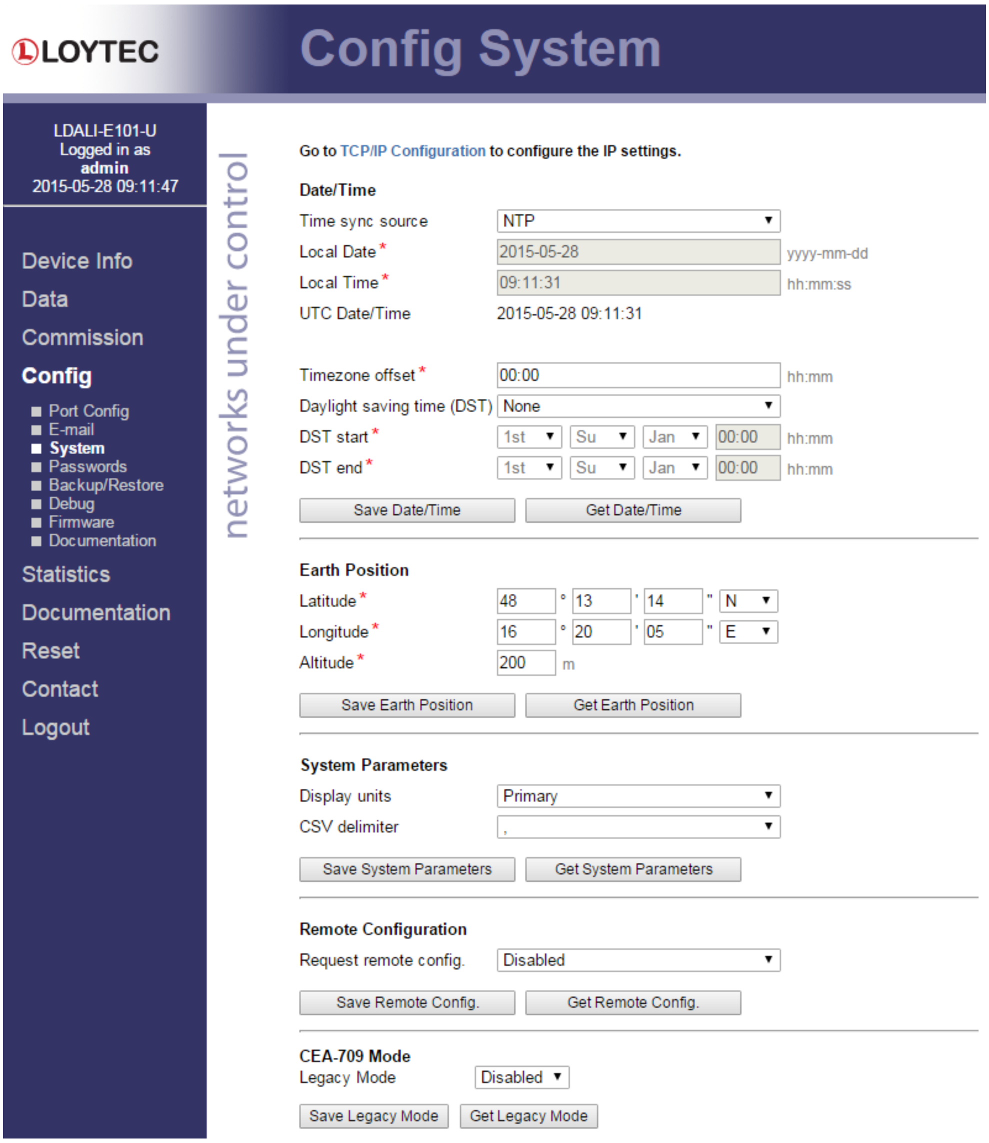


Figure 57: System Configuration Page

Enable the legacy mode if this is required by your network management tool (see Section 7.3.4, LDALI-10X only).

5.2.12 BACnet Configuration (LDALI-20X only)

Figure 58 shows the BACnet device configuration page. This configuration page allows setting the **Device ID**, which is the instance part of the Object_Identifier property of the BACnet Device object. The field **Device name** holds the name of the BACnet device object (property Object_Name).

Important! ***The device ID and device name must be unique within the BACnet internetwork.***

Further, the description and location can be configured. These configuration items correspond to the properties Description, and Location respectively of the BACnet Device object. For tuning BACnet application timing parameters, set **APDU timeout**, **APDU segment timeout**, and **APDU retry count**. The timeout values are entered in seconds allowing decimal notation, e.g. "7.5".

On the settings for BACnet/IP refer to Section 5.2.6. For configuring the MS/TP data link refer to Section 5.2.9.

Note:	*If this page displays the message "Device communication is disabled via BACnet network!" the device has been externally disabled. Reboot the device to activate communication again.*

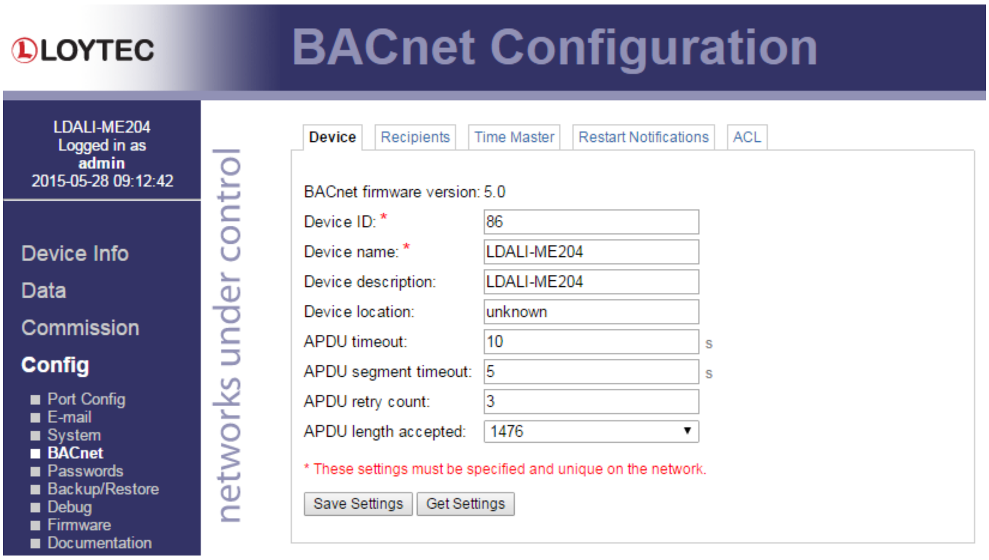


Figure 58: BACnet Device Configuration.

5.2.13 BACnet Recipients (LDALI-20X only)

BACnet notification class (NC) objects have a recipient list. Other BACnet devices, that shall act as alarm recipients and receive alarm notifications need to be added to the recipient list of the respective notification class. The **Recipients** tab of the **BACnet Config** menu can be used to view currently subscribed recipients as shown in Figure 59. Recipient entries can be modified and deleted from the list. It is also possible to add new recipients to the list with the **Add Recipient** button. This way it is possible to integrate third-party devices as alarm recipients without an OWS.

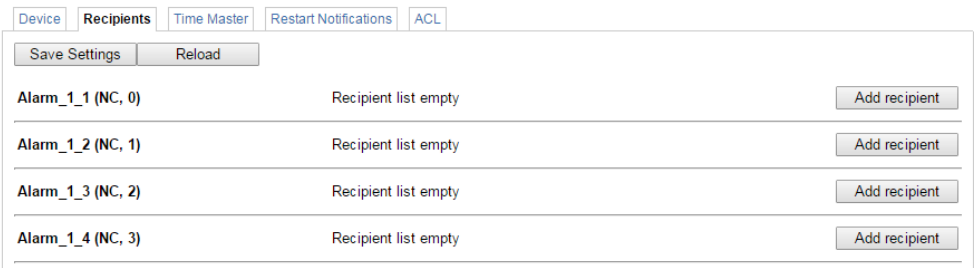


Figure 59: BACnet Recipients Configuration.

5.2.14 BACnet Time Master (LDALI-20X only)

The BACnet time master function relies on a list of time recipients. The **Time Master** tab of the **BACnet Config** Web page (see Figure 60) allows adding and removing time recipients of two classes: UTC time sync recipients, and time sync recipients (receiving local time). The time sync interval can also be configured on this tab. See Section 8.3.2.7 for more information on the settings for time sync interval, interval offset and align intervals.

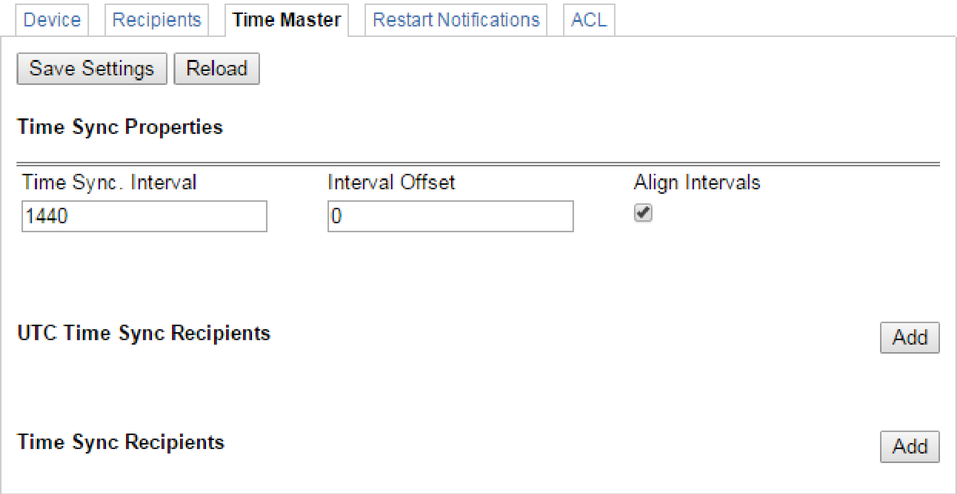


Figure 60: BACnet Time Master Configuration.

5.2.15 BACnet Restart Notifications (LDALI-20X only)

The device can be configured to send out a BACnet restart notification every time the device is starting. The list of recipients for this notification can be configured on the **BACnet Restart Notifications** tab. Click the **Add** button for adding a new line to the list. Then choose the recipient type from the drop-down box; it can be etiher a device instance number or a BACnet address. For broadcasting the restart notification, choose **Addr** and type in an asterisk '*' for a global broadcast or prefix it with a destination subnet, e.g. '12:*' as shown in Figure 61. Then click **Save Settings** to store the new recipient.

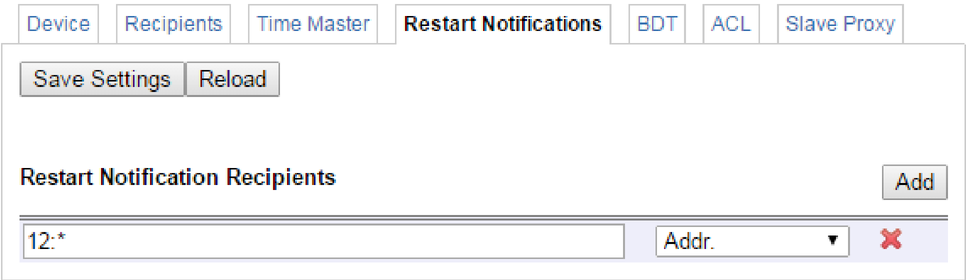


Figure 61: Broadcast BACnet restart notifications to a subnet.

5.2.16 BACnet ACL (Access Control List, LDALI-20X only)

The device provides a feature in BACnet/IP to filter packets from certain sources on the BACnet/IP network. This feature is based on an access control list (ACL). An example of the ACL configuration is shown in Figure 62.

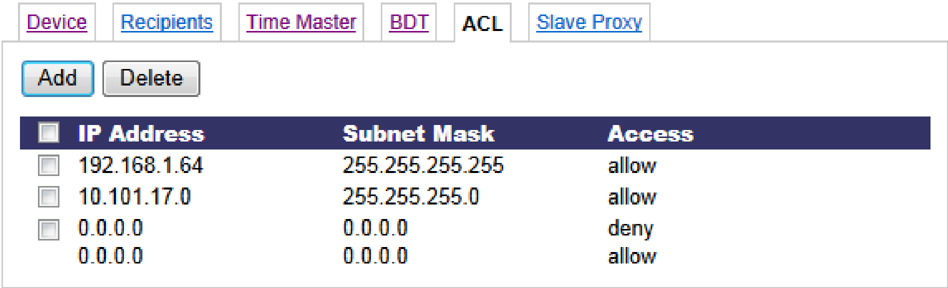


Figure 62: BACnet Access Control List (ACL).

The user can add and delete entries to the ACL. Each entry contains a source specification, which consists of an IP address and an IP mask, and an action (allow or deny). For specifying single hosts use the IP address and the mask '255.255.255.255'. For an address range specify an appropriate mask. For example use '10.101.17.0' and the mask '255.255.255.0' to specify all hosts with IP addresses '10.101.17.xxx'. To specify all IP addresses use '0.0.0.0' and the mask '0.0.0.0'.

The ACL is evaluated from specific host entries down to wider ranges. When adding new entries the ACL is automatically sorted, having the most precise definition at the top and the most general one at the bottom. The default behavior is to allow packets from all IP addresses. This is also the default entry in the ACL.

The example shown in Figure 62 specifies the following behavior for BACnet/IP:

1. Allow packets from the device 192.168.1.64

2. Otherwise allow packets from devices in the network 10.101.17.xxx

3. Otherwise deny packets from all (other) IP addresses. Note, that a "deny" overrules an "allow".

5.2.17 Backup and Restore

A configuration backup of the L-DALI device can be downloaded via the Web interface. Press the **Backup/Restore** link as shown in Figure 63 to start the download. The L-DALI device assembles a single file including all required files. A file requestor dialog allows specifying the location where the backup file shall be stored.

To restore the device settings, simply select a previously generated backup file in the **Restore Configuration** section of the page by clicking the button next to the **Filename** field. Then press the **Restore** button.

The backed up configuration data consists of:

- Device settings (Passwords, IP settings, e-mail config, etc.),
- Data point configuration and persistent values,
- CEA-709 binding information,
- AST settings,
- Light and sunblind application parameters (incl. CLC Bindings),
- DALI configuration,
- Uploaded documentation and documentation links.

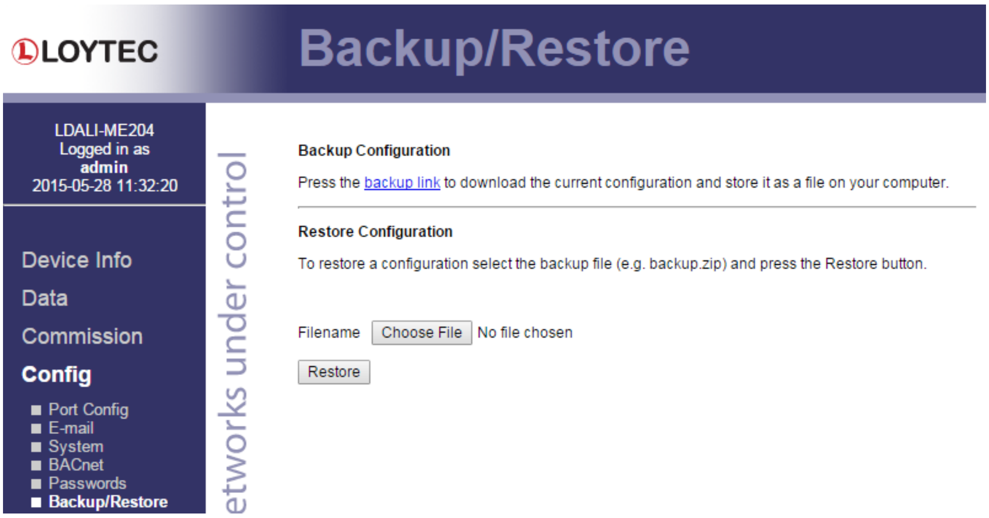


Figure 63: Backup/Restore page.

5.2.18 Debug

Log messages can be activated for each LONMARK or BACnet Object present on the L-DALI to allow analyzing the light and the sunblind applications (see Figure 64).

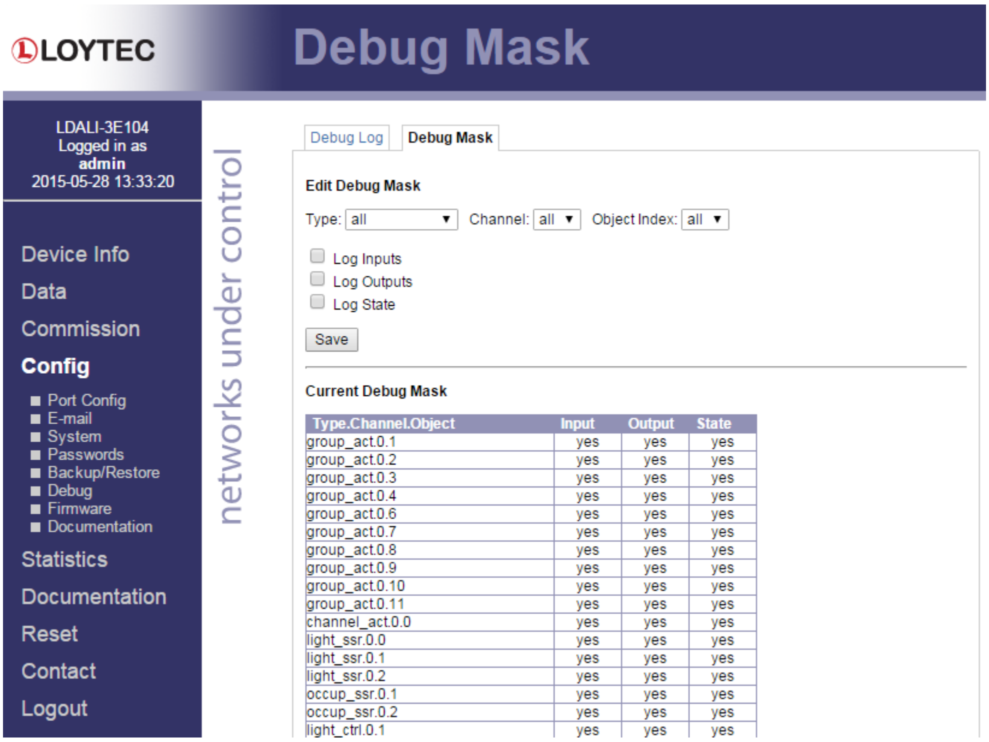


Figure 64: Setup Debug Mask.

The debug log allows recording all changes on inputs and outputs as well as all application internal transitions in case a fieldbus object does not behave as expected. Debug logs are stored in RAM. The messages logged are in most cases self-explanatory.

Further, LOYTEC support requires a debug log to be able to analyze any potential problems. Therefore, such a log should be created before contacting LOYTEC support.

To enable debugging on one or more objects, specify the desired object(s) by setting a filter using the **Type**, **Channel** and **Object Index** drop down boxes. Then check the **Log Input**/**Log Output**/**Log State** check boxes and click on the **Save** button.

To disable debugging select the object(s) by specifying a filter like when enabling debugging, leave the checkboxes **Log Input**/**Log Output**/**Log State** unchecked and click on the **Save** button.

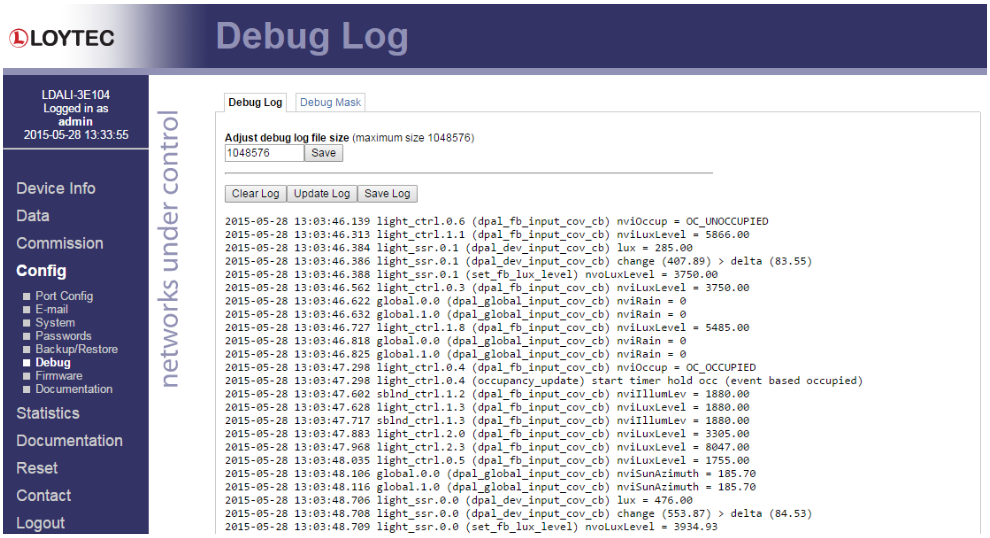


Figure 65: Debug Log.

5.2.19 Firmware

The firmware page allows upgrading the device's firmware over the Web interface. It offers two options:

- **Web Update**: With Web update the device searches for the latest available firmware on the LOYTEC server. Click on the refresh symbol, if no latest version is displayed. Please note, that the device must have a DNS server configured to find the LOYTEC server. Click on the **Install** button to upgrade your device.

- **Local file**: Update the device from a local disk file. For doing so, choose a .dl file on you hard drive and then click on the **Start Update** button.

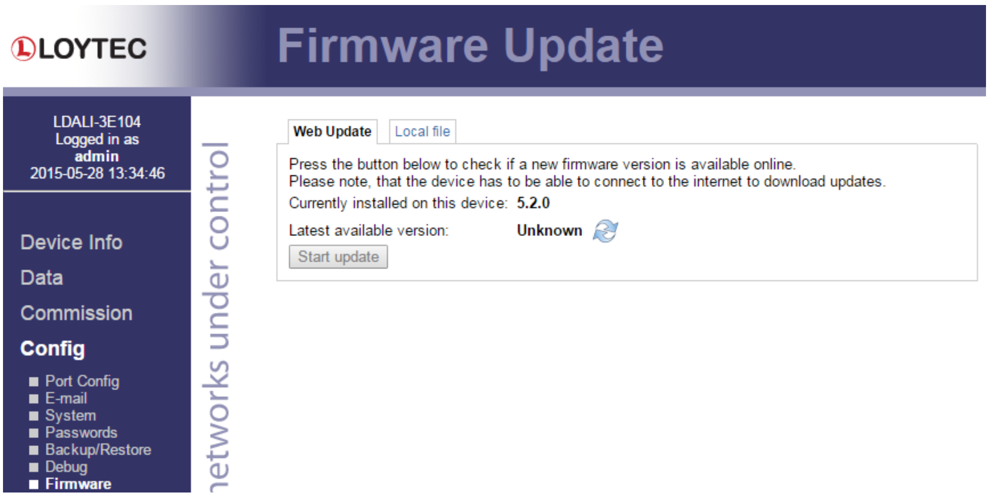


Figure 66: Firmware upgrade over the Web interface.

5.2.20 Documentation

The **Documentation** page in the **Config** menu allows uploading documentation files or configuring links to external documentation (e.g. Wiring plans, etc.). The documentation configured on this page is accessible via the **Documentation** menu (see Section 5.6).

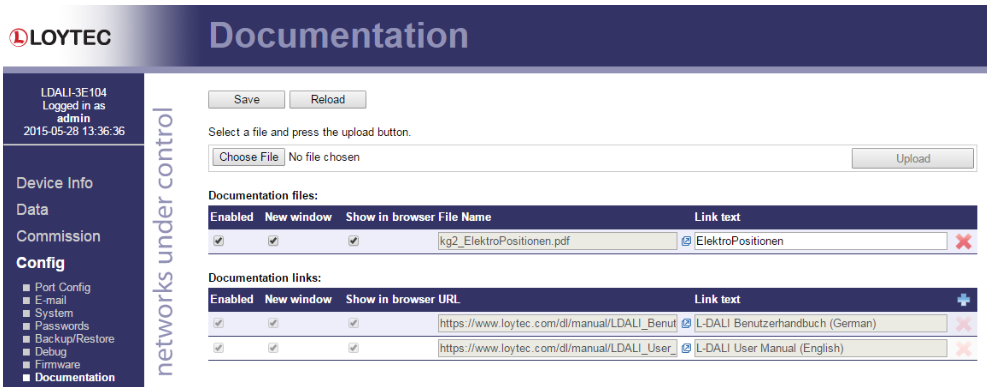


Figure 67: Upload and configure documentation.

To upload a documentation file click on the **Choose File** button. This opens a file dialog. Chose the file to upload. Click on the **Upload** button to start the upload of the selected file. After the upload is completed the file appears in the **Documentation files** section. Enter a link text used to display the uploaded file on the **Documentation** page.

To add a documentation link, click on the symbol in the header row of the **Documentation links** section. Enter the URL and the text used to display the link on the **Documentation** page.

Links and files can be set active and inactive on the **Documentation** page by checking the **Enabled** check box. Inactive entries are not displayed on the **Documentation** page. The check box **New window** determines if the link or file is opened in a new browser tab. If **Show in browser** is checked the browser will try to render the file in the browser, otherwise it will try to download the file. To remove a link or file click on the symbol on the right side of the row. To commit your changes click on the **Save** button.

5.3 Commission

The commissioning pages allow performing commissioning tasks.

5.3.1 BACnet (LDALI-20X only)

The commissioning Web UI allows assignment of physical devices to existing devices in the data point configuration, that have been created with the commission later option. Under the **Commission** menu choose the BACnet technology to open the BACnet commissioning interface.

The Web page shows a list of all **Devices in configuration**. An example is shown in Figure 68. Each line represents a device and shows the device name, the device **Instance** and the optional BACnet **Address**. The **Static Binding** checkbox defines, whether static device binding is configured for this device and requires a BACnet address. The **Status** column shows their current status. It can be one of the following:

- OK: The device is configured for communication.
- Offline: The device is configured for communication but appears offline.

- Uncommissioned: The device is not yet commissioned.
- Disabled: The device is disabled.

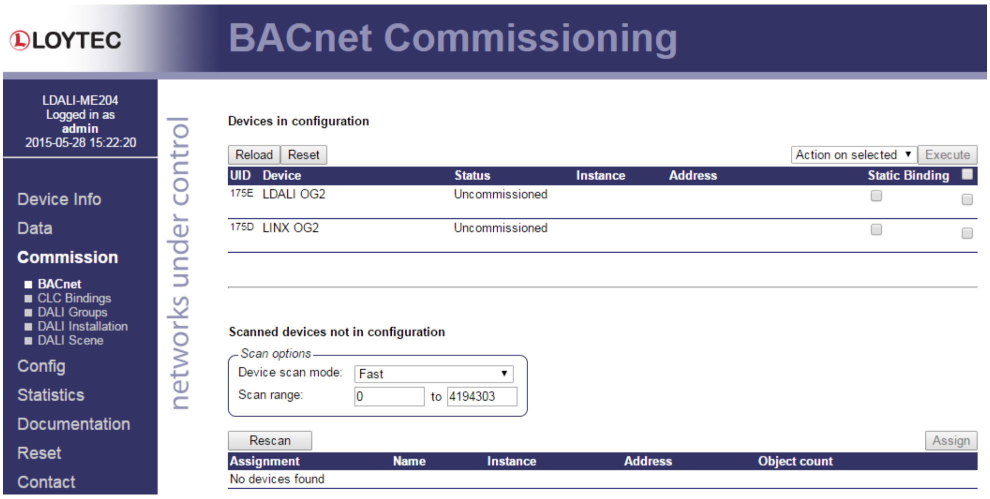


Figure 68: BACnet commissioning Web interface.

In order to execute an action on devices, select the checkbox at the end of the respective lines. Then choose an action in the drop-down **Action on selected** and click the **Execute** button. Actions that can be executed on all devices are enable and disable. A disabled device will stop communication on the network until it is enabled again.

Those devices created as commission later can be assigned to physical devices on the network. The device description displayed beneath the device name can be edited, where the edit symbol appears. The assignment can be done manually by editing the fields in the **Instance** column and **Address** column (for static device binding). It can also be done by executing a network scan. Edit the scan options as appropriate for your BACnet network and click on **Scan**. The scan progress will be displayed and fill the list for **Scanned devices not in configuration**. An example is shown in Figure 69.

Rescan | Assign

Assignment	Name	Instance	Address	Object count
UNASSIGNED	VAVtestArea11	0	192.168.5.18:47808	319
UNASSIGNED	AS_1	1	192.168.32.7:47808	48
UNASSIGNED	asa 000A	10	10001:0A	10
UNASSIGNED	FX03-00000C	12	10001:0C	124
UNASSIGNED	L-DALI-161	161	192.168.38.161:47808	1198

Figure 69: Result of a BACnet scan on the Web interface

To assign a scanned device to an uncommissioned device in the configuration, select the corresponding device name from the drop-down box in the **Assignment** column. Repeat that for all other devices and then click the button **Assign**.

5.3.2 Constant Light Controller Bindings

To determine which sensors (occupancy & lux) are used as inputs to and which light groups are controlled by a constant light controller instance go to the **CLC Bindings** page (see Figure 70).

Again, selected the DALI channel by clicking on the different tabs at the top of the page labeled **Channel 1**, **Channel 2**, etc.

Click on the symbol to add an input or output. Use the drop down box to select a sensor (input) or a light group (output). Click on symbol to remove an input or output.

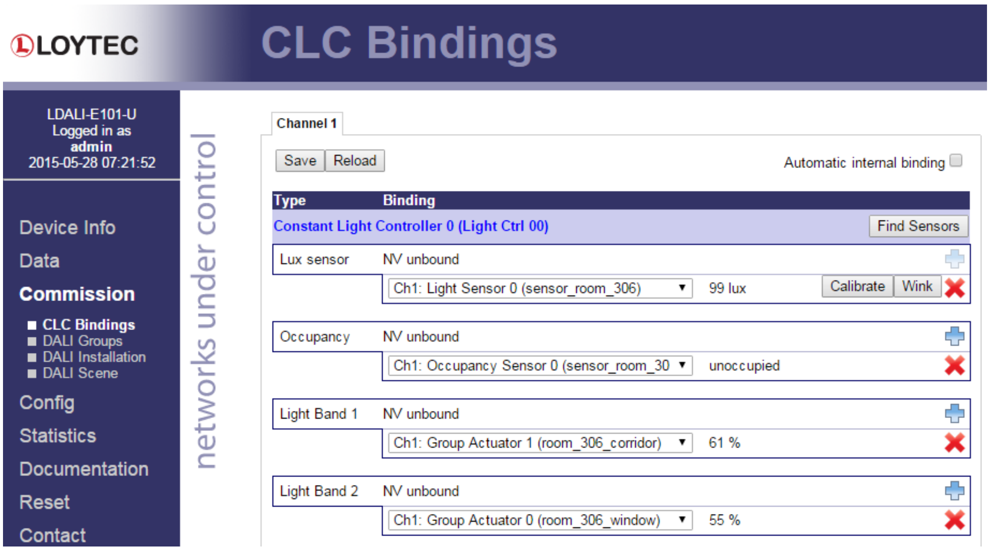


Figure 70: CLC Bindings.

For each constant light controller one lux level sensor, but up to 16 occupancy sensor inputs can be configured. Whenever at least one occupancy sensor reports occupancy the area controlled by the controller instance is considered occupied.

Similar for each of the two light bands up to 16 DALI groups can be selected as outputs. The primary light band is near the inside of the building, the secondary light band is near the window front. Depending on the outdoor light intensity the primary light band has to be brighter than the secondary light band to illuminate the room evenly.

On the LDALI-10X NV bindings can be used in parallel to the internal CLC bindings. The binding status of the corresponding NV is displayed (NV bound/unbound). To enable manual internal CLC bindings the automatic binding algorithm has to be disabled by unchecking the **Automatic internal binding** checkbox. See Section 8.2.8.1 for details on the automatic internal binding algorithm.

On the LDALI-20X the BACnet **Write priority** for the outputs (light bands) can be configured. This priority will be used by the Constant Light Controller application to write to the corresponding Analog Output object controlling the light group.

Once at least one output is configured the button **Find sensors** appears. Click on the button to start a wizard, which automatically tries to find the DALI sensors belonging to the configured light bands. The wizard will switch the light bands on and off. All sensors, which change their lux level reading when switching lights on and off will be considered to belong to the constant light controller instance. The wizard will chose one of the sensors as lux sensor and will add all sensors as occupancy sensor.

Click on the name to jump to the data point configuration page of the fieldbus object corresponding to the Constant Light Controller instance. Once saved the current lux and occupancy sensor reading and the current light level output of the configured groups will be displayed.

To calibrate a lux level sensor click on the **Calibrate** button next to it. This will open the Sensor Calibration page as described in Section 5.3.4.6, with the sensor’s Constant Light Controller instance selected.

For further details on the functionality of the constant light controller applications see sections 8.2.8 (LDALI-10X models) and 8.3.5 (LDALI-20X models).

5.3.3 DALI Groups

The DALI ballasts can be assigned to DALI groups as shown in Figure 71. Check the check box to add ballasts to groups, uncheck it to remove a ballast from a group. Commit changes by clicking on the **Save** button.

The lamp symbol shows whether the group is on or off. Clicking on it toggles the group between override to on, override to off and automatic mode. In the **Override** row a dim level override can be entered. Enter '--' to relinquish the override. In the **Feedback** row below the current average dim value (0%-100%) of the group is shown.

Click on the name to jump to the data point configuration page of the fieldbus object corresponding to the group. The name can be changed by editing the *nciLocation* (LONMARK) or *Object_Name* (BACnet) property.

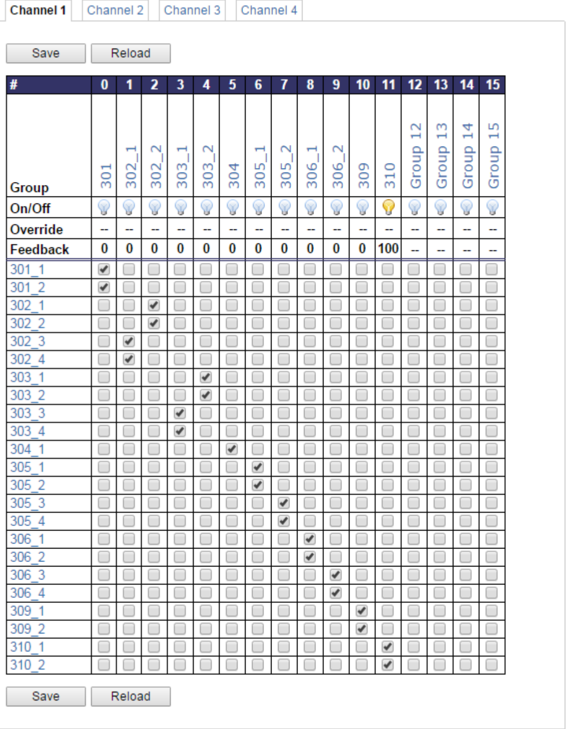


Figure 71: DALI Group Configuration.

5.3.4 DALI Installation

Figure 72 shows the initial DALI configuration page. If the device offers multiple DALI channels, the channel can be selected by clicking on the different tabs at the top of the page labeled **Channel 1**, **Channel 2**, etc.

If there is a problem with the DALI bus power on the selected channel, "**Bus supply failed**" will be displayed in the upper right corner of the tab.

The page is separated in three sections:

1. **Devices in Database**: Lists all devices on the DALI channel which were already commissioned.

2. **Scanned Devices not in Database**: Lists the uncommissioned DALI devices found during the last DALI scan.

3. **Unassigned Devices**: Lists the devices set up using the LINX Configurator PC software during an (optional) off-line preparation (see Section 7.5.1.2), which were not yet assigned to a physical DALI device.

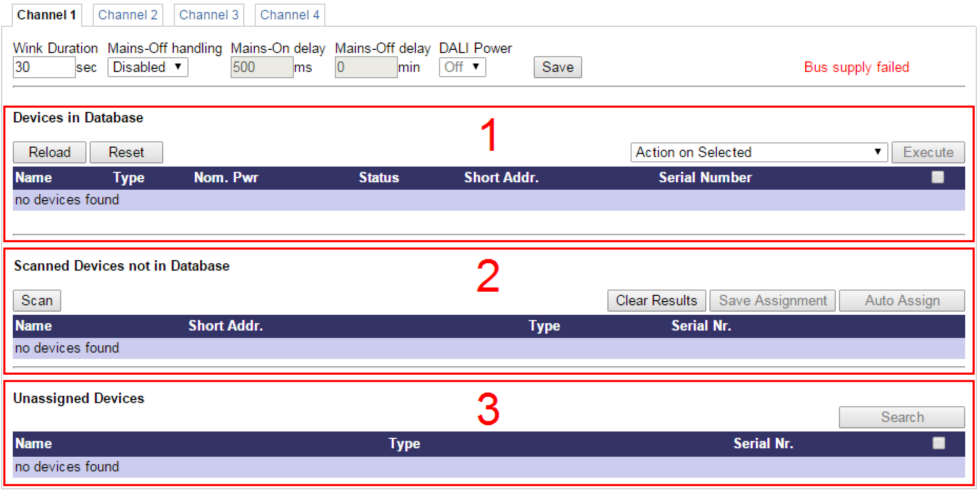


Figure 72: DALI Installation: Initial View

5.3.4.1 Installing DALI devices

To install DALI devices press the **Scan** button. The L-DALI scans the DALI channel and lists the detected devices under **Scanned Devices not in Database** in the middle of the page (see Figure 73). In case an error occurs see Section 11.4 for a description of the error codes and possible reasons.

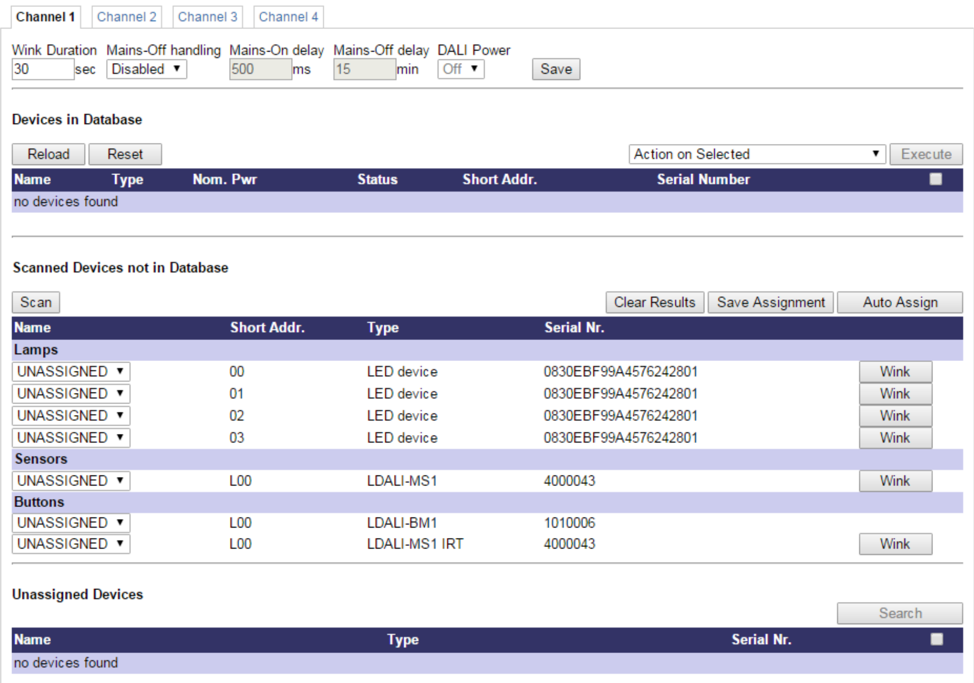


Figure 73: DALI Installation: Scan

The scanned DALI devices have to be assigned to LONMARK or BACnet objects respectively. This can be done by one of the following ways:

- **Auto Assign**: Press the button **Auto Assign** to assign the scanned lamps, sensors and buttons randomly to LONMARK or BACnet objects.

- **Manual Assign**: For each detected DALI lamp/sensor/button a drop-down list of available objects is displayed. Select an object and press the button **Save Assignment**. To identify a DALI device press the **Wink** button. The duration for how long a device winks can be configured.

- **Assignment Wizard**: If the DALI devices have been set up during the off-line preparation steps of the L-DALI configuration, a search wizard can be used to locate and assign the DALI devices to the pre-configured objects in a convenient way (see Section 5.3.4.2).

If a DALI device type (e.g. emergency lighting) has been configured during the off-line preparation steps of the L-DALI configuration, this device type must match the device type of the assigned device. In case of **Manual Assign**, the drop-down list will only offer devices with matching device type.

Depending on the DALI devices type, devices can be identified by one or more of the following ways:

- **Wink**: Devices providing some means of visual feedback can be winked. Clicking on the **Wink** button will trigger this visual feedback. DALI ballasts are typically switched on and off for the wink process. DALI sensors and DALI emergency lights usually come with a status LED which starts blinking when the device gets winked. The duration of the wink process can be configured in the field **Wink Duration** in the upper left corner of the DALI Installation page. Once a new wink is triggered any other active wink is terminated. That is, only one device will wink at a time.

- **Physical selection**: Devices which can be physically selected can be identified by selection. A DALI button can be selected by pressing one of the buttons, an occupancy sensor can be selected by triggering occupancy. The last unassigned DALI button which was pressed is marked by a symbol, the last unassigned DALI sensor which detected occupancy is marked by a symbol.

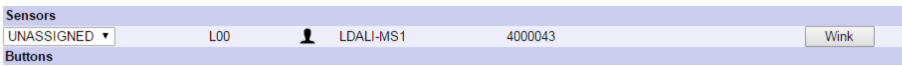


Figure 74: Sensor: Physical selection by occupancy.

  If a single physical device is listed in different device type sections, because its different functions are represented as different device types (e.g. the LDALI-MS1 is listed as sensor and its IR receiver is listed as LDALI-MS1 IRT in the button section) all sub-devices are marked in case of physical selection.

- **Serial number**: Most DALI devices come equipped with a serial number. Specifically for sensors and buttons the serial number is a convenient way to identify the device. For devices, which do not allow physical selection and cannot provide visual feedback (e.g. DALI temperature sensors or relay modules) the serial number might even be the only way to identify the device. Therefore, it is highly recommended to document the serial number of a device – if known – when the device is physically installed together with the location of the installed device. For this purpose all LOYTEC DALI devices come with an additional sticker showing the serial number of the device in text and as a bar-code. This sticker can be attached to a plan or list as part of the installation process to mark the devices location in a convenient way.

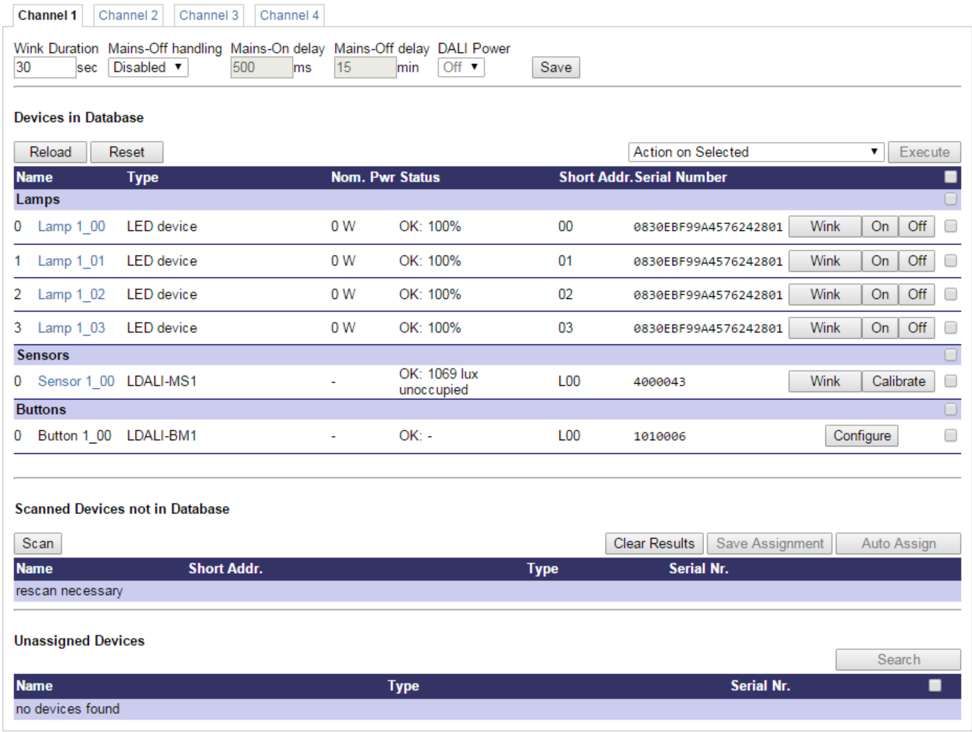


Figure 75: DALI Installation: Device Assignment

After the devices have been assigned they are listed under **Devices in Database** in the upper half of the Web interface (see Figure 75). The table displays the following information:

- **Name**: This column displays the name of the DALI device. Click on the name to jump to the data point configuration page of the corresponding fieldbus object. The name can be changed by editing the *nciLocation* (LONMARK) or *Object_Name* (BACnet) property.

- **Type**: Displays the type of the DALI device.

- **Nominal Power**: Displays the nominal power for DALI lamps. Some DALI ballasts report their nominal power. For DALI ballasts which do not support this feature the nominal power can be configured by the configuration property *nciNominalPwr* (LONMARK) or *Nominal_Power* (BACnet) of the corresponding fieldbus object.

- **Status**: This column displays the status of the DALI device and the current value. In addition the battery charge is displayed for self-contained emergency lights providing this information.

- **Short Address**: DALI short address which was assigned to the device by the L-DALI.

- **Serial Number**: This column displays the serial number of the DALI device if available. Not all DALI devices have a serial number.

- **Buttons**: Each DALI device providing some means of visual feedback can be winked. The wink duration can be configured. DALI lamps can be switched on/off manually. DALI light sensors can be calibrated (see Section 5.3.4.6). If supported by the DALI button its functions can be configured (see Section 5.3.4.7).

5.3.4.2 DALI Device Search Wizard

If the DALI devices have been assigned a name using the LINX Configurator PC software during the (optional) off-line preparation (see Section 7.5.1.2) and this configuration has been downloaded to the L-DALI a search wizard is available to assign physical DALI devices to the corresponding objects and therefore the prepared configuration:

1. Create a DALI configuration offline using the LINX Configurator software. Preconfigured DALI devices must be named to allow correct assignment once online.

2. Connect to the L-DALI and download the configuration (DALI configuration and Parameters).

3. Perform a network scan. In our example the result will look like in Figure 76.

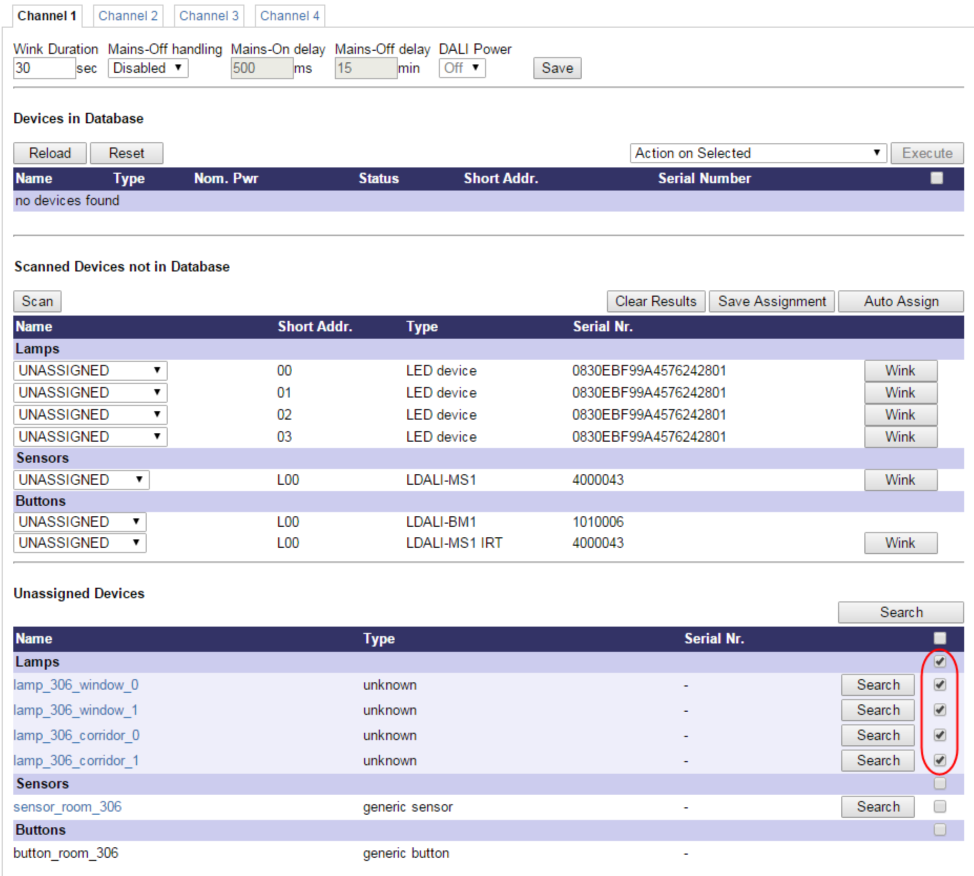


Figure 76: Scan results with unassigned devices.

4. Select the devices or device types to be identified and assigned by the search wizard by checking the check boxes on the left side of the list of unassigned devices. All devices which can provide visual feedback can be selected. And press the **Search** button on top of the list. Alternatively the wizard can be started for a single device by pressing the **Search** button in the devices row.

5. This starts a binary search to identify the ballast(s). A dialog as shown in Figure 77 appears.

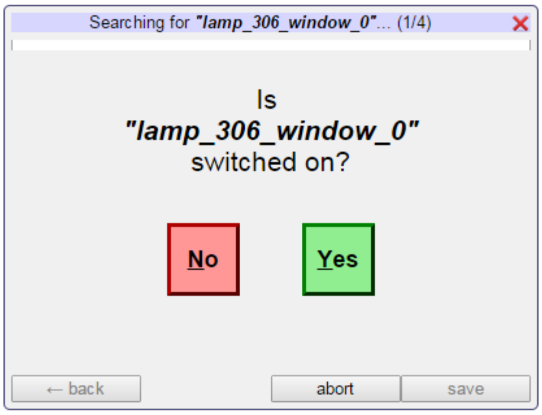


Figure 77: DALI Search Wizard.

6. Go to the luminaire/device, check whether the light is on or off and answer the question accordingly. The process is repeated until the device could be identified.

Figure 78: DALI Search Wizard: Device found.

7. A dialog as shown in Figure 78 is displayed. Then the wizard continues with the next device in the list of devices to be identified.

8. Once all DALI devices are identified press the **Save** button complete the assignment and commission the DALI devices.

For the identification during the search process DALI ballasts are switched on and off, all other devices (e.g. sensors) are winked.

5.3.4.3 Reset a DALI network

In case of a misconfigured DALI network, the **Reset** button can be used to reset the DALI configuration of all DALI devices in the network including their short address assignment. Note that if the DALI network is reset, all DALI related configuration data is lost.

5.3.4.4 Manage Devices

The devices listed under **Devices in Database** can be managed by checking the box at the right of the devices row. Then the desired function must be selected with the drop down box **Action on Selected**. Finally the **Execute** button must be pressed to perform the function. The following management functions are supported:

- **Delete**: Delete the selected device(s) from the data base. This clears the assignment to a LONMARK/BACnet object, the group assignment, and the device name. Further, it resets the DALI device to factory defaults.

- **Unassign**: Clear the assignment to a LONMARK/BACNET object, but keep the group assignment and the device name.

- **Wink**: Wink the selected devices.

- **On**: Switch the selected devices on (applies only to lamps).

- **Off**: Switch the selected devices off (applies only to lamps).

- **Reset Run Hours**: Reset the run hours of the selected lamps (applies only to lamps).

- **Reset Energy Count**: Reset the run hours of the selected lamps (applies only to lamps).

- **Start Burn-In**: Start the burn-in mode. Some lamps require a burn-in time during which they must not be dimmed. The burn-in time is defined by the configuration property *nciBurnInTime* (LONMARK) or *Burn_In_Time* (BACnet) of the corresponding channel fieldbus object. During this time the lamps will only be switched to on (100%) or off (0%) but not dimmed. The remaining burn-in time is displayed in the status column (see Figure 79 for an example).

- **Abort Burn-In**: Abort burn-in mode.

- **Emergency Light: Configure Test**: Configure auto-test calendar of self-contained emergency lights. See Section for 5.3.4.7 details.

- **Emergency Light: Start Function Test**: Start function test of self-contained emergency lights supporting this function. Please refer to the documentation of the ballast vendor to determine whether the ballast supports execution of a function test. Whether the function test is executed, pending or failed is shown in the status of the selected devices. A test is pending if its execution is delayed as the current state does not permit the execution of the test (e.g. battery not fully charged, other test being performed, etc.). Test results will be stored in the appropriate emergency light test log (see Section 8.1.4).

	Name	Type	Nominal Pwr.	Status	Short Addr.	Serial Number		
Lamps								
0	Lamp 1_00	low voltage halogen lamp	0 W	Burn-In: 100 h 100%	00	-	Wink On Off	
1	Lamp 1_01	low voltage halogen lamp	0 W	Burn-In: 100 h 0%	02	-	Wink On Off	
2	Lamp 1_02	low voltage halogen lamp	0 W	OK: 0%	03	-	Wink On Off	
3	Lamp 1_03	low voltage halogen lamp	0 W	OK: 0%	04	-	Wink On Off	
Sensors								
0	Sensor 1_00	TRIDONIC sensor	-	OK: 52 lux, occupied	01 DIAL F	-	Wink Calibrate	

Figure 79: DALI Lamps in Burn-In Mode

- **Emergency Light: Stop Function Test**: Abort any function test currently executed or pending.

- **Emergency Light: Start Duration Test**: Start duration test of self-contained emergency lights supporting this function. Please refer to the documentation of the ballast vendor to determine whether the ballast supports execution of a duration test. Whether the duration test is executed, pending or failed is shown in the status of the selected devices. A test is pending if its execution is delayed as the current state does not permit the execution of the test (e.g. battery not fully charged, other test being performed, etc.). Test results will be stored in the appropriate emergency light test log (see Section 8.1.4).

- **Emergency Light: Stop Duration Test**: Abort any duration test currently executed or pending.

5.3.4.5 Replace a DALI device

If one or more broken DALI device must be replaced, the following steps must be performed:

1. Install the new device.

2. Press the **Scan** button to detect the newly installed and unconfigured device.

3. After the scan, the DALI configuration page should look similar to Figure 80. The broken device should be marked “Offline” in the Status field and the new device should be listed in the **Scanned Devices not in Database** section. Select the defective device in the drop-down list and press the **Save Assignment** button.

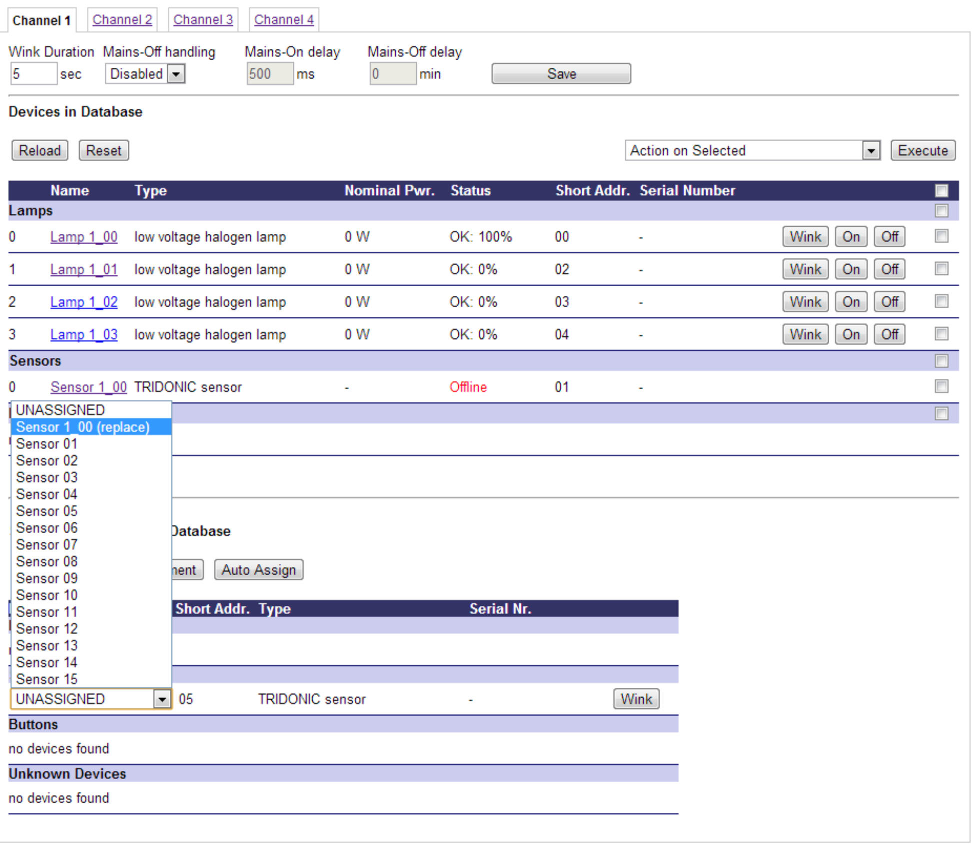


Figure 80: Replacing a defective DALI device

5.3.4.6 Sensor Calibration

To calibrate a light sensor press the **Calibrate** button on the DALI Installation page. The DALI sensor calibration page is shown in Figure 81. The L-DALI device allows calibrating the light sensor under up to seven different light conditions to counter any non-linearity of the sensor.

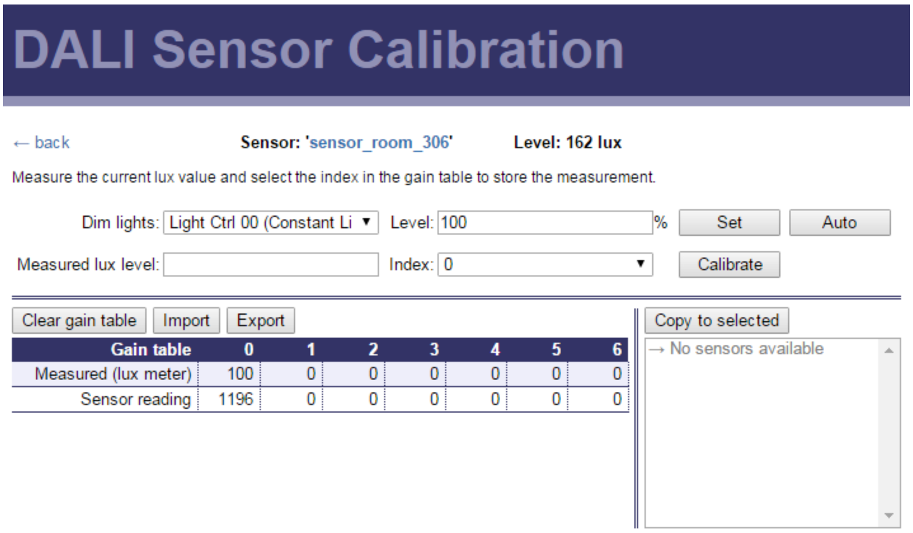


Figure 81: Sensor Calibration

To calibrate the sensor perform the following steps:

1. Measure the current lux level at the reference area (e.g. desk) using a luxmeter.

2. Optionally, the rooms light level can be adjusted. Select the appropriate DALI group or Constant Light Controller instance located in the vicinity of the sensor in the **Dim lights** drop down box. Then enter a desired dim level in the **Level** input field and press the **Set** button. To resume normal operation press the **Auto** button.

3. Enter the measured lux level in the input field and select an unused index.

4. Press the **Calibrate** button.

5. To get the more accurate sensor reading, perform steps 1. – 3. With different light conditions.

Note: *It is recommended to calibrate the sensor at least near the desired lux setpoint!*

If the sensor installation scenario is similar for multiple sensors, the calibration information can be applied to other sensor instances by selecting them in the box below the button **Copy to selected** and clicking the button. Similar the calibration information can be exported and imported by using the buttons **Import** and **Export** to transfer the data to other L-DALI devices.

To reset the calibration table press the **Clear Gain Table** button.

5.3.4.7 Button Configuration

If supported by the DALI button the function to be performed, when a button is pressed can be configured. On the DALI Installation page click on the **Configure** button next to the button device that is to be configured. The dialog shown in Figure 82 appears.

For DALI push-button coupler devices the **Mode** drop down box allows to select whether a switch or a push button is connected to the button input. The **Function** drop down box allows selecting the function to be performed when the button is pressed. See Table 11 for a list of available function. If the function is used to control light, the **Destination** column allows selecting the DALI group or channel to be controlled. If the function requires additional arguments (e.g. dim values or scene numbers) they are configured in the columns **Argument 1** and **Argument 2**.

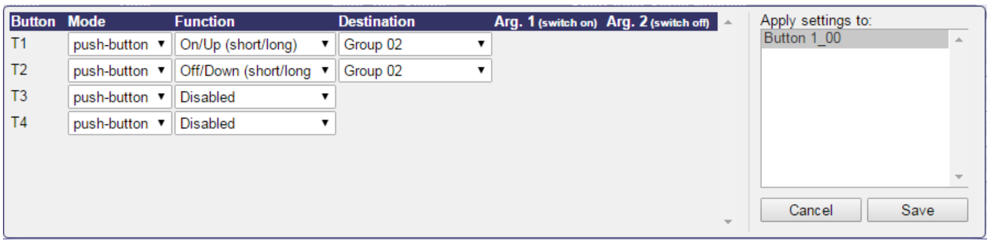


Figure 82: Configure DALI button functions.

If multiple buttons require identical or similar configuration those buttons can be selected in the **Apply settings to** selection box. Click on **Save** to write the configuration to the button device.

If DALI buttons are used with Constant Light Controller applications to allow manual and automatic operation please see Section 8.2.8.9 (LONMARK) and 8.3.5.8 (BACnet) on how DALI buttons can interact with the Constant Light Controller application.

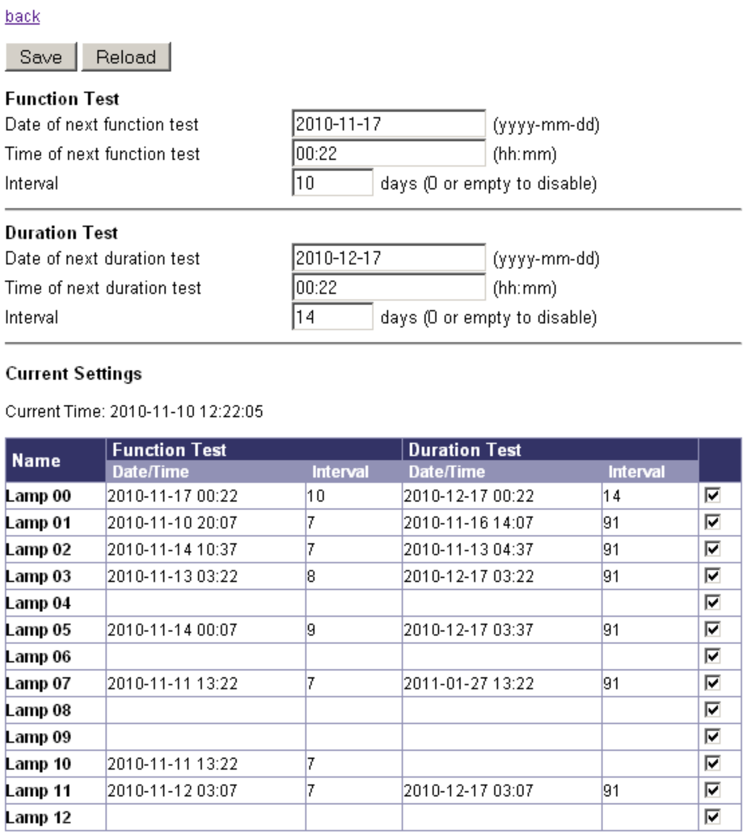
Function	Description
Disabled	No action
Generic	Button input status will be reflected by the assigned LONMARK network variable (see Section 8.2.11) or BACnet object if available (see Section 8.3.6).
On (maximum)	Switch on (recall maximum).
On (last value)	Switch on to last known value. *Note: Requires storing value when switching off.*
Off	Switch off.
Off (store value)	Store last value and switch off.
On/Off	Toggle between on and off whenever button is pressed or switch is changed depending on current status (changeover switch). Switch light on when it is off and switch light off when it is on.
Up	Dim up while button is pressed. The dim speed is determined by the parameter nciFadeRate (LONMARK) and Ramp_Rate (BACnet) of the affected ballasts.
Down	Dim up while button is pressed. The dim speed is determined by the parameter nciFadeRate (LONMARK) and Ramp_Rate (BACnet) of the affected ballasts.
Up/Down	Toggle between dimming up and down whenever button is pressed. Dim as long as button is kept pressed. The dim speed is determined by the parameter nciFadeRate (LONMARK) and Ramp_Rate (BACnet) of the affected ballasts.
On/Up	Switch on (recall maximum) if button is pressed shortly, dim up when button kept pressed. The dim speed is determined by the parameter nciFadeRate (LONMARK) and Ramp_Rate (BACnet) of the affected ballasts.
On (last value)/Up	Switch on to last known value if button is pressed shortly, dim up when button kept pressed. The dim speed is determined by the parameter nciFadeRate (LONMARK) and Ramp_Rate (BACnet) of the affected ballasts.
Off/Down	Switch off value if button is pressed shortly, dim down when button kept pressed. The dim speed is determined by the parameter nciFadeRate (LONMARK) and Ramp_Rate (BACnet) of the affected ballasts.
Off (store value/Down)	Store last value and switch off if button is pressed shortly, dim down when button kept pressed. The dim speed is determined by the parameter nciFadeRate (LONMARK) and Ramp_Rate (BACnet) of the affected ballasts.
On/Off and Up/Down	Toggle between on and off whenever button is pressed shortly depending on current status (changeover switch). Toggle between dimming up and down whenever button is pressed longer. Dim as long as button is kept pressed. The dim speed is determined by the parameter nciFadeRate (LONMARK) and Ramp_Rate (BACnet) of the affected ballasts.
On (last value)/Off and Up/Down	Toggle between switching on to last value and off (including storing the last value) whenever button is pressed shortly depending on current status (changeover switch). Toggle between dimming up and down whenever button is pressed longer. Dim as long as button is kept pressed. The dim speed is determined by the parameter nciFadeRate (LONMARK) and Ramp_Rate (BACnet) of the affected ballasts.
Dim to	Dim to the value entered as argument 1.
Dim to (toggle)	Toggle between dim value entered as argument 1 and dim value entered as argument 2 whenever button is pressed or

Function	Description
	switch is changed.
Recall scene	Recall the scene configured with argument 1.
Recall scene (toggle)	Toggle between recalling scene configured with argument 1 and scene configured with argument 2 whenever button is pressed or switch is changed.
Colour warmer	Ramp up colour temperature as long as button is pressed. *Note: Requires ballasts of type colour control (device type 8) supporting the colour type colour temperature.*
Colour cooler	Ramp down colour temperature as long as button is pressed. *Note: Requires ballasts of type colour control (device type 8) supporting the colour type colour temperature.*

Table 11: DALI button functions

5.3.4.8 Emergency Light Auto-Test Configuration

To configure the test calendar for the automatic function and duration tests of self-contained emergency lights supporting this function check the box at the right of the devices row of the DALI Installation page, choose **Emergency Light: Configure Test** from the **Action on Selected** drop-down box and click on the **Execute** button. Now the L-DALI will read the currently configured test calendar from the selected devices. When done it will show a page similar to the one shown in Figure 83.



Name	Function Test Date/Time	Function Test Interval	Duration Test Date/Time	Duration Test Interval	
Lamp 00	2010-11-17 00:22	10	2010-12-17 00:22	14	☑
Lamp 01	2010-11-10 20:07	7	2010-11-16 14:07	91	☑
Lamp 02	2010-11-14 10:37	7	2010-11-13 04:37	91	☑
Lamp 03	2010-11-13 03:22	8	2010-12-17 03:22	91	☑
Lamp 04					☑
Lamp 05	2010-11-14 00:07	9	2010-12-17 03:37	91	☑
Lamp 06					☑
Lamp 07	2010-11-11 13:22	7	2011-01-27 13:22	91	☑
Lamp 08					☑
Lamp 09					☑
Lamp 10	2010-11-11 13:22	7			☑
Lamp 11	2010-11-12 03:07	7	2010-12-17 03:07	91	☑
Lamp 12					☑

Figure 83: Emergency Light Auto-Test Configuration

For both tests – function and duration test – a test interval in days and the time and date of the next execution of the test can be specified. Click **Save** to store the new values in the devices selected by the check box at the right of the devices row.

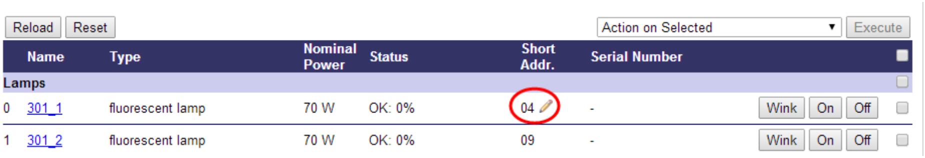
Note: *The resolution of the duration test interval is 7 days, the resolution of the delay to the next test execution is 15 minutes. In both cases the value entered will be rounded to the next appropriate value. Further, the time and date of the next test execution is calculated based on the system time and time zone of the L-DALI as configured in the System Configuration (see Section 5.2.11).*

Test results will be stored in the appropriate emergency light test log (see Section 8.1.4).

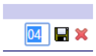
5.3.4.9 Changing the DALI Short Address of Devices

If required, the short address of a DALI device can be changed once it is assigned:

1. Click on the pencil, which appears next to the devices short address when moving the mouse over it.



2. Enter a valid short address in the text field that appears and press ENTER. A valid short address is in the range 0 to 63 and must be unique within one DALI channel.



Changing the short address has no effect to the assignment of the device in the L-DALI or any other function performed by the L-DALI.

5.3.4.10 Mains-Off Handling

When all ballasts on a channel are off the L-DALI allows switching off the ballast mains. This function allows saving the standby energy consumed by the ballasts. The drop down box **Mains-Off handling** and the parameters **Mains-On delay** and **Mains-Off delay** allow configuring this function. For further details see Section 8.4.7.

5.3.4.11 Enable/Disable Internal DALI Bus Power Supply

L-DALI devices with internal DALI bus power supply show the status of the internal DALI bus power supply in the drop-down box **DALI Power** (see Figure 72). If supported by the device the internal DALI bus power supply can be enabled using the drop-down selection. Changes take effect when pressing the **Save** button to the right of the drop down box.

5.3.4.12 Maintenance

For maintenance purposes several functions of the DALI Installation web-UI page are available for the user “operator” as well. These functions are:

- Replacing a broken DALI device (including scan).
- Reset Run Hours/Energy count.
- Start/stop burn in
- All Emergency Light functions.

All other functions are not available when logged in as “operator” user.

5.3.5 DALI Scene

Figure 84 shows the **DALI scene** page. It allows the manual configuration of DALI scenes. If the device offers multiple DALI channels, the channel can be selected by clicking on the different tabs at the top of the page labeled **Channel 1**, **Channel 2**, etc.

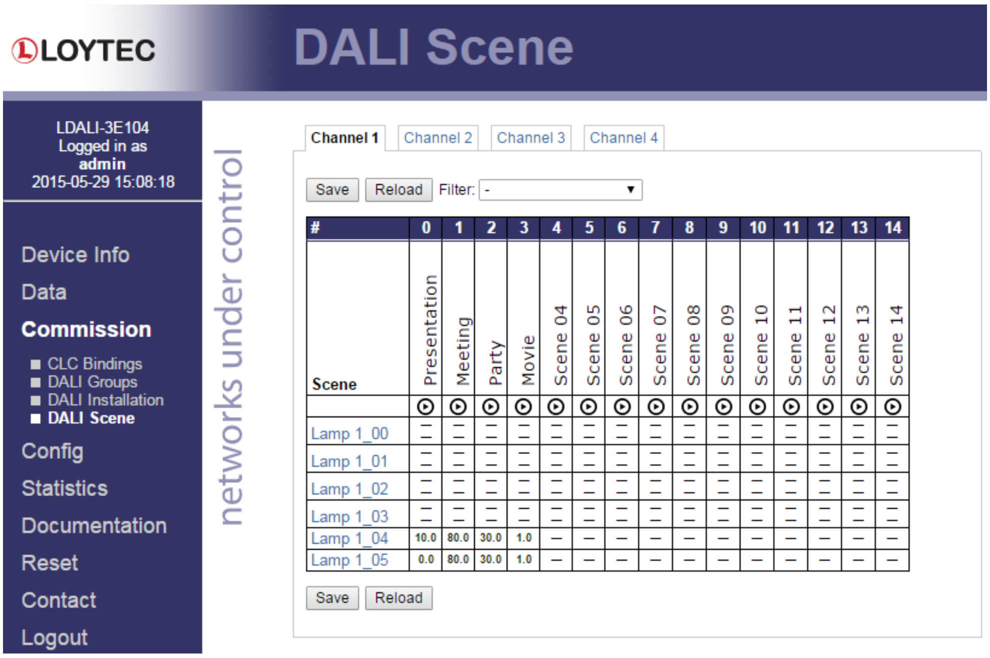


Figure 84: DALI Scene configuration.

Each DALI ballast allows to store up to 15 scenes[1]. For each scene a name can be configured. Click on the scene name to edit it. For each ballast a different dim level can be configured for each of its scenes. If recalling the scene shall not affect the ballast's dim level set the value to '--'.

For ballasts supporting colour control (DALI device type 8) the scene can include colour information, too. Depending on the colour type(s) supported the colour information is configured as follows:

- **Colour Temperature only**: For devices supporting *Colour Temperature* only two values can be entered for each scene. The upper value is the dim level, the lower value is the colour temperature for the scene.

- **XY Coordinates**: For devices supporting *XY Coordinates* a colour picker dialog as shown in Figure 85 appears when clicking on the lower value. Either manually enter the x and y coordinate of the scene colour within CIE 1931 colour space or pick the colour by clicking in the colour diagram on the left side of the dialog. When the check box **Live preview** is checked the ballast will dim to the selected colour whenever a new value is selected. The last six colour values used are shown in the history below the colour diagram for quick reference. If the ballast supports *XY Coordinates* and *Colour Temperature* the colour temperature value can be entered as alternative to the colour coordinates.

[1] DALI ballasts support up to 16 scenes. Scene 15 is used by the L-DALI controller and the LOYTEC DALI buttons to store the last dim value when switching off and therefore is not available.

- RGBWAF and primary-N colour types are currently not supported.

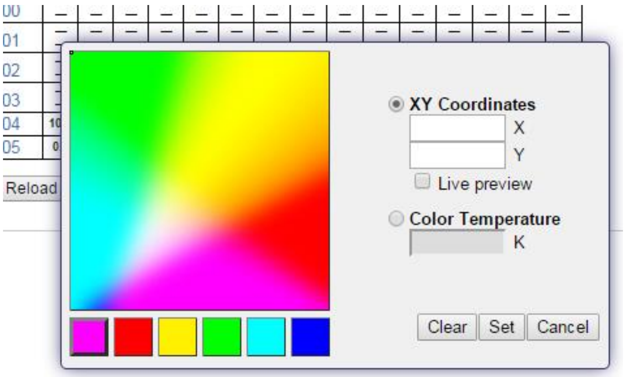


Figure 85: Scene colour selection for devices supporting colour type XY Coordinates.

In all cases the enter '--' in the upper value if the dim level shall not be affected when the scene is recalled and enter '--' in the lower value if the colour shall not change when the scene is recalled.

Scenes are stored for each ballast, but are typically recalled for a group. To show only the ballasts belonging to a certain group select the group in the **Filter** drop down box on the top of the page.

To test a scene configuration before saving it click on the symbol. This will dim the ballasts selected by the current filter to the values configured for the scene.

Scenes can be recalled using DALI buttons or by the L-DALI controller (see 8.2.5.1 and 8.3.3.1).

5.4 Data Management

5.4.1 Data Points

The device's Web interface provides a data point page, which lists all configured data points on the device. An example is shown in Figure 86. The data point page contains a tree view. Clicking on a particular tree item fills the right part of the page with a data point list of that tree level. A breadcrumb navigation header allows navigating back to an arbitrary level in the tree.

The data point list displays the data point name, direction, type, data point state, current value, and a description. All values are updated live. Inactive points are displayed in gray. If the data point list does not fit on one page, there are page enumerator links at the bottom. Important data point states and their implications are listed in Table 12. Values can be directly edited in the list where the pencil symbol appears. Data point structures can be expanded or collapsed for better overview.

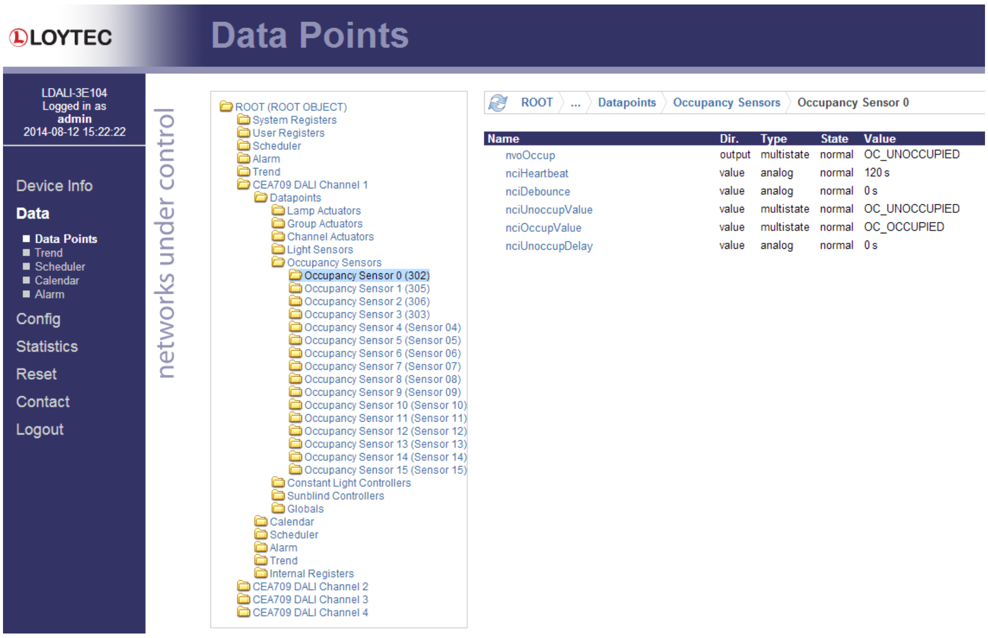
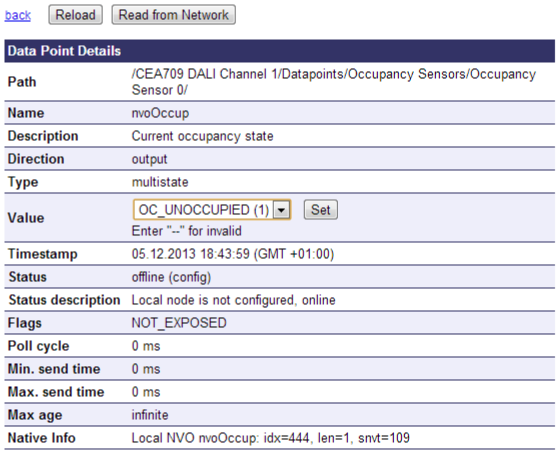


Figure 86: Data point page.

Data Point Status	Description
normal	The data point is in normal operation state and possesses a value.
Invalid value	The data point has no valid value.
normal (config)	The data point has a normal value but it is not fully configured on the network (not commissioned, no binding, no client mapping, etc.)
offline	The data point has a value but it is not reflected on the network due to a communication error (e.g., the peer node is not online).
unreliable (offline)	The data point is in normal operation. The value of it, however, is qualified as unreliable because a connected data point is offline. For an output data point it means that the value was fed from a connection, where the source is offline. For an input data point it means that the connected output data point could not send the value to the network.
unreliable (range)	The data point is in normal operation. The value of it, however, is qualified unreliable because the value is an out-of-range value for the connected data point. The value is limited to the supported range.
unreliable	The data point is in normal operation. The value of the data point or a connected data point has been tagged as unreliable over the network. This is the case when the BACnet reliability has been written.
not configured	The data point is mapped to a port, which is not configured (e.g., the port is disabled).
inactive	The data point is inactive and the line is grayed-out. Values can be written but no network communication is triggered. This can be the case, if a data point is not used in the configuration or it is connected to a BACnet server object, which is not present on the device.

Table 12: Data Point States.

The data point names are links. Clicking on such a link opens a detailed page on that data point. If the data point supports it, the user can also enter a new data point value as depicted in Figure 87. The **Status** field is discussed in Table 12. The field **Status Description** contains a describing text for the data point status. The **Native Info** field displays detailed information on the underlying technology object.



back | Reload | Read from Network

Data Point Details	
Path	/CEA709 DALI Channel 1/Datapoints/Occupancy Sensors/Occupancy Sensor 0/
Name	nvoOccup
Description	Current occupancy state
Direction	output
Type	multistate
Value	OC_UNOCCUPIED (1) Set Enter "--" for invalid
Timestamp	05.12.2013 18:43:59 (GMT +01:00)
Status	offline (config)
Status description	Local node is not configured, online
Flags	NOT_EXPOSED
Poll cycle	0 ms
Min. send time	0 ms
Max. send time	0 ms
Max age	infinite
Native Info	Local NVO nvoOccup: idx=444, len=1, snvt=109

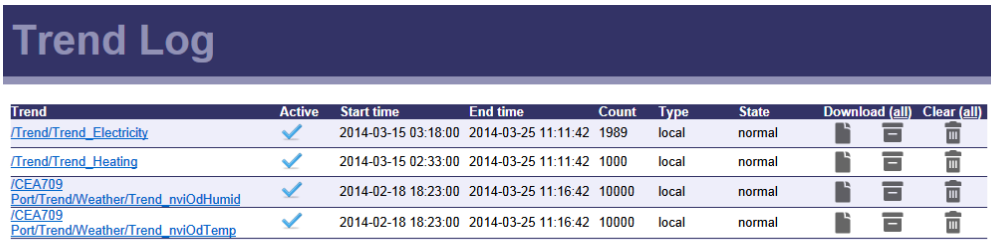
Figure 87: Data point details page.

Clicking on the **Set** button writes the new value to the device's data server. When setting a value, the Web page displays the status of the action:

- **Successfully set value**: The new value has been successfully set in the data point and the update has been sent on the network, if it is a network data point.
- **Could not send value update**: The new value has been set but it has not been sent out on the network. The reason can be that the peer node is currently offline or there is a configuration error. The data point status reflects this error.
- **Could not set value (error code)**: The new value has not been set because of an internal error. Please contact LOYTEC with the error code.

5.4.2 Trend

The Web interface provides a trend log overview page to see all available trend logs and their current state (active, first available date/time, last date/time, number of records). An example is shown in Figure 88. This list allows a convenient upload of single trend data in CSV format by clicking on the respective icons. To upload an archive of all trend data click on the **all** link in the **Download** column heading. It is also possible to purge single or all trend logs directly from that list.



Trend Log

Trend	Active	Start time	End time	Count	Type	State	Download (all)	Clear (all)
/Trend/Trend_Electricity	✓	2014-03-15 03:18:00	2014-03-25 11:11:42	1989	local	normal		
/Trend/Trend_Heating	✓	2014-03-15 02:33:00	2014-03-25 11:11:42	1000	local	normal		
/CEA709 Port/Trend/Weather/Trend_nviOdHumid	✓	2014-02-18 18:23:00	2014-03-25 11:16:42	10000	local	normal		
/CEA709 Port/Trend/Weather/Trend_nviOdTemp	✓	2014-02-18 18:23:00	2014-03-25 11:16:42	10000	local	normal		

Figure 88: Trend log overview on Web UI.

Click on a trend log and re-configure local trend logs at run-time. The changes made to the trend logs take effect immediately without the needs for a reboot of the device. Allocating new trend logs can only be done in the configuration software (see Section 7.13.1) or in case of the BACnet version using a BACnet Operator Workstation. The trend log main page displays all available trend logs. Click on the trend log to be edited. This opens the trend log configuration page. An example is shown in Figure 89.

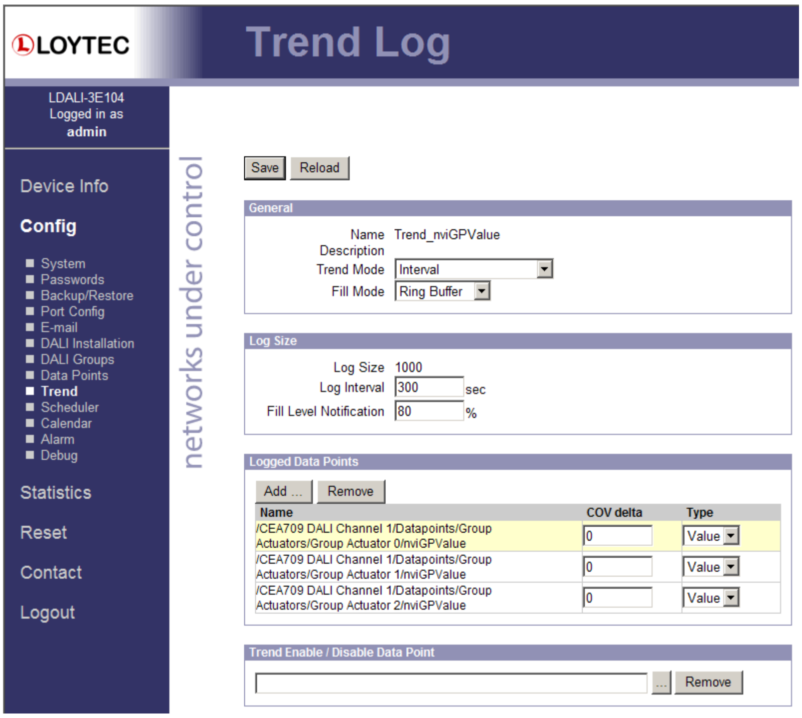


Figure 89: Trend log configuration page.

The user can change the **Trend Mode**, the **Fill Mode**, the **Log Interval** and the **Fill Level Notification**. Furthermore, data points can be added to the trend log by clicking the **Add…** button. A data point selector dialog opens. Click on a data point for adding it. For removing a data point from the trend log, click on it in the **Logged Data Points** list and hit the **Remove** button. Save the changes made by clicking the **Save** button. For more information on how a trend log can be configured please refer to the Configurator Section 7.13.2.

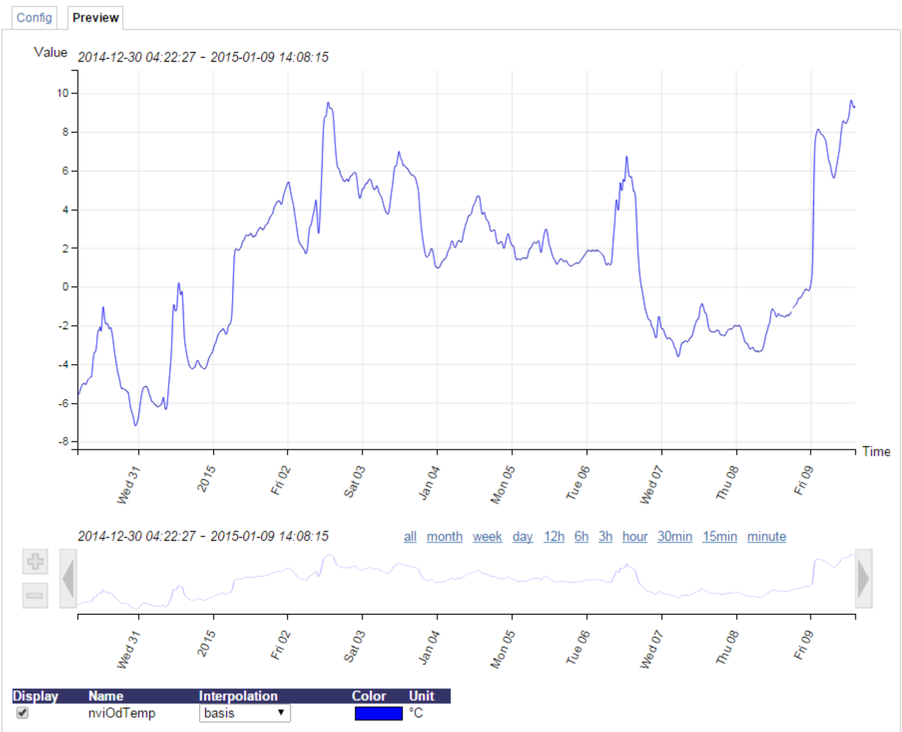


Figure 90: Web UI trend chart preview

To look at the historical trend data in a chart view select the **Preview** tab as shown in Figure 90. Trend logs with multiple data points are shown with multiple color-coded curves. A legend at the bottom of the page identifies the trended data points. Moving the mouse over the trend chart shows a data curser displaying time stamp and actual value.

Using the chart slider below the trend chart, one can zoom in and out in time as well as shift the time axis. Click into the slider and drag the mouse while keeping the button pressed in order to span a sub-interval, which is displayed in the chart view. Alternatively, select one of the pre-defined sub-intervals (week, day, etc.) and drag the sub-interval along the time axis.

Data points can be deselected in the legend at the bottom of the window. This hides the respective curves in the chart view and may improve visibility for certain detail. Enable the data points again and the curves will re-appear.

5.4.3 Scheduler

The Web interface provides the scheduler page to edit its schedules at run-time, i.e., change the times and values that shall be scheduled. Setting up new schedules (configuring scheduled datapoints etc.) can be done in the configuration software (see Section 7.10) or in case of the BACnet version using a BACnet Operator Workstation. The scheduler main page displays all available schedules. Click on the schedule to be edited. This opens the scheduler page. An example is shown in Figure 91.

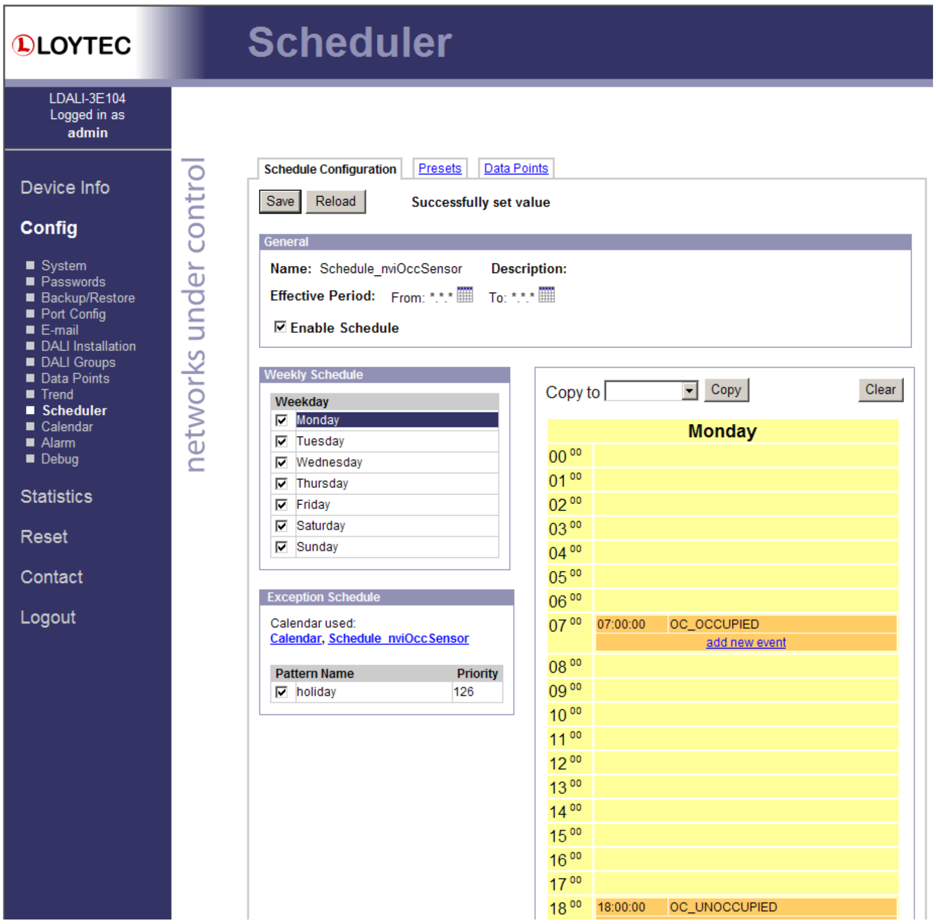


Figure 91: Schedule Configuration Page.

The **effective period** defines when this schedule shall be in effect. Leave **From** and **To** at '*.*.*' to make this schedule always in-effect. Otherwise select the desired start and/or end dates by clicking the calendar icons. To entirely disable a scheduler de-select the **Enable Schedule** check box.

Schedules are defined per day. On the left-hand side, the weekdays **Monday** through **Sunday** can be selected, or exception days from the calendar, e.g., Holidays. Once a day is selected, the times and values can be defined in the daily planner on the right-hand side. In the example shown in Figure 91, on Monday the value **OC_OCCUPIED** is scheduled at **7:00am** and the value **OC_UNOCCUPIED** is scheduled at **6:00pm**. The same principle applies to **exception days**. **Exception days** override the settings of the normal weekday. Put a check mark on those exception days from the calendar, which shall be used in the schedule. To edit the date ranges of exception days click on the links to the used calendars, e.g., 'Calendar' or 'Schedule_nviOccSensor'. The 'Schedule_nviOccSensor' is a calendar, which is embedded into the schedule and not accessible by other schedulers. For more information on how to set up schedules and calendars refer to Section 7.10.

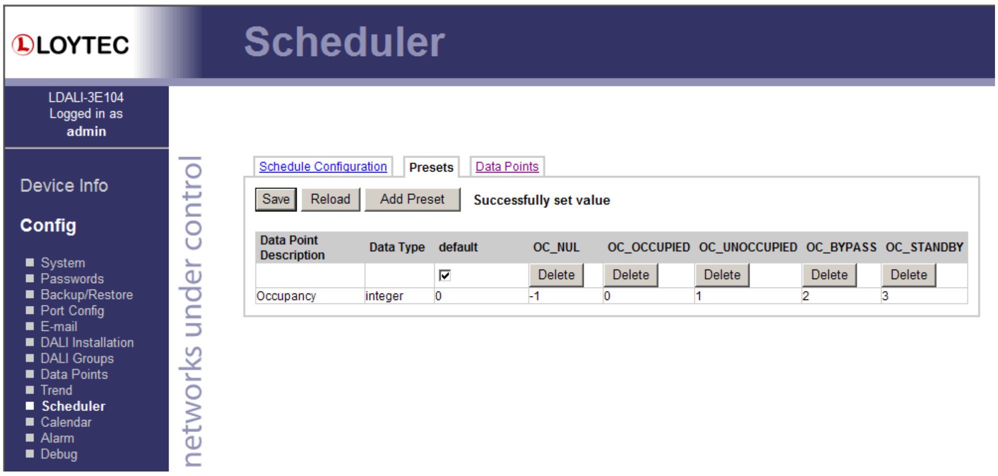


Figure 92: Scheduled Presets Configuration Page.

To define actual values for the names such as **OC_OCCUPIED** click on the tab **Presets** as shown in Figure 92. To define a new value, click on the button **Add Preset**. This adds a new column. Enter a new preset name. Then enter values for the data points in the **preset** column. The **Data Point Description** column displays the short-hand name defined in the configuration software. Click on this description to change it on the Web interface.

You can switch back and forth between the two tabs. Once the configuration is complete, click on the **Save** button. This updates the schedule in the device. Any changes made become effective immediately.

Note: *Clicking **Save** may remove any presets which are currently not used in any of the daily schedules. This happens for example in native BACnet schedules, where the underlying network technology cannot store presets individually. Therefore always complete the daily schedules first and then press save as the last step.*

For local schedulers using the CEA-709 network technology the Web UI also allows reconfiguring the scheduled data points. This change takes effect immediately without a reboot of the device. To add and remove data points to the scheduler, go to the **Data Points** tab. The configuration page is depicted in Figure 93. To add a new data point, click the **Add…** button. To remove a data point, select the data point in the list **Scheduled Data Points** by clicking on it and then press the **Remove** button. Finally, store the changes by clicking the **Save** button. After modifying the scheduled data points, go back to the Presets tab and enter descriptive value label names. For more information on how to configure a scheduler please refer to the Configurator Section 7.11.3.

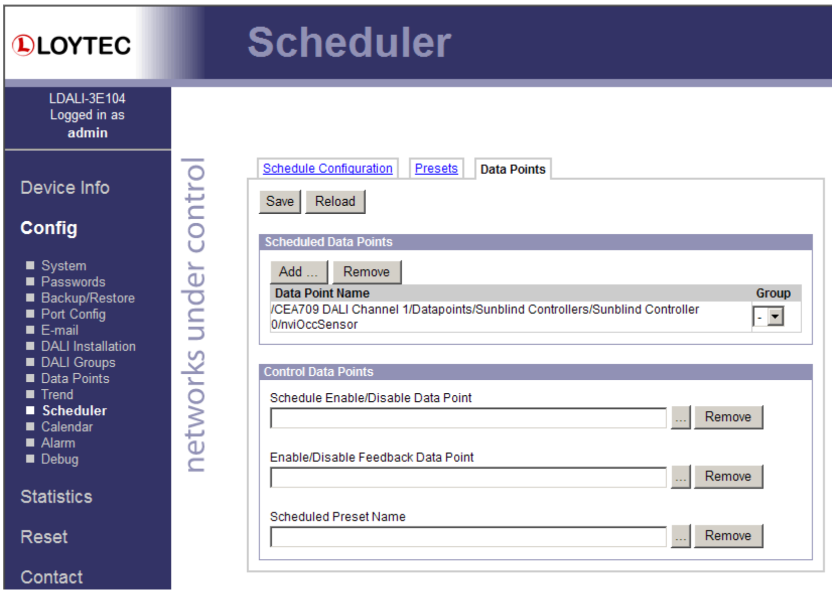


Figure 93: Re-configure scheduled data points on the Web UI.

5.4.4 Calendar

The Web interface provides the calendar page to edit its calendars at run-time, i.e. change the exception days. The calendar main page displays all available calendars. Click on the calendar to be edited. This opens the calendar configuration page. An example is shown in Figure 94.

The **effective period** defines when this calendar shall be in effect. Leave **From** and **To** at '*.*.*' to make this calendar always in-effect. Otherwise enter dates, such as '30.1.2000'.

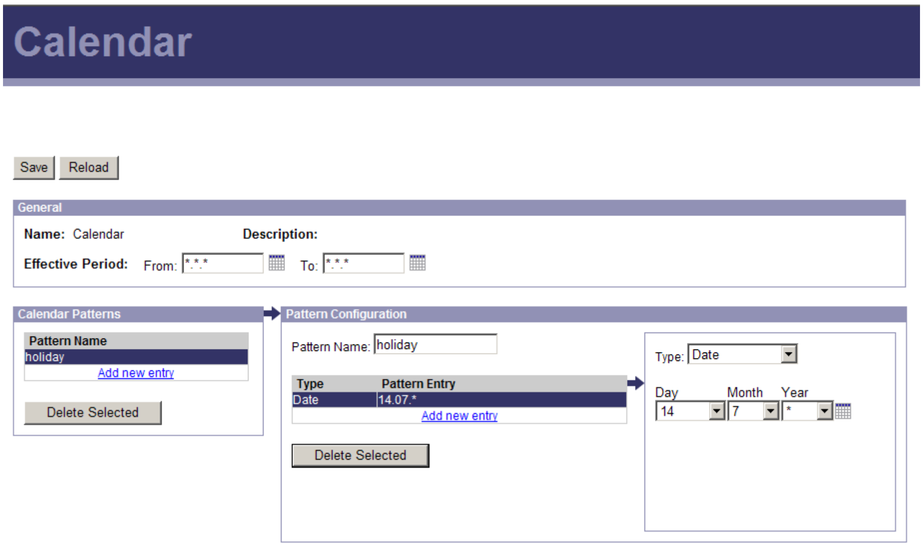


Figure 94: Calendar Configuration Page.

On the remainder of this page, work from left to right. Click on a calendar pattern or create a new calendar pattern by clicking **Add new entry**. A calendar pattern defines a set of pattern entries, which defines the actual dates or date ranges. In the example in Figure 94, the calendar pattern **holidays** is selected.

In the **Pattern Configuration** box, the calendar pattern's name can be edited if supported by the underlying network technology. Otherwise, an auto-generated name will be assigned and the pattern name box is not shown. Below the pattern name is a list of the individual pattern entries. New entries can be added by clicking **Add new entry**.

Note: *Embedded calendar patterns can only have exactly one entry to define the dates at which the pattern should be in effect. Only calendar patterns in global calendars may consist of multiple entries.*

Existing entries can be selected and edited in the box on the right-hand side. In the example in Figure 94, the date **14.7.*** is selected, which means “The 14.7. of every year”. Other entry types such as **Date Range** and **Week-and-Day** can be selected. See Section 6.4.3 for more information about defining exception dates.

5.4.5 Alarm

The Web interface provides the alarm page to view the currently pending alarms of its alarm data points. The alarm main page displays all available alarm data points. Alarm objects which have active alarms are displayed in red. Click on the alarm object to be viewed. This opens the alarm summary page. An example is shown in Figure 95.

Active alarms are highlighted red. Inactive alarms which have not been acknowledged are rendered in green. Alarms that can be acknowledged have an **Ack** button. Press on the **Ack** button to acknowledge the alarm. Depending on the technology, this and older alarm records will be acknowledged. Acknowledged, active alarms are rendered in red. Click on **Reload** to refresh your alarm list.

Inactive alarms that have been acknowledged disappear from the list. To record historical information about those alarms, the alarm log must be used. See Section 5.5.14 for the alarm log Web interface.

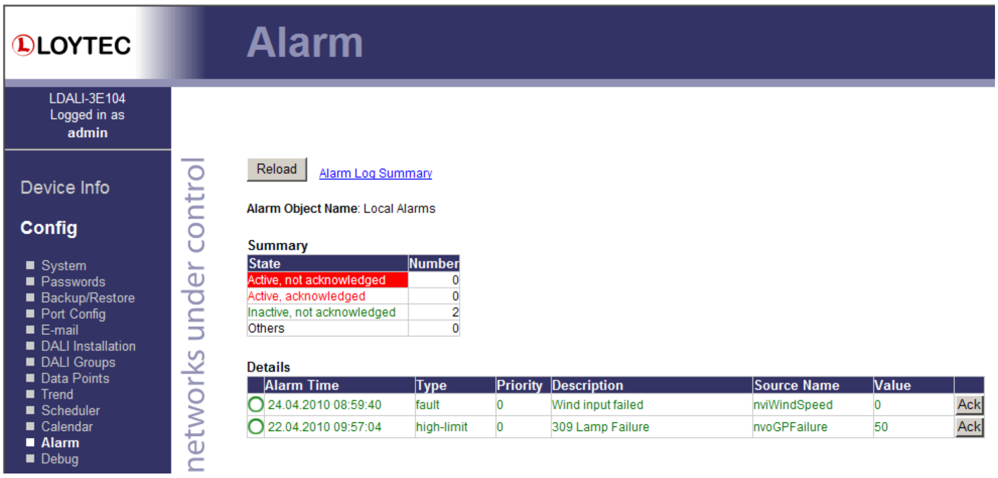


Figure 95: Alarm Summary Page.

5.5 Device Statistics

The device statistics pages provide advanced statistics information about the CEA-709 device, the CEA-852 device, the BACnet device, the System Log, the scheduler, the Alarm Log and the Ethernet interface.

5.5.1 System Log

The **System Log** page prints all messages stored in the system log of the device. An example is shown in Figure 96. This log data is important for trouble-shooting. It contains log entries for reboots and abnormal operating conditions. Errors and warnings are color-coded in red and yellow. The default log direction is newest entries on top. The direction can be edited by clicking on the arrow ⇑ in the column header.

To save the log click on the **Save System Log** button. When contacting LOYTEC support, have a copy of this log ready.

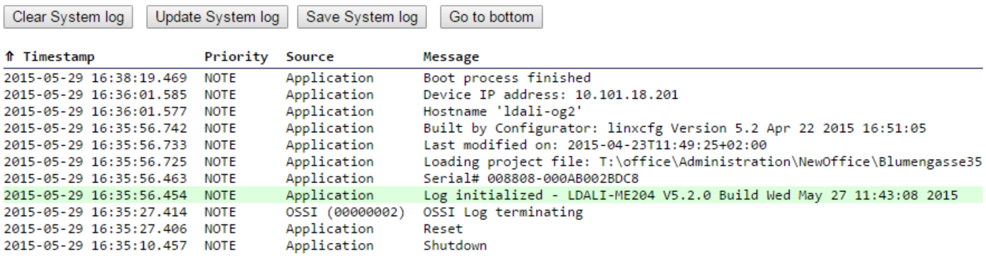


Figure 96: System Log Page.

5.5.2 BACnet Bindings Statistics (LDALI-20X only)

The BACnet bindings statistics page displays a list of all currently active address bindings. This list can be used for troubleshooting to see, which BACnet device instance numbers could be resolved and to what BACnet network number and MAC address. See Figure 97 for an example list. In this case the device instance 224220 has been resolved to the local network and MAC address 192.168.24.220:BAC0.

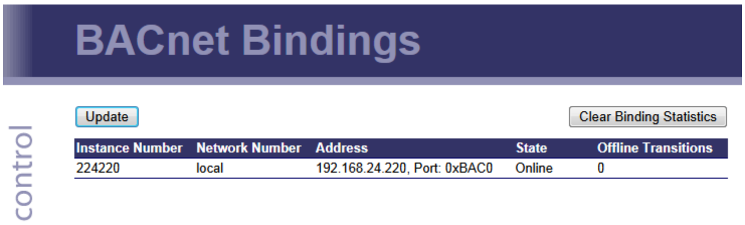


Figure 97: BACnet bindings statistics page.

5.5.3 BACnet MS/TP Statistics (LDALI-ME204 only)

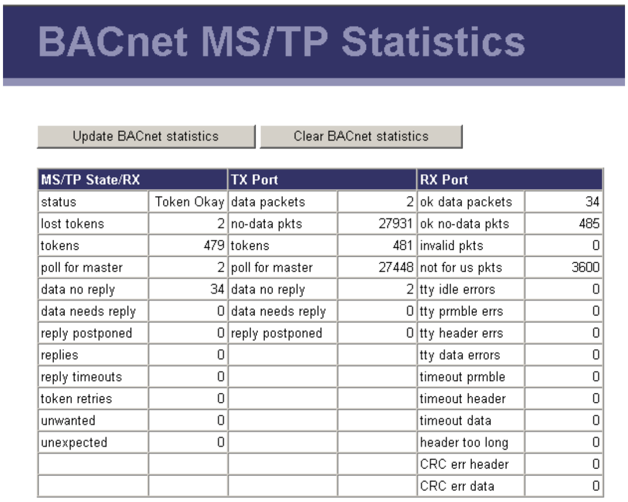
The BACnet MS/TP statistics page is only available, when the MS/TP data link layer is enabled (see Section 5.2.9) and supported by the L-DALI model. The three statistics items displayed are: Device Statistics, Bus History, and Token History.

The **MS/TP Device Statistics** (see Figure 98) is split into three major columns, **MS/TP State/RX**, **TX Port**, and **RX Port**. The MS/TP State/RX column contains information related to the status of the MS/TP machine as well as packets received and processed by the MS/TP state machine. The TX Port column counts packets sent by the device according to their types, and the RX Port column tracks packets and errors seen by the MS/TP receive state machine.

The most prominent information in the **MS/TP State/RX** column is the **status** entry which describes the current status of the MS/TP token as perceived by the device. In status **Token Ok**, the token is circulating between the masters. This is the normal state, when multiple masters are on the MS/TP network. The status **Sole Master** is the normal state when the device is the only master on the network. If there are multiple masters on the network, token passing has been interrupted and this state is a hint to a broken cable. In state **Token Lost**, the token is currently not circulating.

While **status** reflects the current state the device is in, the **lost tokens** counter is more indicative for communication problems on the MS/TP network. If it increases, there are cabling, ground, or termination issues.

Note, that the **RX Port** column monitors all packets seen on bus, not only those addressed to the device. Statistics related to received packets that are addressed to the device are tracked in the **MS/TP State/RX** column.


BACnet MS/TP Statistics

Update BACnet statistics | Clear BACnet statistics

MS/TP State/RX		TX Port		RX Port	
status	Token Okay	data packets	2	ok data packets	34
lost tokens	2	no-data pkts	27931	ok no-data pkts	485
tokens	479	tokens	481	invalid pkts	0
poll for master	2	poll for master	27448	not for us pkts	3600
data no reply	34	data no reply	2	tty idle errors	0
data needs reply	0	data needs reply	0	tty prmble errs	0
reply postponed	0	reply postponed	0	tty header errs	0
replies	0			tty data errors	0
reply timeouts	0			timeout prmble	0
token retries	0			timeout header	0
unwanted	0			timeout data	0
unexpected	0			header too long	0
				CRC err header	0
				CRC err data	0

Figure 98: BACnet MS/TP Device Statistics.

The **MS/TP Bus History** (see Figure 99) presents information related to the MS/TP bus as a whole over the last minute, split into 10 second time slices.

The convenient **health** indicator, a percentage in the range 0 – 100%, gives an overall impression of the communication quality on the bus: The higher the percentage, the better the MS/TP communication between devices on the bus. Reasons for **health** to be low are:

- Superfluous PollForMaster requests (because MS/TP node addresses in use contain gaps or Max_Master of the node with the largest node address is not set to the same value as the node's address),
- token losses,
- reply timeouts,
- slow token passing.

The **load** percentage simply displays how much of the available bandwidth is used for data. Note, however, that actual application data is only a subset of the amount of data taken into account here.

Statistics reflecting the average ability of devices to initiate communication are **roundtrip** and **token/dev/sec**. They give an impression on how long the token requires to circulate once (in milliseconds), and how often a device on the bus receives the token per second.

Other counters of interest are: **tk passes** (the number of times the token was passed), **tk misses** (the number of times the receiver of a token did not continue passing the token), **tk retry** (the number of times passing of token was retried), **postponed** (the number of ReplyPostPostponed packets seen), **pfm** (the number of PollForMaster packets seen), **data pkt**, **data pkt rx**, **data pkt tx** (the number of data packets seen, the number of data packets received and transmitted by the device), ***data, data rx, data tx*** (the amount of data seen, the amount of data received and transmitted by the device), **token rx** (the number of tokens received by the device).

Update BACnet statistics | Clear BACnet statistics

Bus History							
09:12:50 - 09:13:00 (0.0sec sole master, 0.0sec sole master activity)							
health	33%	load	0%	roundtrip	120ms	token/dev/sec	8.301
tk passes	753	token rx	83	data pkt	0	data	0byte
tk misses	0	postponed	0	data pkt tx	0	data tx	0byte
tk retry	0	pfm	124	data pkt rx	0	data rx	0byte
09:12:40 - 09:12:50 (0.0sec sole master, 0.0sec sole master activity)							
health	31%	load	0%	roundtrip	126ms	token/dev/sec	7.900
tk passes	707	token rx	79	data pkt	0	data	0byte
tk misses	0	postponed	0	data pkt tx	0	data tx	0byte
tk retry	0	pfm	115	data pkt rx	0	data rx	0byte
09:12:30 - 09:12:40 (1.1sec sole master, 0.0sec sole master activity)							
health	13%	load	0%	roundtrip	104ms	token/dev/sec	9.601
tk passes	554	token rx	96	data pkt	0	data	0byte
tk misses	1	postponed	0	data pkt tx	0	data tx	0byte
tk retry	1	pfm	122	data pkt rx	0	data rx	0byte
09:12:20 - 09:12:30 (10.0sec sole master, 0.0sec sole master activity)							
health	12%	load	0%	roundtrip	0ms	token/dev/sec	0.000
tk passes	38	token rx	0	data pkt	0	data	0byte
tk misses	5	postponed	0	data pkt tx	0	data tx	0byte
tk retry	0	pfm	187	data pkt rx	0	data rx	0byte
09:12:10 - 09:12:20 (10.0sec sole master, 0.0sec sole master activity)							
health	100%	load	0%	roundtrip	0ms	token/dev/sec	0.000
tk passes	0	token rx	0	data pkt	0	data	0byte
tk misses	0	postponed	0	data pkt tx	0	data tx	0byte
tk retry	0	pfm	189	data pkt rx	0	data rx	0byte

Figure 99: BACnet MS/TP Bus History.

The **MS/TP Token History** (see Figure 100) shows the most recent token passes on the bus. The syntax used is simple: '`40<15`' means that the node with address '0x15' has passed the token to the node with address '0x40'.

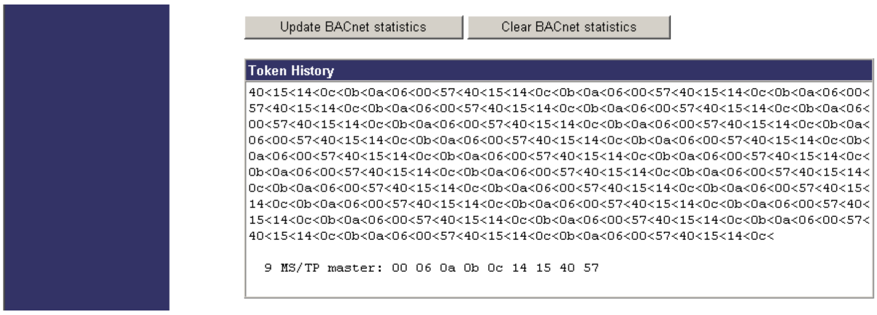


Figure 100: BACnet MS/TP Token History.

If token losses or token sending retries have been recorded, these are marked by substituting '`.`' for '<'. For example, '`40<15.40<15`' either means that '0x15' retried sending the token to '0x40', or that passing the token to '0x40' failed and '0x15' created a new token and sent it to '0x40'.

Transitions to or from sole master mode can be spotted by looking out for '`XX`', e.g., '`XX<15`' means that after '0x15' received the token, the device entered sole master mode. Finally, based on the recorded token passes, the MS/TP Token History also lists the node addresses of MS/TP masters detected on the bus.

5.5.4 CEA-709 Statistics (LDALI-10X only)

The CEA-709 statistics page displays statistics data of the CEA-709 nodes. Each DALI channel corresponds to a separated CEA-709 node. If the device supports multiple DALI channels it is possible to switch between the CEA-709 nodes using the tabs on the top (see

Figure 101). This data can be used to troubleshoot networking problems. To update the data, click on the button **Update CEA-709 statistics**.

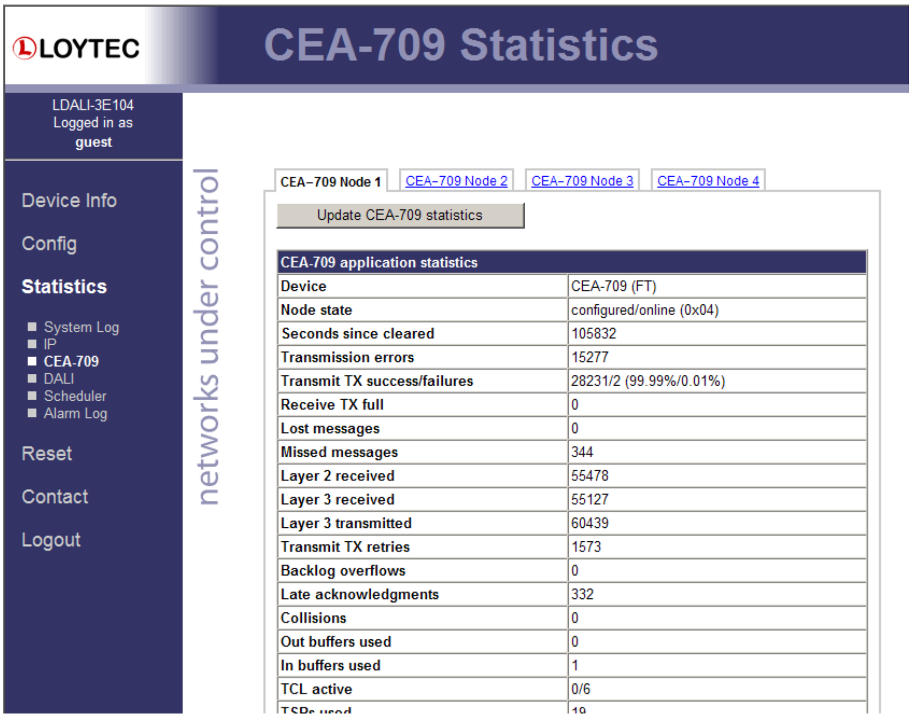


Figure 101: CEA-709 Statistics Page

5.5.5 CEA-852 Statistics

The CEA-852 statistics page displays the statistics data of the CEA-852 device on the L-DALI. It is only displayed if the CEA-852 interface is enabled. The upper part of the CEA-852 statistics page is depicted in Figure 102. To update the statistics data press the button **Update all CEA-852 statistics**. To reset all statistics counters to zero, click on the button **Clear all CEA-852 statistics**. The field **Date/Time of clear** will reflect the time of the last counter reset.

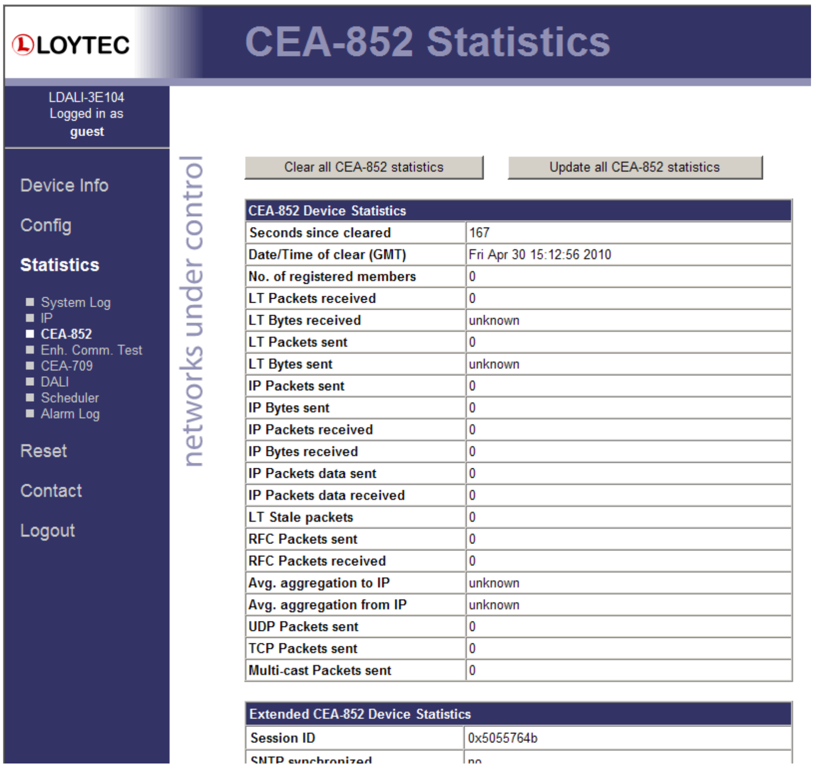


Figure 102: Part of the CEA-852 Statistics Page

5.5.6 Enhanced Communications Test

The Enhanced Communications Test allows testing the CEA-852 communication path between the CEA-852 device on the L-DALI and other CEA-852 devices as well as the configuration server. The test thoroughly diagnoses the paths between individual members of the IP channel and the configuration server in each direction. Port-forwarding problems are recognized. For older devices or devices by other manufacturers, which do not support the enhanced test features, the test passes as soon as a device is reachable, but adds a comment, that the return path could not be tested. A typical output is shown in Figure 103.

The round-trip value (RTT) is measured as the time a packet sent to the peer device needs to be routed back to the L-DALI. It is a measure for general network delay. If the test to a specific member fails, a text is displayed to describe the possible source of the problem. The reasons for failure are summarized in Table 13.

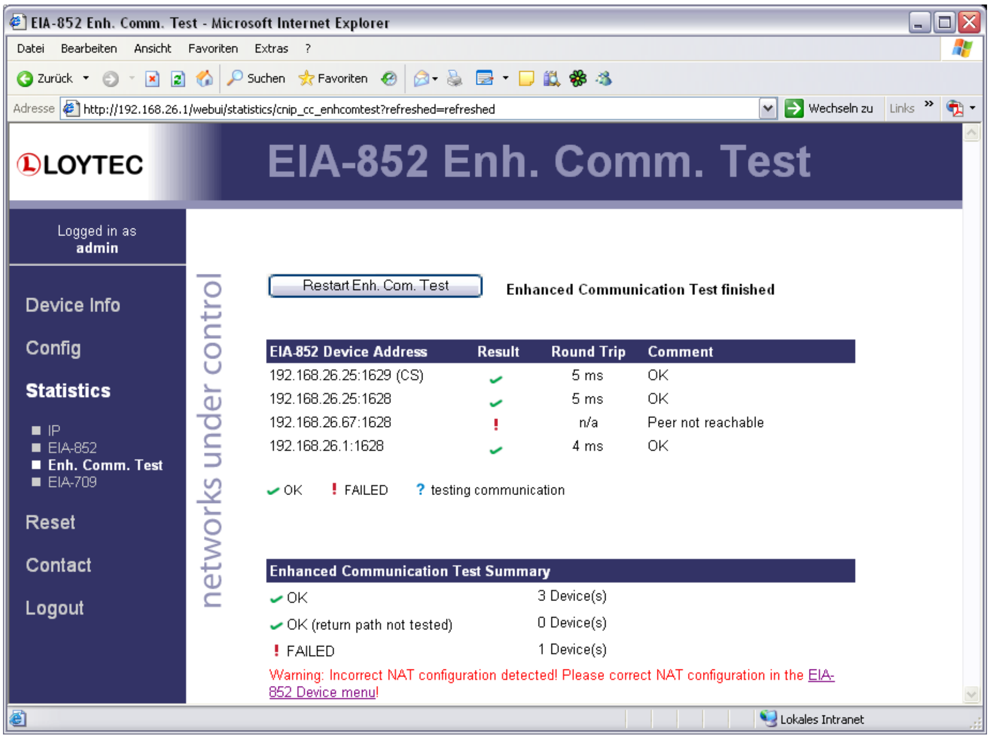


Figure 103: Enhanced Communication Test Output

Text displayed (Web icon)	Meaning
OK, Return path not tested (green checkmark)	Displayed for a device which is reachable but which does not support the feature to test the return path (device sending to this CEA-852 device). Therefore a potential NAT router configuration error cannot be detected. If the tested device is an L-IP, it is recommended to upgrade this L-IP to 3.0 or higher.
Not reachable/not supported (red exclamation)	This is displayed for the CS if it is not reachable or the CS does not support this test. To remove this uncertainty it is recommended to upgrade the L-IP to 3.0 or higher.
Local NAT config. Error (red exclamation)	This is displayed if the CEA-852 device of the L-DALI is located behind a NAT router or firewall, and the port-forwarding in the NAT-Router (usually 1628) or the filter table of the firewall is incorrect.
Peer not reachable (red exclamation)	Displayed for a device, if it is not reachable. No RTT is displayed. The device is either not online, not connected to the network, has no IP address, or is not reachable behind its NAT router. Execute this test on the suspicious device to determine any NAT configuration problem.

Table 13: Possible Communication Problems.

5.5.7 Global Connections Statistics

The global connections statistics page shows all currently configured communication groups. For each group the list displays name, address hash, receive, transmit, poll-on-startup status, the most recently communicated value and its timestamp. An example is shown in Figure 104. The receive/transmit/poll-on-startup status displays an ✕ if the direction is configured, but no value was communicated. A green check mark ✔ is shown as soon as a value was received or transmitted, respectively.

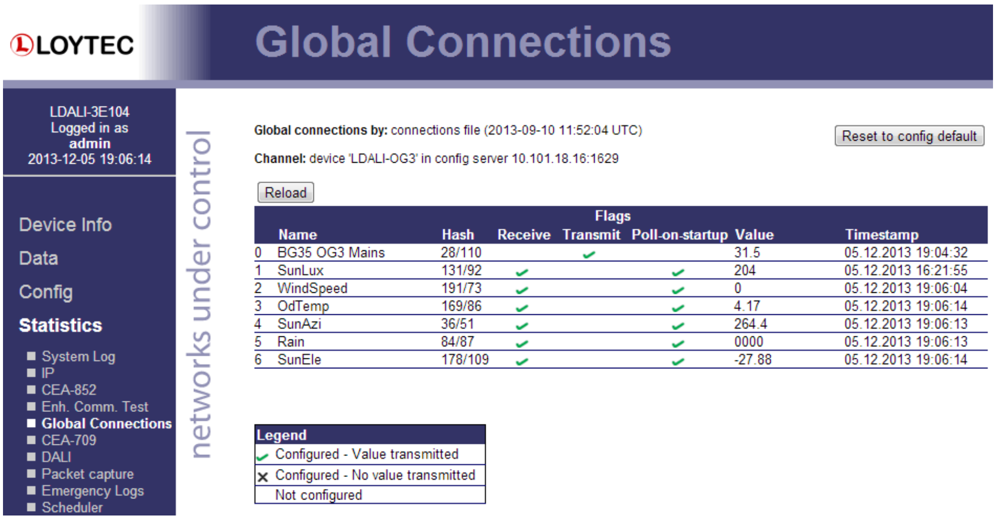


Figure 104: Global Connections Statistics

The **Reload** button refreshes the status. The button **Reset to config default** removes any global connections configured by LWEB-900 at run-time and reverts to the configuration default. A reboot is required in this case.

5.5.8 OPC XML-DA Server Statistics Page

The OPC XML-DA server statistics page shows statistics data, which contains information on currently and previously connected clients. An example list of OPC clients is shown in Figure 105. Clicking on the **Update OPC XML-DA statistics** button retrieves the current statistics.

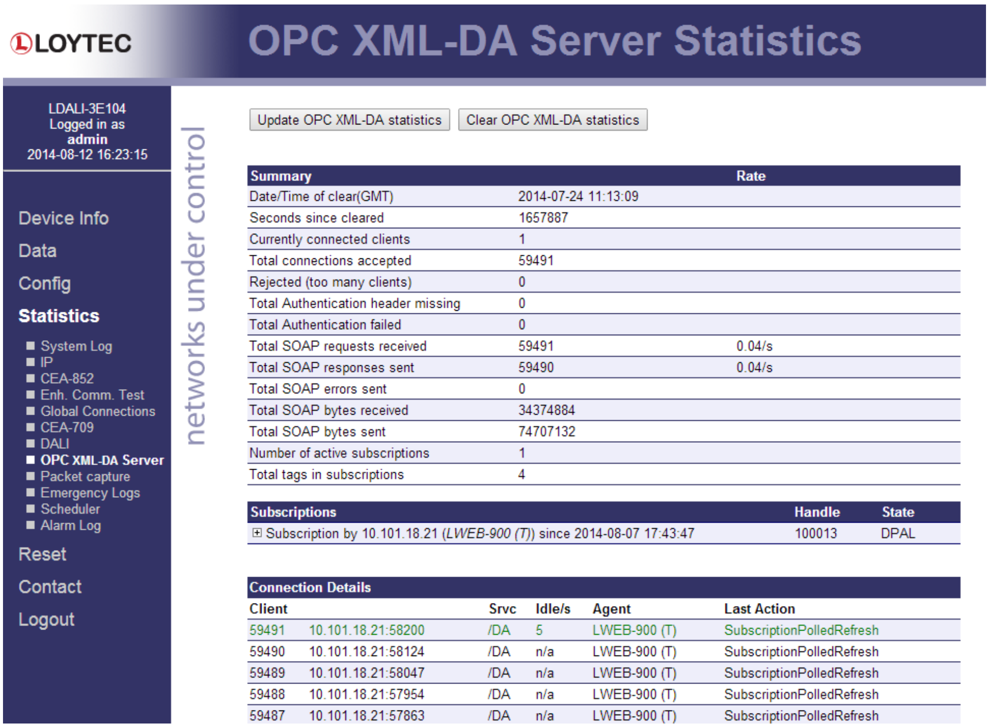


Figure 105: OPC XML-DA Server Statistics Page.

The **Summary** table on the top of the page displays the number of currently connected clients. These clients occupy TCP connections. The next line specifies the total number of accepted client connections since the device is running. The figure for rejected connections can be used to detect situations, where too many clients try connecting at the same time.

The **Connection Details** list shows more information on the history of client connections. The green lines at the top denote currently active connections. Active connections have an idle time figure specified in seconds. The following lines in black represent a history of the most recent connections. Inactive connections read "n/a" in the **Idle** column.

All lines contain client information, which specifies the client IP address and port of the connected client. The **Srvc** column specifies the type of Web service (Web, DA, and DL). The **Agent** column contains information on the HTTP agent of the client, and the **Last Action** column contains information on the last known Web service SOAP action the client has requested.

5.5.9 IP Statistics

Figure 106 shows the IP statistics page. It allows finding possible problems related to the IP communication. Specifically any detected IP address conflicts are displayed (if the L-DALI's IP address conflicts with a different host on the network).

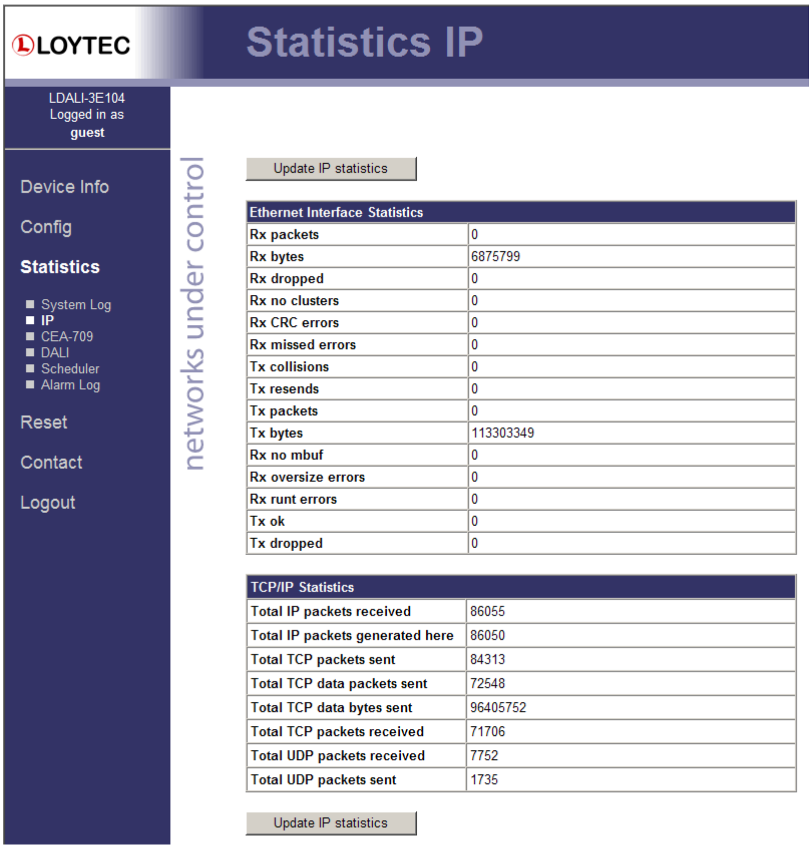
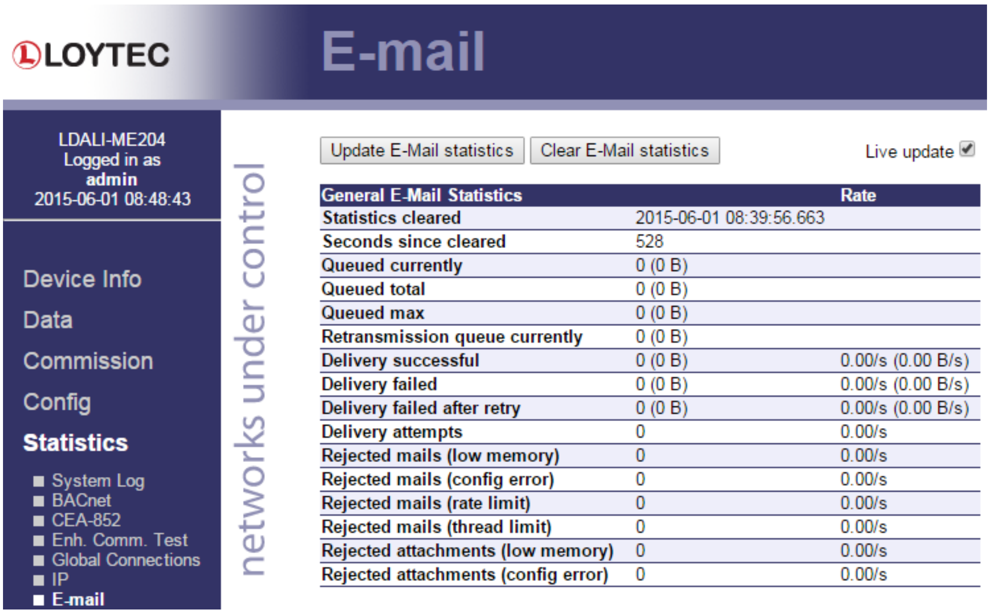


Figure 106: IP Statistics Page

5.5.10 E-mail

The E-mail statistics page shows information regarding the devices SMTP client (e-mail transmission). This includes information regarding the number of messages queued for transmission (Queued currently/total/max), transmitted messages (delivery successful/failed/failed after retry) and dropped messages (with reason for dropping).



General E-Mail Statistics		Rate
Statistics cleared	2015-06-01 08:39:56.663	
Seconds since cleared	528	
Queued currently	0 (0 B)	
Queued total	0 (0 B)	
Queued max	0 (0 B)	
Retransmission queue currently	0 (0 B)	
Delivery successful	0 (0 B)	0.00/s (0.00 B/s)
Delivery failed	0 (0 B)	0.00/s (0.00 B/s)
Delivery failed after retry	0 (0 B)	0.00/s (0.00 B/s)
Delivery attempts	0	0.00/s
Rejected mails (low memory)	0	0.00/s
Rejected mails (config error)	0	0.00/s
Rejected mails (rate limit)	0	0.00/s
Rejected mails (thread limit)	0	0.00/s
Rejected attachments (low memory)	0	0.00/s
Rejected attachments (config error)	0	0.00/s

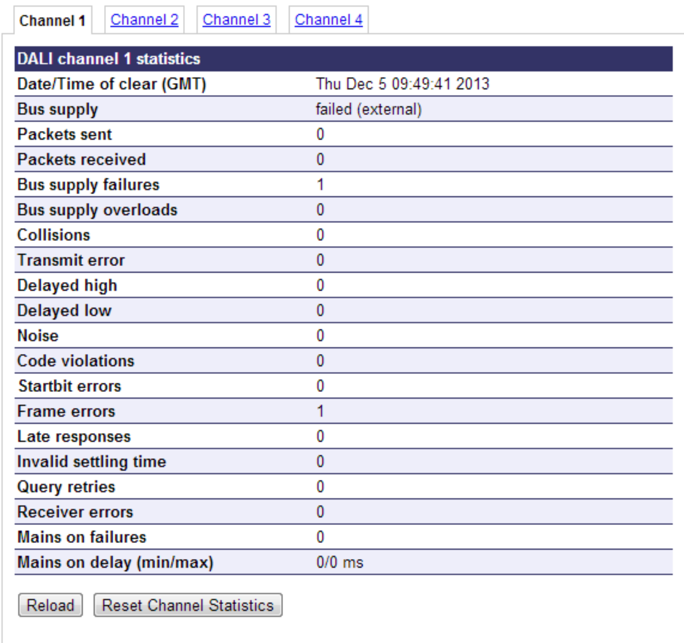
Figure 107: E-mail Statistics.

5.5.11 Packet Capture

The packet capture feature allows configuring and running a local packet capture for the Ethernet port. Please refer to Section 11.6 for more information on how to set up local capture and configure remote packet capture with Wireshark.

5.5.12 DALI Statistics

The DALI statistics page displays the statistics data of the DALI channels. If the device supports multiple DALI channels it is possible to switch between the channels using the tabs on the top (see Figure 108). To update the statistics data press the button **Reload**. To reset all statistics counters to zero, click on the button **Reset Channel Statistics**. The field **Date/Time of clear** will reflect the time of the last counter reset.



DALI channel 1 statistics	
Date/Time of clear (GMT)	Thu Dec 5 09:49:41 2013
Bus supply	failed (external)
Packets sent	0
Packets received	0
Bus supply failures	1
Bus supply overloads	0
Collisions	0
Transmit error	0
Delayed high	0
Delayed low	0
Noise	0
Code violations	0
Startbit errors	0
Frame errors	1
Late responses	0
Invalid settling time	0
Query retries	0
Receiver errors	0
Mains on failures	0
Mains on delay (min/max)	0/0 ms

Figure 108: DALI Statistics

5.5.13 Emergency Logs

This section allows displaying log files containing detailed information on tests executed on DALI emergency light equipment. There is a log file for each group and one for each channel, which contains entries from emergency lights not assigned to a group. Figure 109 shows an example.

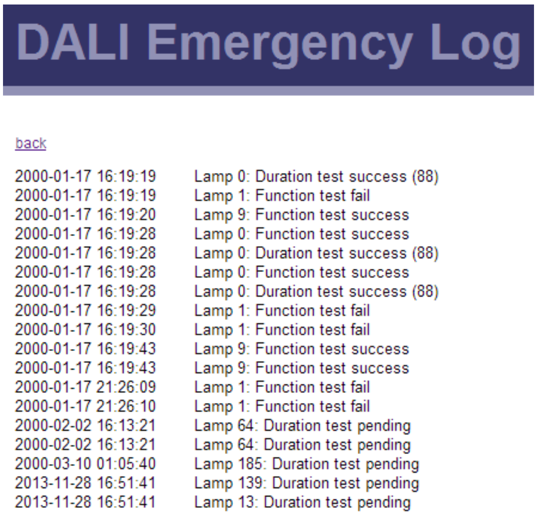


Figure 109: DALI Emergency Log.

5.5.14 Alarm Log Page

The alarm log page provides an overview of all alarm logs on the system. Click on one of the links to view a specific alarm log. Each alarm log contains a historical log of alarm transitions. When an inactive and acknowledged alarm disappears from the alarm summary page (live list), the alarm log contains this last transition and maintains it over a reboot. An example is shown in Figure 110.

To refresh the alarm log contents click on the **Reload** button. Currently active alarms cannot be acknowledged in this historical view. Follow the link to the attached alarm objects to get to the respective live lists, where alarms can be acknowledged on the Web interface (see Section 5.4.5).

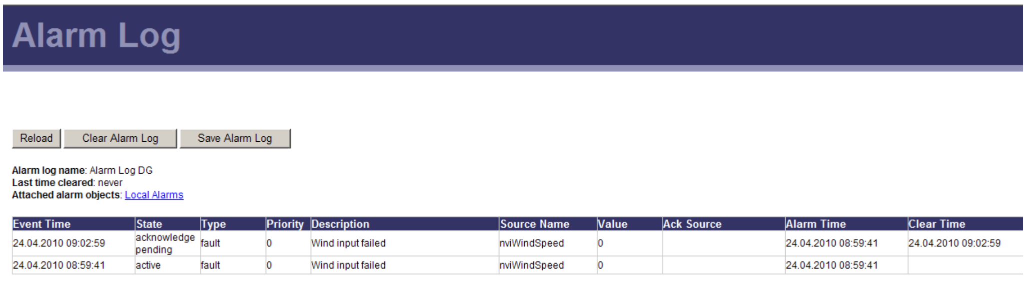


Figure 110: Alarm Log Page.

The alarm log contents can be uploaded from the device in a CSV formatted file. Click on the button **Upload Alarm Log** to upload the current log. To clear the log, press the button **Clear Alarm Log**. Please note, that this permanently purges all historical alarm log data of this alarm log.

5.5.15 Scheduler Statistics Page

The scheduler statistics page provides an overview of what is scheduled at which day and which time. In the **Display Schedules** list select a single schedule to view its scheduled values and times. Use the multi-select feature to get the overview of more schedules. An example is shown in Figure 111.

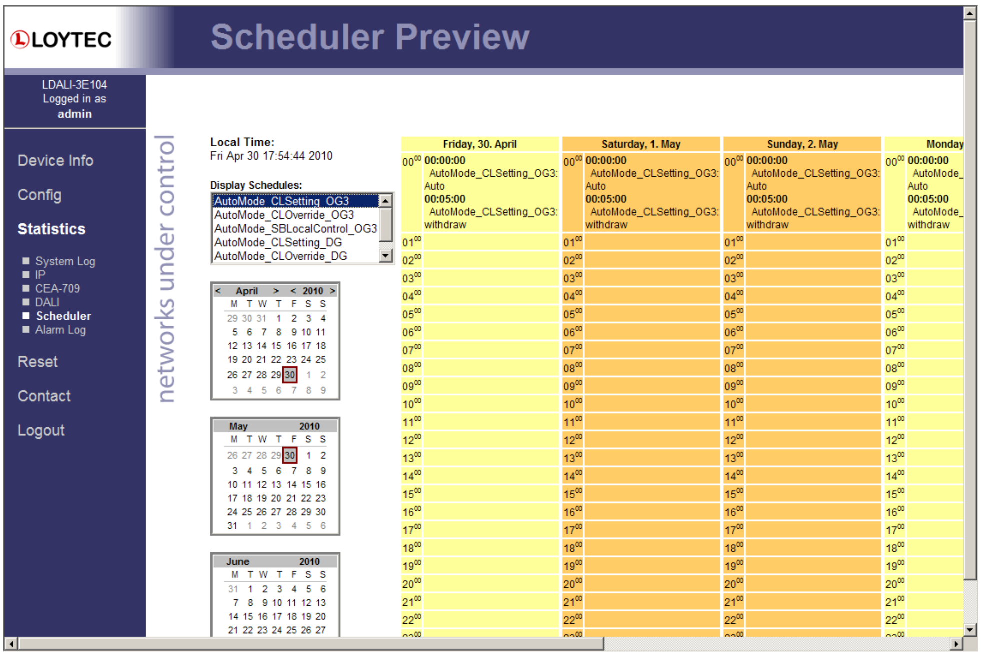


Figure 111: Scheduler Statistics Page

5.6 Documentation

The documentation page allows to access documentation related to the device. See Section 5.2.20 on how to configure documentation links and upload documentation files accessible via this page.

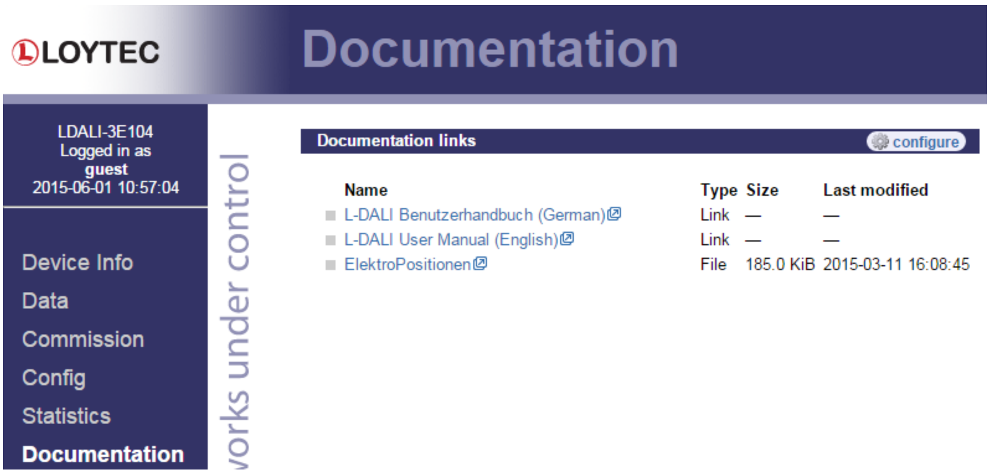


Figure 112: Documentation Page.

Note: *The Documentation page and all files available on it are accessible for all users (incl. Guest).*

5.7 Reset, Contact, Logout

The menu item **Reset** allows the following essential operations:

- Rebooting the L-DALI from a remote location.
- Resetting the data point configuration from a remote location. This option clears all data points incl. parameter values, the entire port configuration, and the DALI configuration stored in the L-DALI. It leaves the IP settings intact.
- Reverting run-time changes to defaults of the data point configuration. This applies to persistent data points, parameter values, and the DALI configuration stored in the L-DALI.

The **Contact** item provides contact information and a link to the latest user manual and the latest firmware version.

The **Logout** item closes the current session.

6 Concepts

6.1 Data Points

6.1.1 Overview

Data points are part of the fundamental device concept to model process data. A data point is the basic input/output element on the device. Each data point has a value, a data type, a direction, and a set of meta-data describing the value in a semantic context. Each data point also has a name and a description. The entire set of data points is organized in a hierarchy.

At the data point level, the specific technological restrictions are abstracted and hidden from the user. Working with different technologies at this level involves common work-flows for all supported technologies.

The direction of a data point is defined as the “network view” of the data flow. This means, an input data point obtains data from the network. An output data point sends data to the network. This is an important convention to remember as different technologies may define other direction semantics. If a data point can both receive and send data on the network, its direction is set to value, indicating no explicit network data flow.

The basic classes of data points are:

- **Analog**: An *analog* data point typically represents a scalar value. The associated data type is a *double precision* machine variable. Meta-data for analog data points include information such as value range, engineering units, precision, and resolution.
- **Binary**: A *binary* data point contains a Boolean value. Meta-data for binary data points includes human-readable labels for the Boolean states (i.e., active and inactive texts).
- **Multi-state**: A *multi-state* data point represents a discrete set of states. The associated data type is a signed integer machine variable. Each state is identified by an integer value, the *state ID*. State IDs need not be consecutive. Meta-data of a multi-state data point includes human-readable descriptions for the individual states (state texts) and the number of available states.
- **String**: A *string* data point contains a variable-length string. The associated data type is a character string. International character sets are encoded in UTF-8. A string data point does not include any other meta-data.
- **User**: A *user* data points contains un-interpreted, user-defined data. The data is stored as a byte array. A user data point does not include any other meta-data. This type of data point also serves as a container for otherwise structured data points and represents the entirety of the structure.

6.1.2 Timing Parameters

Apart from the meta-data, data points can be configured with a number of timing parameters. The following properties are available to input or output data points, respectively:

- **Pollcycle** (input, value): The value is given in seconds, which specifies that this data point periodically polls data from the source. This is referred to as static polling.
- **Receive Timeout** (input, value): This is a variation on the poll cycle. When receive timeout is enabled, the data point must receive a value update within the receive timeout period. If it does not receive a value, a technology may actively poll the source. If no value has been received after another period, the data point is set offline and triggers a fault alarm, if configured. Writing data from any source (network technology, connection, logic program) the receive timeout is reset.
- **Poll-on-startup** (input, value): If this flag is set, the data point polls the value from the source when the system starts up. Once the value has been read, no further polls are sent unless a poll cycle has been defined.
- **Minimum Send Time** (output): This is the minimum time that elapses between two consecutive updates. If updates are requested more often, they are postponed and the last value is eventually transmitted after the minimum send time. Use this setting to limit the update rate.
- **Maximum Send Time** (output): This is the maximum time without sending an update. If no updates are requested, the last value is transmitted again after the maximum send time. Use this setting to enable a heart-beat feature.

Dynamic polling is a feature that some network technologies offer. With static polling the pollcycle is used to permanently poll values over the network. This is required for data points that require constant value updates a fixed pollcycle (for example to trend the data). For other data points that do not need permanent value updates, so-called dynamic polling is activated, as soon as the values are needed (for example displayed on the data pont Web UI or in L-WEB). If dynamic polling is active, the data points are polled using the configured pollcycle. When the data is no longer needed, polling stops and no longer puts a burden on the network. The advantage is that a few data points can be refreshed at a higher rate at a time compared to static polling, where all data points must permanently share the available network bandwidth.

Background polling can be enabled in the project settings. With this feature enabled, all input data points, which rely on polling depending on the underlying network technology, are polled one-by-one in a round-robin fashion. This happens even if no pollcycle is set or dynamic polling is activated on those data points. The frequency of the background polling can be defined in the project settings. The default is 60 polls per minute.

6.1.3 Default Values

Default values can be defined for data points when needed. The value of a data point will be set to the defined default value, if no other value source initializes the data point. Default values are beneficial, if certain input data points are not used by the network and need a pre-defined value, e.g., for calculations. Default values are overridden by persistent values or values determined by poll-on-startup.

6.1.4 Persistency

Data point values are by default not persistent. This means that their value is lost after a power-on reset. There exist different strategies for initializing data points with an appropriate value after the device has started.

For input data points, the value can be actively polled from the network when starting up. Use the Poll-on-Startup feature for this behavior. Polling the network values has the advantage that intermediate changes on the network are reflected. An input data point can be made persistent, if the last received value shall be available after a power-on reset before a poll-on-startup completes. This can be beneficial, if the remote device is temporarily offline and the last value is considered usable.

For output data points, the value can be restored after starting up by the application. For example, if the output data point's value is determined by an input data point and a math object, or the output data point is in a connection with an input, the input can poll its value on startup. If the output data point has no specific other value source, e.g., it is a configuration parameter set by the user, it can be made *persistent*.

To make a data point persistent, enable the Persistent property of the respective data point. The persistency option is only available for the base data point classes analog, binary, multi-state, string and user. More complex objects such as calendars, schedules, etc., have their own data persistency rules. Persistency is also available for unlinked favorites.

For structured data points, only all or none of the structure members can be made persistent. The configuration of the top-level data point, which represents the entire structure, serves as a master switch. Setting the top-level data point to be persistent enables persistency for all sub-data points. Clearing it disables persistency for all sub-data points.

6.1.5 Parameters

A data point can be qualified as a *parameter* data point. This is accomplished in the Configurator software by setting a **Parameter** check box on the data point. Those parameter data points are automatically persistent and will typically have a default value. Their purpose is to store parameterization values, which can be changed from the default value at run time and influence the behavior of the device or the logic running on the device. This way, a number of devices can have the same basic configuration and be adapted by parameter values. Examples are sunblind run times for control logic or descriptive strings for the L-WEB visualization.

The qualified parameter data points are also exported via a parameter file, which contains the entire set of current parameter values including meta-information for external tools to display parameter data in a human-readable way. The LWEB-900 parameter view can process such parameter data points and manage them for a large number of devices. For more information on how to manage parameters on your devices please refer to the LWEB-900 manual [5].

6.1.6 Behavior on Value Changes

The value of a data point can change, if it is written by the application or over the network. For all data points (input, output and value) the application (connection, user control, etc.) can be notified, when the value is written to. The property **Only notify on COV** defines, whether the notification is done with each write or only if the value changes (change-of-value, COV). If only notify on COV is disabled, writing the same value multiple times will result in multiple notifications.

When the value of an output data point is updated, an update is usually sent out onto the network. The property **Send-On-Delta** decides how the update is reflected on the network. If send-on-delta is inactive, each update of the value is sent, even if the value does not change. If send-on-delta is active, only value changes are sent. The send-on-delta property is only valid for output data points.

For analog data points, the COV or send-on-delta takes an extra argument, which specifies by what amount the value must change to regard it as a change for action. Both, COV and send-on-delta for analog data points check the **Analog Point COV Increment** property. A change is detected, if the value increment is bigger or equal to the specified increment. If the property is '0.0', all updates are reported, even if the value does not change.

Data point usages, such as COV trend logs or math objects may specify their own COV deltas on analog data points. These can be bigger than the data point COV itself, but never smaller.

6.1.7 Custom Scaling

Custom scaling is applied to all analog data points when they communicate values to or from the network. This feature can be used, if a network data point has engineering units not suitable for the application (e.g., grams instead of kilograms). The scaling is linear and applied in the direction from the network to the application as:

$$A = k\,N + d,$$

where N is the network value, *k* the *custom scaling factor*, *d* the *custom scaling offset*, and A the application value. When sending a value to the network, the reverse scaling is applied. If this property is enabled, the analog values are pre-scaled from the technology to the data point. The custom scaling is in addition to any technology-specific scaling factors and can be applied regardless of the network technology.

6.1.8 Protected Data Points

Some data points are created automatically depending on the model currently selected. They are protected against manipulation by the user. Therefore they cannot be deleted or moved and their properties cannot be modified. System registers (see Section 6.1.9) fall into this category. In addition some models (e.g. L-DALI) come with a predefined interface which cannot be changed either.

6.1.9 System Registers

The device provides a number of built-in system registers. They are present without a data point configuration. The system registers, such as the System time or the CPU load, can be exposed to the OPC server. By default, all system registers are checked for being exposed to OPC. To reduce the number of needed OPC tags, you may deselect certain system registers, which are not useful in a specific project.

System registers are read-only by default. System register can also serve as a testing setup for the OPC XML-DA communication without a network data point configuration. The *System Time* register is updated every second and may serve for testing subscriptions. The *Authentication Code* register can be used to verify writing to OPC tags.

The available system registers and a short description of their function are listed below:

- **System Time**: This register is an *analog* data point. It supplies the system time of the local clock in UTC as seconds since 1.1.1970. It increments each second. Example: 1302533716.
- **Time UTC**: This register is a structured data point. It supplies the system time as UTC broken down to year, month, day, hour, minutes and seconds.
- **Time Local**: This register is a structured data point. It supplies the system time as local time broken down to year, month, day, hour, minutes and seconds.
- **CPU Load**: This register is an *analog* data point. It displays the average system CPU load in percent over the last minute. Example: 17 %.
- **Free Memory**: This register is an *analog* data point. It displays the current amount of free RAM memory in Bytes. Example: 20522288 Bytes.
- **Free Flash**: This register is an *analog* data point. It displays the current amount of free memory in Bytes of the Flash storage. Example: 8482688 Bytes.
- **Supply Voltage**: This register is an *analog* data point. It displays the currently measured supply voltage in volts. Example: 15.1 V.

- **System Temp**: This register is an *analog* data point. It displays the currently measured system temperature in degrees Celsius. Example: 39 °C.
- **Application Vendor, Authentication Code,** and **Authentication Result**: These registers can be used to implement an IP protection mechanism for application programs, such as IEC61131 programs.
- **Serial Number**: This register is a *string* data point. It displays the device's serial number as an ASCII string. Example: "011401-000AB001D1E4".
- **MAC Address**: This register is a *user* data point. It displays the device's MAC address as an array of 6 hexadecimal Bytes. Example: 000AB001D1E4.
- **Firmware Version**: This register is a *string* data point. It displays the device's firmware version as an ASCII string. Example: "4.1.0".
- **Device IP Address**: This register is a *string* data point. It displays the device's IP address as an ASCII string. Example: "10.101.18.204".
- **Device IP Port**: This register is an *analog* data point. It displays device's HTTP port as an integer value. Example: 80.
- **TZ Offset**: This register is an *analog* data point. It displays the time zone offset relative to UTC in seconds. This means a positive value for a time zone, which lies east of Greenwich. The offset includes daylight savings time. The local time can be derived by adding this register to the system time register. Example: +7200 for GMT+1 (Paris, Berlin, Vienna) including DST.
- **Device Status**: This register is a *string* data point. It contains an XML document with the device status file contents. It is not displayed on the Web UI.
- **Ethernet Link Mask**: This register is a *multistate* data point. It displays the link information of the Ethernet port. Example: "Eth 1".
- **Hostname**: This register is a *string* data point. It displays the host name, which has been configured in the IP settings. Example: "my_linx".
- **Position Longitude**: This register is an *analog* data point. It displays the longitude part of the device's location in degrees. Writing to the corresponding data point **Position Longitude_Set** sets the device's longitude in degrees. Example: -16.33472.
- **Position Latitude**: This register is an *analog* data point. It displays the latitude part of the device's location in degrees. Writing to the corresponding data point **Position Latitude_Set** sets the device's latitude in degrees. Example: 48.22056.
- **Position Altitude**: This register is an *analog* data point. It displays the altitude of the device's location in meters above sea level. Writing to the corresponding data point **Position Altitude_Set** sets the device's altitude in meters above sea level. Example: 200 m.
- **Secure Mode**: On models providing a firewall, this binary register enable the firewall to restrict access to the services provided in the **Secure Services** register.
- **Secure Services**: On models providing a firewall, this string register selects the services which should be available when **Secure Mode** is **TRUE**. This registers accepts a space-separated list of service names. If the selected services would make the device unconfigurable, a default configuration with HTTPS and SSH enabled is selected. The available service names are:
  - **HTTP**: Enables access to the configuration pages via HTTP.
  - **HTTPS**: Enables access to the configuration pages via HTTPS.
  - **SSH**: Enables access to the SSH server.
  - **OPC**: Enables access to the OPC XML-DA server.
  - **OPCUA**: Enables access to the OPC UA server.

- **ICMP**: Allows incoming ICMP packets (recommended).

6.1.10 User Registers

The device can be configured to contain user registers. In contrast to system registers, these are only available as a part of the data point configuration. User registers are data points on the device that do not have a specific technological representation on the control network. Thus, they are not accessible over a specific control network technology.

A register merely serves as a container for intermediate data (e.g., results of math objects, calculation parameters). The register can have the following, basic data types:

- **Double**: A register of base type *double* is represented by an *analog* data point. It can hold any scalar value. No specific scaling factors apply.
- **Signed Integer**: A register of base type *signed integer* is represented by a *multi-state* data point. This register can hold a set of discrete states, each identified by a signed stats ID.
- **Boolean**: A register of base type *Boolean* is represented by a *binary* data point. This register can hold a Boolean value.
- **String**: A register of base type *string* is represented by a *string* data point. This register can hold a variable-length character string in UTF-8 format.
- **Variant**: A register of base type *variant* is represented by a *user* data point. This register can hold any user-defined data of up to a specified length of Bytes. This length is defined when creating the register and cannot be changed at run time.

Since a register has no network direction, it can be written and read. Therefore, it is created as a value data point by default. It is also possible to create two data points for each register, one for writing the register (output) and one for reading the register (input). In this case a suffix is added to the register name to identify the respective data point. For example, the register *MyValue* will have two data points generated for: *MyValue_Read* and *MyValue_Write*.

6.1.11 Structures

Complex data belonging semantically together may be structured. The data point model allows mapping structure types onto user-defined data points of *variant* type. This can be necessary, if a network technology carries such structured data or if a user-defined register shall provide structured data for access through a single data point. In any case, the structure is modeled as a top-level data point and a hierarchy of sub-data points representing the structure members.

The top-level data point is a user data point of variant data type. It contains the image of the entire structure as a Byte array. Each structure field is then modeled as a sub-data point of the appropriate class (e.g. analog, binary, or multi-state). A structure field may itself be a structure going down one level in the hierarchy of sub-data points.

An example is shown in Figure 113. In this case a user register of two Bytes is bound to a structure type mapping the two bytes on analog data points. The two sub-data points *byte_0* and *byte_1*.

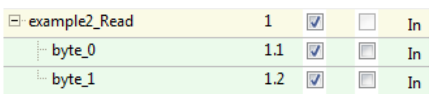


Figure 113: Example of a structured data point.

The structure types are available in a type repository with the Configurator. This repository is divided into scopes. Within each scope a type has a unique name. When selecting a type, the scope and the type name needs to be specified.

Favorites can also be structured. A structured favorite can be created by dragging a structured data point into the favorites folder. As a default, the structure top is linked to the structure top of the target data point while all sub-element are linked to their respective target sub-elements. It is also possible to unlink the structure top and link its sub-elements to different individual data points. When entirely unlinked, the structured favorite behaves like a structured user register.

6.1.12 Property Relations

A data point possesses a number of properties, which influence the behavior and appearance of the data point. Examples are data point name, poll cycle or alarm limits. Most of those properties are determined by the configuration and are static during operation of the device. Some of those properties, however, shall get a default value from the configuration and be modified during run-time. Modification may be carried out by the user by setting the property value over the Web UI or by L-WEB over the Web service.

In some cases property values shall also be updated by other data points, e.g. a user register or a technology data point. In this case the data point property is linked to another data point following a given, semantic relation. This is modeled as a *property relation*. Property relations appear as data point links with the respective property names underneath their governing data point. An example is shown in Figure 114. They are marked with a link symbol . When hovering with the mouse over the link symbol, a bubble help appears describing the property relation.

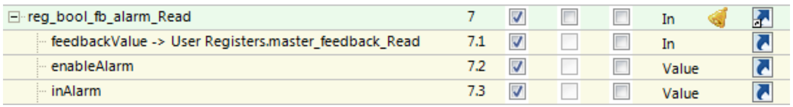


Figure 114: Example of property relations.

The property relations can be accessed like regular sub-data points from the Web UI or by L-WEB over the OPC web service. For this usage, no linkage against other data points is necessary. Property relations may, however, also be linked to other data points, e.g. ‘feedbackValue’ in Figure 114. In this case the linked data point is used as the related property. The user may right-click on a linked property relation and choose **Go to related data point** from the context menu. For mass engineering property relation links to other data points refer to Section 7.7.8.

The following properties are available as property relations:

- **feedbackValue:** This property relation is used for feedback alarm conditions. The data point value is compared against the feedback value. An alarm is generated, if these values differ (by a certain amount). It exists only, if an alarm condition has been created.
- **enableAlarm:** This property relation is used to enable or disable alarm generation on the data point. It exists only, if an alarm condition has been created.
- **inAlarm:** This property relation is TRUE, if the data point is in an alarm. It exists only, if an alarm condition has been created.
- **ackPend:** This property relation is TRUE, if the data point’s alarm needs acknowledgement. It exists only, if an alarm condition has been created.
- **highLimit:** This property relation defines the high limit for analog alarms. It exists only, if an alarm condition has been created.

- **lowLimit:** This property relation defines the low limit for analog alarms. It exists only, if an alarm condition has been created.
- **deadband:** This property relation defines the dead band for analog alarms. It exists only, if an alarm condition has been created.
- **nativeAlarm:** This property relation links to a technology data point, which is required for alarms reported to another technology. It exists only, if an alarm condition has been created and alarms are reported to the given technology (e.g. BACnet). This property relation cannot be modified by the user.
- **reportTo:** This property relation exists only in generic alarm servers. It may be linked to a technology alarm server to report alarms to that network technology.
- **totalActive, totalUnacked, totalAcked:** These property relations exist only in alarm servers. They contain counters for active unacknowledged, inactive unacknowledged, active acknowledged alarm records of the alarm server, respectively.
- **ackAll:** This property relation exists only in alarm servers. When writing TRUE, all alarms on that alarm server are acknowledged.
- **historicFilter:** This property relation exists for data points that have at least one historic filter assigned (see Section 6.4.6).
- **enable:** For Schedulers can be enabled or disabled by the use of this property relation. If the enable data point has been defined in the scheduler's configuration, this property relation is linked to that data point.
- **enableFb:** This property relation shows the enable state of the scheduler. If the enable feedback data point has been defined in the scheduler's configuration, this property relation is linked to that data point.
- **presetName:** This property relation of type string shows the preset name of the currently scheduled value. If the preset name data point has been defined in the scheduler's configuration, this property relation is linked to that data point.
- **timeToNext:** This analog property relation contains the number of minutes till the next scheduled state changes. It can be used to implement an optimum start algorithm.
- **nextState:** This property relation contains the next scheduled state. It can be used to implement an optimum start algorithm.
- **nextPresetName:** This property relation contains the preset name of the next scheduled state, if such name exists. Otherwise it stays at invalid value.

6.1.13 Convertible Engineering Units

Analog data points possess the engineering units property that defines the physical unit of the underlying scalar value, e.g., "kg". The engineering unit is displayed with a human-readable text to the user. The text can be freely entered by the user to describe the nature of the scalar value. The Configurator matches this text against its database of known engineering units. If it can identify the unit, it is denoted as a *convertible unit* with a green checkmark ✅.

Convertible units are linked to additional meta-information in the metric (SI) or U.S. unit system. For those units the Configurator can offer compatible units of the respective other unit system, of different magnitude or different usage. Important properties of convertible units when used are:

- An alternative display unit can be configured that will be displayed on the Web UI of the device.
- Automatic unit conversion in local connections is performed, if data points with compatible convertible units are connected (e.g. '1000 W' are converted to '1 kW'). No custom scaling is required.

- Auto-generated data points in connections are created such that they have a best-matching unit in their target technology (e.g., the best-matching SNVT is created out of a BACnet data point of a certain engineering unit).

6.2 Math Objects

6.2.1 General Properties

Math objects are advanced application objects that can execute mathematical operations on data points. A math object takes a number of input data points (variables v_1, v_2, …, v_n) and calculates a result value according to a specified formula. The result is written to a set of output data points. The formula is calculated each time one of the input data points updated its value. The formula is only evaluated if all of the input data points have a valid value (i.e., don't show the *invalid value* status).

Using the assigned variable names, immediate values, parenthesis, operators and function names, the user can enter a formula in the usual way (infix notation). Apart from the functions in the next Section, the short-hand operators `+`, `–`, `/`, `*`, `%`, `AND`, `OR`, `XOR`, `^`, `&`, `|`, `=`, `!=`, `<`, `>`, `<=`, `>=` can be used directly. Further, it is possible to use parenthesis to define the precedence of the operations.

Example: `(v1 + v2) * sqrt(pow(v3,0.1))`

Note: *As usual practice in programming languages, the comma is used to separate arguments in expressions and the **decimal point** is used in decimal values. The expression `sum(4,5)*2` evaluates to 18, while `sum(4.5)*2` evaluates to 9.*

As you enter the formula, it will be parsed and the resulting sequence of calculations will be displayed in a list at the right of the property page. This list shows your formula in reverse polish notation (RPN), also known as postfix notation, as used by many scientific pocket calculators.

6.2.2 Usage Hints

A few functions end with a … (three dots) in the argument list. This means that they accept a variable number of arguments. When used in the formula, they will fetch all available values from the stack and then calculate the result, which will be put back on the stack and be the only value on the stack, since all other values were used as input to the function.

This behavior causes some limits in how these functions may be used. You are on the safe side, if you use such a function only as the outermost function (infix), or as the last function on the stack (postfix) for example:

`sum(v1, v2, exp(v3, -1))`

Or the postfix equivalent: `v1, v2, v3, -1, exp, sum`

If you have to use it as an argument to another function, it may only be the first argument; otherwise the formula cannot be processed by the math object, which internally uses an RPN machine, with precompiled instructions for optimal performance. Example:

`add(avg(v1, v2, v3), 5)` or `avg(v1, v2, v3)+5` will work.

`add(5, avg(v1, v2, v3))` or `5 + avg(v1, v2, v3)` will NOT work.

Another property of those functions is that they ignore input values, which have the invalid value. Therefore, assuming v1=5, v2=invalid, v3=3 the calculation `add(v1,v2,v3)` evaluates to 8 while `v1+v2+v3` returns invalid. This can be used to purposely allow inputs in the calculation that have no value.

To limit the number of re-calculations, the data point option **Only Notify on COV** should normally be checked on all connected input data points. This avoids recalculating the formula and writing a value to the output data point when it is already clear that the result will be the same, because the input value did not change. The same option can also be checked for the output data point to avoid unnecessary writes to the output data point, in case the inputs changed but the result of the formula is still the same.

6.2.3 Function List

The currently supported math function calls are listed in Table 14.

Function	**Return Value**
add(v1,v2)	v1 + v2
sub(v1,v2)	v1 - v2
mul(v1,v2)	v1 * v2
div(v,d)	v / d
mod(v,m)	Returns the remainder of dividing v by m, where v and m should be integer values. Fractional values will be rounded to the nearest integer automatically
max(v1,...)	Returns the maximum of all values on the value stack
min(v1,...)	Returns the minimum of all values on the value stack
avg(v1,...)	Returns the arithmetic mean value of all values on the stack
log(v)	Returns the natural logarithm of v
log2(v)	Returns the base 2 logarithm of v
log10(v)	Returns the base 10 logarithm of v
exp(v)	Returns the value of e (the base of natural logarithms) raised to the power of v
exp2(v)	Returns the value of 2 raised to the power of v
exp10(v)	Returns the value of 10 raised to the power of v
sqrt(v)	Returns the non-negative square root of v
pow(v,exp)	Returns the value of v raised to the power of exp
round(v)	Round v to the nearest integer
floor(v)	Round v down to the nearest integer
ceil(v)	Round v up to the nearest integer
sum(v1,...)	Returns the sum of all values on the stack
and(b1,b2)	logical AND of the Boolean values b1 and b2 (b1&&b2)
or(b1,b2)	logical OR of the Boolean values b1 and b2 (b1||b2)
xor(b1,b2)	logical exclusive OR of the values b1 and b2 (b1^b2)
not(b)	logical inverse of the Boolean value b (!b)
lt(v1,v2)	returns 1 if v1 is lower than v2, else returns 0 (v1 < v2)
le(v1,v2)	returns 1 if v1 is lower or equal v2, else 0 (v1 <= v2)
eq(v1,v2)	returns 1 if v1 equals v2, else 0 (v1 = v2)
ge(v1,v2)	returns 1 if v1 is greater or equal v2, else 0 (v1 >= v2)
gt(v1,v2)	returns 1 if v1 is greater than v2, else 0 (v1 > v2)
if(b,vt,vf)	returns vt if b is true, else returns vf (b ? vt : vf)
encode(b1,...)	Reads all values from the stack, converts them to Boolean values and encodes them into an integer value, where the first value is used as the LSB and the last value as the MSB.
sin(v1)	Returns the sine of v1, where v1 is given in radians
cos(v1)	Returns the cosine of v1, where v1 is given in radians
tan(v1)	Returns the tangent of v1, where v1 is given in radians
sinh(v1)	Returns the hyperbolic sine of v1, which is defined mathematically as (exp(v1) - exp(-v1)) / 2
cosh(v1)	Returns the hyperbolic cosine of v1, which is defined mathematically as (exp(v1) + exp(-v1)) / 2
tanh(v1)	Returns the hyperbolic tangent of v1, which is defined mathematically as sinh(v1) / cosh(v1)
asin(v1)	Returns the arc sine of v1; that is the value whose sine is v1
acos(v1)	Returns the arc cosine of v1; that is the value (in radians) whose cosine is v1
atan(v1)	Returns the arc tangent of v1; that is the value (in radians) whose tangent is v1
asinh(v1)	Returns the inverse hyperbolic sine of v1; that is the value whose hyperbolic sine is v1
acosh(v1)	Returns the inverse hyperbolic cosine of v1; that is the value whose hyperbolic cosine is v1
atanh(v1)	Returns the inverse hyperbolic tangent of v1; that is the value whose hyperbolic tangent is v1
gamma(v1)	Returns the value of the Gamma function for the argument v1. The Gamma function is defined

Function	Return Value
	by Gamma(x) = integral from 0 to infinity of t^(x-1) e^-t dt. It is defined for every real number except for no positive integers. For nonnegative integral m one has Gamma(m+1) = m! and, more generally, for all x: Gamma(x+1) = x * Gamma(x) For x < 0.5 one can use Gamma(x) * Gamma(1-x) = PI/sin(PI*x)
abs(v1)	computes the absolute value of the argument v1

Table 14: Available math functions.

6.3 Connections

6.3.1 Local Connections

With the use of connections data points can interact with each other. Connections specify which data points exchange values with each other. Various types of connections – from "1:*n*" to "*m*:*n"* connections – are supported. Data points added to a connection specify whether they feed a value into the connection (send) or they receive a value from the connection (receive).

This means, the following connections are possible:

- 1 input data point is connected and writes to *n* output data points,
- *m* input data points are connected and write to 1 output data point,
- *m* input data points are connected and write to *n* output data point.

The most common connection will be the 1:1 connection. This is the type of connection that is auto-generated by the Configurator software. Other types must be created manually or by a template in the Configurator.

In the 1:*n* connection the input value is distributed to all *n* output data points. In the *m*:1 connection, the most current input value is written to the output data point. When polling the output data point in poll-through mode (maximum cache age is set on the output), the value from the first input data point is polled. The same holds true for a *m*:*n* connection. The default data flow of data points in a connection is a result of the data point direction. This can be overridden by a custom setting (i.e. an output data point can be configured as an input to the connection).

Connections can connect data points of different technologies with each other (also mixed among the target data points). When connecting data points of different classes the exchanged values need to be converted. The connection inherits the type of the first data point class. If data points of a different class are added to this connection, an *adaptor* needs to be defined. For example an analog value connection has a multi-state output data point. Adaptors can be saved in a library and re-used later for similar conversions.

The following conversions apply:

- **Analog to Analog**: The value range is capped on the output data points. This means, if the input value in the hub does not fit into the range of an output data point, the value is capped to the biggest or smallest allowed value. If the input and output data points both have convertible units the value is converted. The user can also specify a simple math formula as an adaptor. In this case no implicit unit conversion is performed.
- **Binary/Multi-state to Analog**: The Boolean or state value is directly converted to an analog value (e.g. state ID '4' is written as '4.0') as a default. The user should specify an adaptor to map the Boolean or state value to designated analog values.

- **Analog to Binary/Multi-state**: As a default the analog value is converted to the next Boolean or state value (e.g. '1.2' is written as state ID '2'). The user should specify an adaptor with its own translation of value ranges to state values.
- **Multi-state to Multi-state**: Multi-state data points that have different state maps lead to a conversion of their state values. The state maps of inputs and outputs are ordered by state ID in ascending order. The state value of the input is then ranked as the n-th state and propagated over the connection. For example, the input state ID '1' is the 2nd state and the output's 2nd state has the state ID '0'. If the output data point has less states than the input, the output state is limited to its highest state ID. The user should specify an adaptor that defines which input state maps to which output state.
- **Binary to Binary**: Binary data points can be connected without conversion.
- **String to String**: String data points can only be connected to string data points.
- **User to User**: User data points can only be connected to user data points. If the length is different, only valid bytes are written or excess bytes are truncated, respectively.
- **SNVT_switch to Analog/Binary/Multi-state**: The user data point of a SNVT_switch can be connected to analog, binary, and multi-state data points.
- **Analog/Binary/multi-state to SNVT_switch**: Analog, binary, and multi-state data points can be directly connected to a SNVT_switch user data point.

6.3.2 Multi-Slot Connections

Connections between structured data points often need to connect each structure member separately. To increase the overview in the project on the involved, single connections, a *multi-slot connection* can be created for local connections. This is a connection with several slots for transporting separated values over the connection. Each slot has a number and a name and can connect two or more data points. Data points added to other slots do not share their values across slots. One can think of such a connection as a cable with many wires. An example is shown in Figure 115 (a). The data point 'IN A' sends its value to 'OUT X' but not 'OUT Y'.

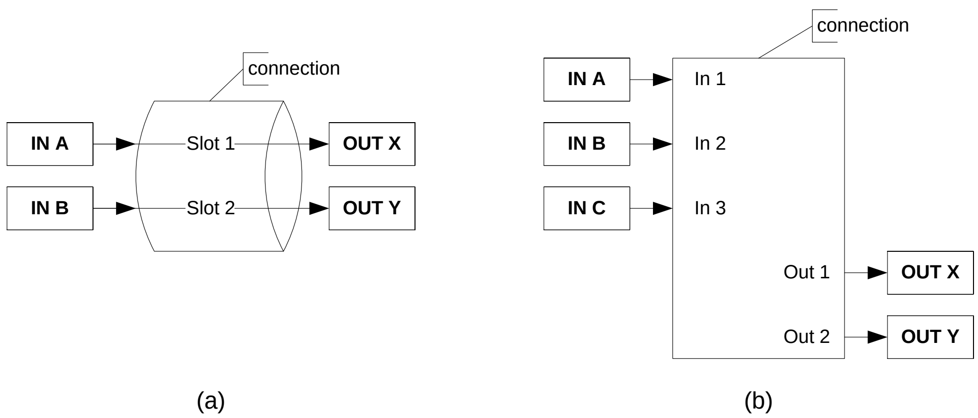


Figure 115: Multi-slot connection (a) and multi-slot with math block adaptor (b).

Some gateway applications also require a functional mapping between different data point structures in one connection. A multi-slot connection can be used with a math block adaptor to accomplish this task. A math block has *n* inputs and *m* outputs. The multi-slot connection has a slot for each input and output, which can be connected to the respective data points as depicted in Figure 115 (b). For this multi-slot connection the math block adaptor defines a fixed layout of the slots; no more slots can be added to this connection.

In a math block adaptor with *n* inputs $v_1, v_2, \dots, v_n$ each output o_i is calculated as a formula depending on all inputs $o_i = f_i(v_1, v_2, \dots, v_n)$. Each output has two math formulae following the same format as used in math objects (see Section 6.2):

- Output value formula: This formula calculates the output value as a function of all input values.
- Output enable formula: This formula calculates an output enable (result > 0 is enable) for the output. If the output is enabled, the output value will be written to the output. If the output is disabled, the calculated output value is not written to the output.

In addition, each input slot can be configured whether it shall trigger the calculation or not. Normally, any change in any input triggers the calculation of all outputs.

6.3.3 Automatic Generation and Templates

In a gateway application the systems engineer has a typical workflow: He will be confronted with some network equipment of one technology that needs to be exposed to another network technology. The task of generating the counterparts of data points in another technology and connecting them is covered by the *smart auto-generate and connect* method. The existing data points are called *sources* and the generated data points are called *targets*.

In principle, the Configurator supports auto-generate for all source technologies but generation is limited to select target technologies. Depending on availability on the device model, the following technologies can be target for auto-generation:

- CEA-709 (static NVs),
- BACnet (server objects),
- Registers,
- Modbus (slave registers).

The target data point is generated with opposite direction and of the same class as the source data point. Depending on the target technology, however, certain restrictions apply on what can be generated. Typical issues are engineering units, state maps and data point structures. The folder structure of the source data points is replicated for the target data points.

For example, when generating matching counter parts to NVs, there are two types of NVs to be considered: Simple NVs that hold only one value (scalar or enumeration), and structured NVs, that consist of a number of fields. For simple NVs only one BACnet object per NV is generated. For structured NVs, one BACnet object is generated for each structure member. This method is called structure flattening. Some target technologies do support structures and no flattening is applied. When generating an analog target, a data point with the best-matching engineering unit is created. If the target allows arbitrary engineering units this will be the same as the source engineering unit. If the target has only a limited number of engineering units, the technology object with the best-matching unit is created. Multi-state target data points are created with an equal number of states and compatible state IDs. For example the CEA-709 state IDs are sorted and renumbered to start at ‘1’ in BACnet (i.e., a ‘-1’ of MOTOR_NUL in CEA-709 maps to a ‘1’ of MOTOR_NUL in BACnet). This is necessary as the SNVT states ‘-1’ and ‘0’ cannot be represented in BACnet as a raw value, because allowed BACnet multi-states start at 1.

The Configurator provides a preview dialog that shows, which target data points will be created. Thus, the implicit generation rules are visible to the user. If the target technology provides several options on what to generate, the user can change the default in this dialog.

The setting is stored in the project and will be applied again with the next generation. The project settings also provide defaults for auto-generation. How exactly data points are created depends on the target technology. Refer to the technology sections for more information how data points are used in connections.

For more advanced connection tasks that involve specific adaptors *auto-generate templates* must be used. An auto-generate template contains the source data point, the desired target data point and the local connection with all appropriate adaptors. There are two types of auto-generate templates:

- **Simple auto-generate template**. This template contains exactly one source data point (scalar or structured). It may contain one or more target data points, which will be generated. This template can be applied on any selection of single source data points. If the type of the source data point matches the one in the template, this auto-generate template can be selected to generate the target data points. This template type can be used to generate special target objects for certain scalar source data points using adaptors. It can also be used to connect structure elements of the source to structure elements of a target using a math block adaptor.

- **Complex auto-generate template**. This template contains more than one source data points. This type must be used, if two or more sources shall generate the targets in a specific way. Since no single source data points can be matched in this case, the source data points which belong together must be grouped under a folder. Math block adaptors can be used with complex auto-generate templates.

Auto-generate templates can use configurable placeholders for data point name, data point description, server object name, server object description. These placeholders are evaluated when the template is applied and new data point instances are created. The available placeholders are listed in Table 15.

Placeholder	Meaning
`%{name}`	In simple auto-generate templates this expands to the source data point name.
`%{descr}`	In simple auto-generate templates this expands to the source data point description.
`%{native_name}`	In simple auto-generate templates this expands to the native name (e.g. register name, NV programmatic name, server object name) of the source data point. If no such native name exists, the data point name is used instead.
`%{native_descr}`	In simple auto-generate templates this expands to the native object description (e.g. server object description) of the source data point. If no such native description exists, the data point description is used instead.
`%{path}`	This placeholder expands to the source data point/folder path. The path extends up until the respective data point folder root folder. Example: The source data point is located in ‘CEA-709 Port.Datapoints.Floor1.Room202’. The path expands to ‘Floor1.Room202’.

Table 15: Placeholders in auto-generate templates.

6.3.4 Global Connections

Global connections provide the same notions as local connections but extend beyond the scope of one device. A global connection establishes a data cloud with a system-wide name. Data points added to a global connection can send data into that connection or receive data from the connection. The data is transferred over an IP-based network. All data is automatically matched by the global connection name. This makes global connections especially useful to provide certain global data in a system, without knowing who will be reading that data. Examples are weather station data, wind alarms or global on/off.

Global connections cannot use adaptors for conversions as in local connections. If conversions are needed, an intermediate register data point must be used to receive/send data from/to the global connection. The adaptor needs to be installed with a local connection between the register and the data point, which requires the conversion.

The system in which the data cloud of a global connection is established is defined by an IP-852 channel. This channel is not related to the CEA-709 technology; it is purely used to define the set of devices exchanging data through global connections. It can, however, coexist with an IP-852 channel for CEA-709. The configuration of the IP-852 channel is done by adding devices to a configuration server.

A global connection has the following properties:

- **Max Send Time**: This timing parameter of the global connection specifies a time in seconds, in which a value update is transmitted into the connection, even if not value has changed. This is typically used for heartbeat functions.
- **Min Send Time**: This timing parameter of the global connection specifies a time in seconds, for which transmissions will be delayed after sending out a value into the connection. This setting can be used to limit the transmission rate to the connection.

The following properties are derived from the data points in a global connection:

- **Receive Timeout**: A data point with a receive timeout will be put into the state offline, if it does not receive a value within the specified period of time (see Section 6.1.2). This also applies to values received from the global connection.
- **Poll on startup**: If a data point in the global connection has the poll on startup feature enabled (see Section 6.1.2), an initial value update will be triggered for the global connection.

How a global connection is created and configured in the Configurator software is described in Section 7.9.7. Note, that the number of configurable global connections on a device is limited (see Section 13.2).

6.4 AST Features

6.4.1 Alarming

The alarming architecture comprises a number of entities. Objects that monitor values of data points and generate alarms depending on an *alarm condition* are called *alarm sources*. The alarms are reported to an *alarm server* on the same device. The alarm server maintains a list of alarm records, called the *alarm summary*. The alarm server is the interface to access the local alarms.

Generic alarm servers provide the maximum set of alarming features and can be accessed over L-WEB (via the Web service) or the Web UI. Data points of all network technologies can be alarmed through generic alarm servers. Technology alarm servers can be used to expose access to the alarms to network technologies that support it. Generic alarm servers can be configured to report their generic alarms to technology alarm servers. For example, a generic alarm server may report its alarms to both CEA-709 and BACnet alarm servers.

An alarm record contains the information about a specific alarm. This includes information about the alarm time, the source of the alarm (i.e., which data point caused the alarm), an alarm message, an alarm value, an alarm type, an alarm priority, and an alarm state. An alarm record undergoes a number of state changes during its life-cycle. When the alarm occurs, it is *active*. At this point the alarm time, alarm message, alarm value is notified using the alarm priority. When the alarm condition subsides, the alarm becomes *inactive*. At this point the clear time and the clear message is notified using the normal priority. The priority levels are configurable on the alarm server, where 0 is the highest and 255 is the lowest priority.

Alarm transitions (to an alarm state, to the normal state) can be acknowledged by an operator. Which of those transitions requires an acknowledgement is configurable on the alarm server. If an active alarm is acknowledged it becomes *active acknowledged*. Active alarms can also become inactive, but an acknowledgement is still required. Then they become *ack-pending*. When an alarm is inactive and was acknowledged it finally disappears from the alarm summary.

An alarm state can be of different alarm types. The alarm type specifies the class of the alarm. The following alarm types exist:

- **Off-Normal Alarm**: This alarm type is a generic alarm class that applies to binary and multi-state alarm conditions. It indicates that the alarmed data point is on an off-normal operating condition that triggered the alarm. An alarm value is supplied. In technology alarm servers, restrictions may apply.
- **High/Low Limit Alarm**: This alarm type is typical for analog alarm conditions. It applies when the alarmed value is over or under the defined alarm limits. An alarm value is supplied. In technology alarm servers, restrictions may apply.
- **Fault Alarm**: This alarm type is indicating that the monitored data point is in a fault state. This is different from off-normal or high/low limit alarms. The value of the data point is within the specifications of the alarm condition but the data point itself is considered faulty. This can stem from an unreliable value or an offline value, i.e., if the data point is offline. No alarm value is supplied.

Alarms may be generated from a given data point value (alarm value or value range) or by comparing a data point command value with a feedback value (feedback alarm). When defining a feedback alarm, the alarmed data point represents the command value and has a ‘feedbackValue’ property relation (see Section 6.1.12). This property relation can be linked to another data point, which effectively provides the feedback value.

Alarmed data points also possess other property relations. The ‘enableAlarm’ property relation can be used to disable or enable alarm conditions when linked to a data point. The property relations ‘highLimit’, ‘lowLimit’, ‘deadband’ can be used to modify analog alarm conditions. The property relations ‘inAlarm’ and ‘ackPend’ are TRUE if a data point is in an alarm state or needs acknowledgement, respectively.

When a data point is alarmed by a generic alarm server, which reports to a technology that requires a dedicated technology data point (e.g., an alarm for a user register is reported to BACnet), the required data point is automatically created and linked via the ‘nativeAlarm’ property relation.

Alarm server objects possess property relations that provide a counter value of active unacknowledged, active acknowledged, and inactive unacknowledged alarms. These property relations may be linked to other data points that can be used to process this information.

Other devices can access the alarm information through a technology alarm server or the Web service. These devices are *alarm clients*. They register with the alarm server and get notified about changes to the alarm summary. Alarm clients can be used to display the current alarm summary and to acknowledge alarm transitions. Depending on the underlying technology, some restrictions may apply to the available alarm information and acknowledgement behavior. Refer to the technology sections for more information.

6.4.2 Historical Alarm Log

The alarm summary of the alarm objects contains a live list of currently active and acknowledge-pending alarms. As soon as an alarm becomes inactive and has been acknowledged, it disappears from the alarm summary. To store a historical log of alarm transitions an *alarm log* is utilized. An alarm log can log transitions of one or more alarm objects.

The alarm log is always local and stored as a file on the device. The size of an alarm log is configurable. The alarm log operates as a ring buffer. As soon as its size limit is reached, the oldest alarm log records are overwritten by newer alarm transitions. The alarm log is available on the Web UI or can be uploaded from the device as a CSV file. The CSV file can also be used as an e-mail attachment.

6.4.3 Scheduling

Schedulers are objects that schedule values of data points on a timely basis. A scheduler object is configured by which data points it shall schedule. This configuration is done by the system engineer once, when the system is designed. The configuration of the times and values that shall be scheduled is not part of that initial configuration and may be changed later. This distinction has to be kept in mind.

A scheduler object sets its data points to predefined values at specified times. The function of the scheduler is state-based. This means, that after a given time, the scheduler maintains this state. It can re-transmit the scheduled values as appropriate (e.g., when rebooting). The predefined values are called *value presets*. A value preset contains one or more values under a single label (e.g., “occupied” schedules the values { 20.0, TRUE, 400 }). Each preset can also be configured with a display color.

Which value preset is scheduled at what time is defined by a *scheduled event*. The event defines the starting time, value preset and end time in a 24-hour period. Events can be one-time events or recurring events. A schedule typically consists of a number of recurring and one-time events, for instance one event for the weekdays Monday through Sunday. See Figure 116 for an example of a schedule in a given calendar week.

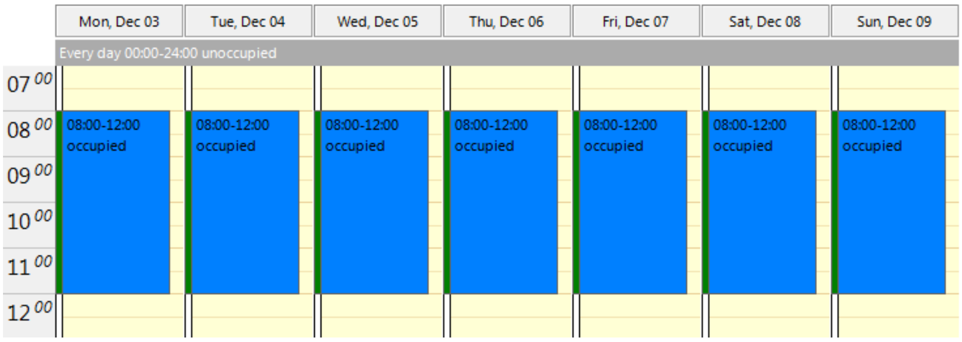


Figure 116: Example of a recurring event in a schedule.

How scheduled events are recurring can be defined by choosing the appropriate event type:

- **One-time**: This event occurs exactly on one defined date.
- **Daily**: This event occurs every day, starting at a given date and ending at a given date.
- **Weekly**: This event occurs every week on the specified weekday.
- **Monthly**: This event occurs every month following a date range or a defined rule (e.g. every last Friday).
- **Yearly**: This event occurs every year following a date range or a specific day every year.
- **Default**: This is a special event. The selected preset value will be in effect 00:00 to 24:00 hours every day if no other event occurs.
- **Calendar**: For some tasks the regular recurrence such as on weekdays is not sufficient. This can be implemented by defining events based on a *calendar*. For instance, there may be a calendar for holidays. The calendar contains a number of *calendar patterns*.

Each calendar pattern describes a pattern of dates on which an event shall occur, e.g., *Holidays*.

One can define a set of scheduled events that are recurring differently. For example one event is defined for regular workdays (Monday through Friday). Another event is defined based on the holidays calendar pattern. This will lead to overlapping events between workday and holiday for those weekdays, which are holidays.

The resolution of this overlap is simple: Each event is configured with a *priority*. Should an overlap occur, the event with the higher priority will be in effect (e.g., Dec 25th in Holidays overrides the regular workday event). An example is shown in Figure 117. The detailed view shows the two overlapping events and the preview shows the effective schedule. Note, if two events with the same priority exist, it is not defined, which one is in effect. Therefore, always use distinct priorities.

Priorities are numbers, but some priorities have been pre-assigned, e.g. highest, override, normal, low. Please also refer to the technology-specific limitations described in Section 7.10 to learn about special behavior of the respective networking technology.

If no event is in effect at a given time, the *schedule default* becomes effective. This can be defined to be any of the defined presets. In Figure 117 the scheduler will write out "unoccupied" after 12:30 as no other event exists. A special schedule default is *silent*. With the silent default the scheduler will be inactive, if no event is in effect. This means it will not update its scheduled data points until the next scheduled event, not even at midnight. Thus, using the silent default one can build an event-based scheduler.

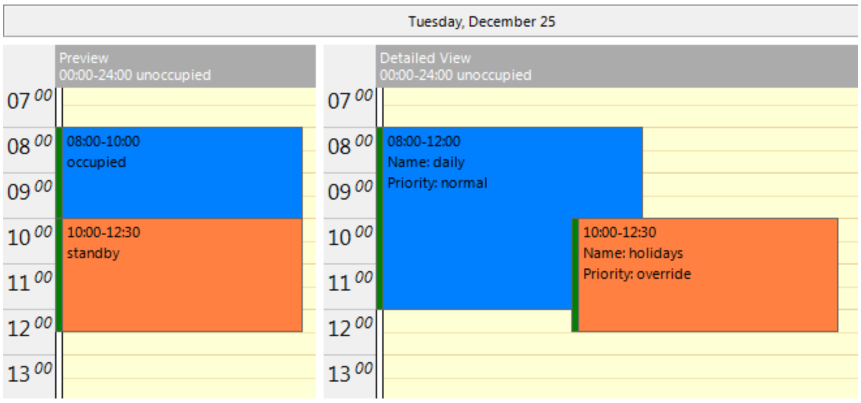


Figure 117: Example with overlapping events and different priorities.

The configuration of calendar-based recurrence is done by calendar patterns in the calendar. Each calendar pattern contains a number of pattern entries. These entries can define the following:

- A single date: This defines a single date. Wildcards may be used in the year to specify Dec 25th of every year.
- A date range: This defines a range. Starting with a start date and ending with the end date. No wildcards should be used.
- A Week-and-Day definition: This defines dates based on a week, such as every 1st Friday in a month, every Monday, every last Wednesday of a month.

A schedule defines at which time instants certain states of the scheduled data points are maintained. The *next-state* feature allows looking up to 48 hours ahead into the future and predicts when the next scheduled state change will occur. There are two data points involved: the timeToNext is a counter in minutes to the next scheduled event, and the nextState data point is the state of the next scheduled event. This information can be used by controllers for optimum start algorithms (e.g., pre-heat a room for the scheduled occupancy state). Use the SNVT_tod_event in CEA-709 to accomplish this task. With generic schedulers and BACnet schedulers use the scheduler's property relations timeToNext and nextState (see Figure 118).

Datapoint Name	No.	Direction	OPC	Use	ID
calendar	1	Value	☑	1	1039
◢ Schedule_reg1	2	Value	☑	0	1060
enable -> User Registers.sched_enable	2.1	Value	☑	0	106F
enableFb	2.2	In	☑	0	1070
nextPresetName	2.3	In	☑	0	1074
nextState	2.4	In	☑	0	1073
presetName	2.5	In	☑	0	1071
timeToNext	2.6	In	☑	0	1072
User Registers.reg1	2.7				105F

Figure 118: Property relations of a scheduler object.

When a scheduler is executing the schedule on the local device, it is called a *local scheduler*. Such a scheduler is configured to schedule data points and later its daily schedules can be modified. When accessing the daily schedules of a scheduler, which executes on a remote device, the object is called a *remote scheduler*. A remote scheduler has the same interface to the user to modify daily schedules. A remote scheduler object can be used as a user-interface for schedulers that execute on different devices.

6.4.4 Trending

Trending refers to the ability to log historical values of data points over time. A trend log object is responsible for this task. The generic trend log object provides the maximum set of features and can be accessed by L-WEB and the Web service. It can be configured to record historical data of any data point on the device. Log records are generated either in fixed time intervals, on change-of-value (COV) conditions, or when a trigger is activated. The fixed intervals can be optionally aligned to the wall time (e.g., to the top of the hour). After a reboot the recording is resumed at the aligned intervals. Trend log objects can trend either local or remote data points. Technology trend log objects can be used to record historical values of the respective technology data points and expose them to network technologies that support it. These historic logs are separate from the generic trend logs and certain restrictions of the technology may apply.

The trend data is stored in a binary format on the device. The capacity of a given trend log is configured. The trend log can be operated in one of two modes: In *linear mode* the trend file fills up until it reaches its capacity. It then stops logging. In *ring buffer* mode the oldest log records are overwritten when the capacity is reached.

Devices with SD cards also allow backups of the trend logs on external Flash storage. This backup can be triggered by the user over the LCD display or be triggered by certain actions. The trend data is stored in CSV format under a folder identifying the device by serial number and the trends sub-directory, e.g. '016101-8000000DEA51/trends'. The SD card can be used on different devices. In this case different device directories will be created. The trend backup files can be opened directly on a PC. The backup on external storage can be enabled individually per trend log.

A fill-level action can be activated, whenever the trend log has logged a percentage of its log size with new log records. A fill-level condition of 70% on a trend log with 1000 items capacity will activate the fill-level trigger every 700 logged records. This trigger can be used to send E-Mails or backup trend data on external storage if available.

Trended data points can be logged as their actual values at given time instants or as an aggregated value over the defined log interval. Aggregation can be calculated as minimum, maximum, or average. Aggregation can be beneficial, if the trended value changes more frequently than the selected log interval. Using aggregation, the log interval can be chosen to limit the amount of logged data while preserving information of the trended value.

For technology trend log objects, certain restrictions apply as to how many data points can be trended in one trend log and which trend modes are available. Refer to the technology sections for more information.

6.4.5 E-mail

The e-mail function can be combined with the other AST features. The format of an e-mail is defined through *e-mail templates*. An e-mail template defines the recipients, the e-mail text, value parameters inserted into the text and triggers, which invoke the transmission of an e-mail. An e-mail template can also specify one or more files to be sent along as an attachment.

The e-mail text content can contain text and configurable placeholders. The placeholders expand to their content when the e-mail is transmitted. Placeholders can also be used in other text fields, such the Subject field. The placeholders available for e-mail templates are listed in Table 16.

Placeholder	Meaning
%{v*n*}	This placeholder expands to the content of a data point variable defined in the e-mail template. The n refers to the n-th data point variable in the list. The data point variable list specifies this index.
%{v*n*.src_name}	If the data point variable is an alarm, this placeholder expands to the source name of the alarmed data point, for which a new alarm is reported.
%{v*n*. al_descr}	If the data point variable is an alarm, this placeholder expands to the message of the reported alarm. For a to-alarm transition it contains the alarm message, for a to-normal transition it contains the clear message, for a to-fault transition it contains the fault message.
%{v*n*. al_type}	If the data point variable is an alarm, this placeholder expands to the alarm type of the reported alarm.
%{v*n*. al_state}	If the data point variable is an alarm, this placeholder expands to the state of the reported alarm.
%{v*n*. al_tm}	If the data point variable is an alarm, this placeholder expands to the alarm time of the reported alarm.
%{v*n*. cl_tm}	If the data point variable is an alarm, this placeholder expands to the clear time of the reported alarm.
%{v*n*. ack_tm}	If the data point variable is an alarm, this placeholder expands to the acknowledge time of the reported alarm.
%{v*n*. ack_src}	If the data point variable is an alarm, this placeholder expands to the acknowledge source text of the reported alarm. If the alarm has not been acknowledge, this is empty.
%{v*n*. al_val}	If the data point variable is an alarm, this placeholder expands to the value which triggered the alarm (alarm value).
%{mailid}	This placeholder expands to the mail ID used for the transmitted message. This mail ID is different for each message.
%{timestamp}	This placeholder expands to the mail timestamp seen in the transmitted message.

Table 16: Placeholders in e-mail templates.

A prerequisite to sending e-mails is the configuration of an e-mail account on the device. This can be done on the Web UI (see Section 5.2.10). It is recommended to use the e-mail server of your Internet provider. For public mailers, enable the required authentication. SSL/TLS e-mail authentication is supported for using Hotmail, gmail or Yahoo!.

The amount of generated e-mails can be limited using a rate limit algorithm. The transmission of e-mails can be disabled altogether by using a special data point. That data point can be scheduled or driven over the network.

If an e-mail cannot be sent (e.g. the mail server is not reachable), the mail delivery is retried up to 24 times every 30 minutes.

6.4.6 Historic Filters

For certain applications historic values of a given base data point, both recent and far into the past, can be of interest. This can be accomplished with *historic filters*. Historic filters allow processing historic values of the base data point according to a filter function. One or more such functions can be defined per base data point. The result of the historic filter is written to *historicFilter* property relations. For each historic filter function a time period can be defined at which the base value is sampled, e.g., every first of the month at midnight, and how many samples ago. Historic filters can be created for any analog, binary, or multi-state data point. It is not necessary to create a trend log.

The following sampling periods can be defined:

- Value every *x* minutes aligned to full hour (*x* = 1, 2, 5, 10, 15, 20, 30 min), 0 or 1 samples ago,
- Hourly value at full hour, 0..24 samples ago,
- Daily value at HH:MM:SS of the day, 0..60 samples ago,
- Weekly value at HH:MM:SS on weekday (Mon..Sun), 0..10 samples ago,
- Monthly value at HH:MM:SS on day of month (1..31, last), 0..24 samples ago,
- Yearly value at HH:MM:SS on DD/MM of the year, 0..5 samples ago.

By using historic filter data points it is possible to implement numerous calculations on historic values of the base data point. For example it is possible to create two filter data points with a daily sampling period recording the energy consumption at midnight, one holding the most current sample (today at midnight) and the other the previous sample (yesterday day at midnight). This is shown in Figure 119.

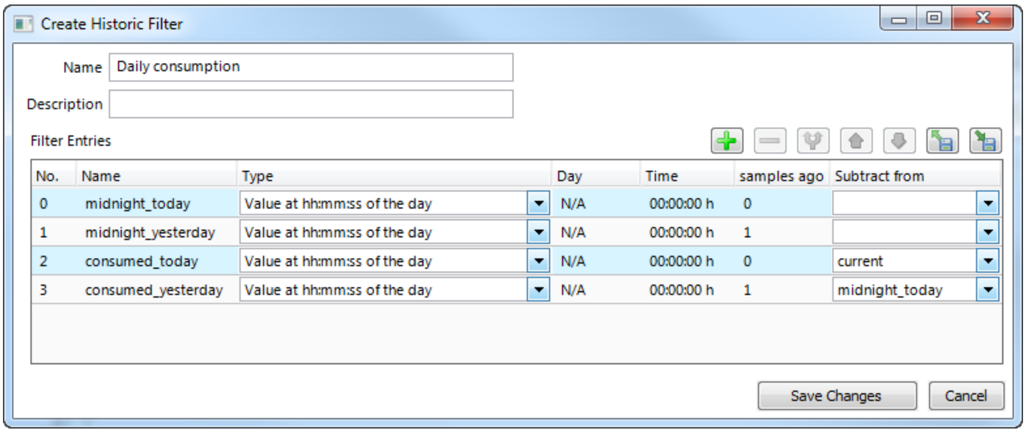


Figure 119: Example historic filters for daily consumption.

For calculating the difference between the current value and any historic value, the filter definition can be configured in a delta mode. This is a shortcut to creating a math object subtracting the historic filter data point value from the current value of the underlying data point or the value of another filter item. The example shows two results: Filter item ‘2’ yields the consumption to-the-hour of the current day (subtract value at midnight from current value). Filter item ‘3’ yields yesterday’s consumption (subtract the value of midnight yesterday from midnight today). The resulting values are available in data points, which can be visualized or trended.

The historic filters definitions are managed by historic filter resources. These are templates and stored in the project resources. They can be applied to data points. When editing an historic filter template, all existing historic filter relations are updated accordingly. For more information on how to configure historic filters in the Configurator please refer to Section 7.15.

6.5 CEA-709 Technology

6.5.1 Limitations for Local CEA-709 Schedulers

CEA-709 schedulers and the CEA-709 calendar adhere to the LONMARK standard objects. For CEA-709, certain restrictions exist that need to be kept in mind. Attached data points can either represent an entire NV or individual elements of a structured NV. CEA-709 schedulers may have several different groups of data points attached, i.e., the value preset may consist of more than one element. For example, a CEA-709 scheduler might schedule a SNVT_temp and a SNVT_switch and have 3 elements in each value preset as depicted in Figure 120.

Datapoint	Description	Location	Group	Default	day	night
nvo_setpoint		LINX-110.CEA709 Port.Datapoints	-	0.00	21.00	16.00
nvo_switch.value		LINX-110.CEA709 Port.Datapoints	-	0.00	0.00	50.00
nvo_switch.state		LINX-110.CEA709 Port.Datapoints	-	0.00	0.00	1.00

Figure 120: Example value presets in CEA-709 schedulers.

Priorities of exception days in a CEA-709 scheduler range from 0 (the highest) to 126 (the lowest). The value 127 is reserved as a default for weekdays.

Further, the implementation as LONMARK standard objects requires the use of configuration properties. If the number of CEA-709 schedulers or their capacities for daily schedules and value presets is changed, the resource and static interface of the CEA-709 port changes. The resources reserved for LONMARK calendar and scheduler objects can be changed in the project settings (see Section 7.16.4). When downloading a project, the software verifies if sufficient resources have been configured.

6.5.2 Limitations for CEA-709 Alarm Servers

Local CEA-709 alarming supports only one alarm server object. This alarm server object is represented by the device's LONMARK node object and facilitates the SNVT_alarm2 output network variable. Acknowledging alarms in the alarm server is adhering to the LONMARK specification and relies on the RQ_CLEAR_ALARM mechanism.

Note: *On the L-DALI the alarm server object is pre-allocated for each interface. It is called "DALI channel x Alarms" (x = 1-4).*

6.5.3 Limitations for Local CEA-709 Trends

Local CEA-709 trend objects support trending multiple data points in all trend modes, interval, COV, and trigger, including aligned intervals. The enable data point is also supported. All data points can be NVs, registers or of any other technology. There is no LONMARK object linked to the trend object. Consequently, trend data cannot be accessed over a LONMARK mechanism.

6.6 BACnet Technology

6.6.1 BACnet Data Points

Data points in the BACnet technology are known as BACnet objects. They have a specific type (e.g. analog input or binary output) and a set of properties, which describe the data point more closely. The actual value is stored in the "Present_Value".

On the device, there exist two classes of BACnet data points:

- **BACnet server objects** (SO): These BACnet objects configured by the Configurator software to be allocated *locally* on the device. These objects can be accessed by the BACnet building control system or operating workstations. They support COV subscriptions to deliver value changes in an event-driven way. Local server objects can be created as AI, AO, AV, BI, BO, BV, MI, MO, MV, Accumulator, Pulse_Converter, Loop objects.

- **BACnet client mappings** (CM): For certain applications, it is necessary that the device acts as a BACnet client. This functionality can be configured by activating a *client mapping*. Client mappings can be of the type *Poll*, *COV*, *Write*, *Auto* or *Value*. This specifies how the BACnet client accesses other BACnet objects on the BACnet network. The *Auto* method determines the best way (poll, COV, or write) to talk with other server objects. *Poll* is used for objects that need to read data from other BACnet objects in a periodic manner. *COV* is used to subscribe for COV at other BACnet objects in order to get updates in an event-driven fashion. *Write* is used to send updates to other BACnet objects. *Value* refers to a combined read and write client mapping. When writing a value to this client mapping, the value is written to the remote BACnet object. As soon as the Present_Value of the remote BACnet object is updated, the value is transferred back.

The direction of BACnet server objects deserves a closer look. The direction specified for data points in the Configurator software always refers to the network view of the communication. The definition of input and output objects in BACnet, however, refers to the process view, which is opposite to the network. Therefore, a BACnet analog input (AI) object is modeled as an analog output data point. The direction of client mappings naturally refers to the network communication. Therefore, a write client mapping is represented as an analog output data point.

In BACnet commandable objects can be written with values at a certain priority. The value with the highest priority is in effect. When revoking a written value, the NULL value is written. This takes back the value. When all written values are withdrawn, the Relinquish_Default value is in effect.

For BACnet server objects the write priority defines which priority is written to the commandable server object. It is possible to create additional *priority write* data points, which can be configured to write at other priority levels. This may be necessary, if two parts of an application are required to write with two different priorities. To know, which priority slots are used in a commandable object, additional *priority read* data points can be added. They reflect the value on a given priority slot.

The default value feature of a data point is mapped to the Relinquish_Default property for commandable objects. For BACnet objects, which are not commandable, the Present_Value is initialized with the specified default value.

6.6.2 BACnet Alarming

BACnet alarming on the device is based on the *intrinsic reporting* mechanism. Currently, algorithmic reporting is not supported. Alarm conditions can only be applied to data points, which map to BACnet server objects. If defined, the intrinsic reporting properties of the underlying BACnet objects are enabled. Alarm conditions can be specified for analog input, output, value objects (AI, AO, AV), for binary input, output, value objects (BI, BO, BV), and for multi-state input, output, value objects (MSI, MSO, MSV). With BACnet intrinsic reporting alarm conditions on binary output (BO) and multi-state output (MSO) can only be feedback alarms.. These restrictions do not apply, if the alarm condition reports to a generic alarm server.

Alarm servers in the BACnet technology are mapped to BACnet Notification Class (NC) objects. Each alarm server is mapped to one NC. The notification class number can be configured in the object instance number property of the alarm server object.

Remote alarms in the BACnet technology refer to a remote NC object. When the device starts up, the remote alarm object reads out the current alarm state of the remote NC and reporting objects. To get notified about alarm transitions during run-time, the device registers in the Recipient_List of the remote NC object.

Some BACnet devices do not send a usable text in their alarm notification messages. For those devices the alarm client provides the option **Ignore alarm message text**. If this option is enabled, the alarm client ignores the message text of an alarm notification and reads the description property of the alarmed object instead.

6.6.3 BACnet Schedulers and Calendars

BACnet schedulers and the BACnet calendar adhere to the standard schedule and calendar object in BACnet. For each scheduler a BACnet Schedule object is created. The calendar deserves a closer look. For each calendar pattern a BACnet Calendar object is created. The visible calendar on the Web UI is therefore a collection of BACnet calendar objects. Each calendar pattern therefore is associated with a BACnet object instance number. The calendar pattern “Holidays” is for example visible as CAL,1 on the BACnet port.

The BACnet schedule object allows only objects of one selected data type to be scheduled. Therefore, schedulers on BACnet can only schedule one class of data points (e.g., only one group of analog data points). As a consequence, the value preset in BACnet always has only one element. The name of the value preset is not stored in BACnet. It is not accessible over the BACnet network, either. Therefore, a default name is created, such as ‘22 °C’ for an analog value. An example of two scheduled BACnet objects is shown in Figure 121. With the extended BACnet features enabled in the project settings, a preset label can be assigned to a specific scheduled value. For example the value ’16 °C’ can be assigned ‘night’. Click in the column header and type the desired text.

Datapoint	Description	Location	Group	Default	22 °C	night
bac_temp1	temp	BACnet Port.Datapoints	1	0.00	22.00	16.00
bac_temp2	temp	BACnet Port.Datapoints	1	0.00	22.00	16.00

Figure 121: Example value presets in BACnet schedulers.

Priorities of exception days in a BACnet scheduler range from 1 (the highest) to 16 (the lowest). Weekdays in BACnet have no priority.

Changing the number of calendar patterns in a BACnet calendar can only be done through the configuration software and not during run-time. The individual calendar pattern entries in the calendar patterns can be changed at run-time. Therefore, it is advisable to reserve a suitable number of calendar patterns in a BACnet calendar and leave them empty if not needed immediately.

6.6.4 BACnet Trend Logs

Trending in the BACnet technology is based on the BACnet TrendLog object. A number of restrictions apply to trend log objects in BACnet. Trend log objects must be created by the

Configurator software. These objects are accessible over the BACnet network for other BACnet devices and operator workstations (OWS). All configuration properties can be modified by the Configurator software as well as an OWS. The number of trend log objects cannot be changed at run-time. Therefore, if it is intended that an OWS configures the trend logs, a suitable number of empty trend log objects (i.e., without attached data points) must be created in the Configurator software.

In BACnet trend logs, only one data point can be trended per trend log object. The trended data point can be either a local BACnet server object or a remote BACnet object accessed through a client mapping, showing the referenced property for trending to the OWS. Data points of other technologies and the min/max/avg algorithms can be trended as generic data points without having a BACnet property reference.

BACnet trend logs support interval, COV and trigger-mode logs, aligned intervals are available in interval mode. The setting linear and ring-buffer logging is mapped to the Stop_When_Full property of the underlying BACnet trend log object. This setting in the Configurator software is a default and can be overridden by writing to the Stop_When_Full property by the OWS. The trend log object adheres to BACnet revision 12.

If an enable data point is configured by the Configurator software, the Log_Enable property is written with the value of that data point. If no enable data point is configured, the Log_Enable is TRUE as a default and can be modified over the network.

The fill-level action is mapped to generating a buffer event notification in the BACnet trend log object. The fill-level trigger can still be used for e-mails even if no notification class is configured in the BACnet trend log object. The fill-level percentage maps to the Notification_Threshold property. The percentage setting in the Configurator software is a default and can be changed by the OWS over the network.

The BACnet technology also supports *remote trend logs*. A remote trend log is basically a BACnet trend log client, which accesses trend data on another device. The remote trend can load the trend data from the remote device and supply it to L-WEB or the trend CSV files.

6.6.5 Dynamic Polling in BACnet

Reading client mappings in BACnet rely either on COV or on polling. Static polling can be configured as a fallback, if COV is not supported, by setting the pollcycle (see Section 6.1.2). Data points which are not used by other objects do not subscribe via COV or perform polling. In addition, this technology also supports dynamic polling. If the data point Web UI or L-WEB requires a refresh on those client mappings, COV subscriptions are made or polling is activated at the configured pollcycle. If those data points go out of scope, the polling on the remote BACnet object stops and COV is unsubscribed.

If no static polling is needed at all, the pollcycle setting can be left at zero in the client mapping. In this case, only a COV subscription is made, if the device supports COV. If COV is not supported, polling is only started, as soon as the values are required. This is especially important on MS/TP channels with devices, that do not support COV.

6.6.6 BACnet Data Points in Connections

BACnet data points can be used in local and global connections. In BACnet commandable objects can be written with values at a certain priority. The value with the highest priority is in effect. When revoking a written value, the NULL value is written. This takes back the value. When all written values are withdrawn, the Relinquish_Default value is in effect. In other technologies there is no notion of taking a value back. To model this behavior, a distinctive *invalid* value can be written to such a data point. For those data points that do not an intrinsic invalid value, it can be specified when editing the data point. To make a BACnet object convey that invalid value to the opposite side, enable the property **Relinquish to Invalid**.

BACnet data points can be auto-generated from other data point sources (see Section 6.3.3). Only BACnet server objects can be generated and the connected value is reflected in the Present_Value property. Which type of BACnet object is created depends on the type of the source data point or of the source structure member. For analog sources, analog objects are created. The best-matching BACnet engineering unit is chosen. Other properties of analog objects are copied from the source data point, including min and max present value. Multi-state objects are created for source enumeration types. Which state IDs exist is documented in the BACnet multi-state texts array. This information is copied from the source and made compatible with BACnet restrictions by renumbering state IDs.

Normally BACnet AI, BI, MI are created out of input source data points and AO, BO, MO out of output source data points. The BACnet project settings allow changing this default to BACnet value objects AV, BV, MI. In the auto-generate preview the user can review and change those object types individually before generating the data points.

6.7 Regular Expressions

Some features of the Configurator use regular expressions to perform complex operations on text. These apply to data point filters, naming rules and folder copy and rename. A regular expression is a pattern that describes a set of strings. It is applied to an input text and performs pattern-matching by evaluating the expression string consisting of literal characters to match and meta-characters. Literals match themselves (**`abc`** matches exactly 'abc') whereas meta-characters may match one or more characters of the input text. The available metacharacters are listed in Table 17. Regular expressions can handle abbreviations, such as **`\d`** instead of **`[0-9]`**. The abbreviations provided are listed in Table 18.

Meta-Character	Meaning
`.`	Matches any single character.
[ ]	Indicates a character class. Matches any character inside the brackets (for example, `[abc]` matches ‘a’, ‘b’, and ‘c’).
`^`	If this meta-character occurs at the start of a character class, it negates the character class. A negated character class matches any character except those inside the brackets (for example, `[^abc]` matches all characters except ‘a’, ‘b’, and ‘c’). If `^` is at the beginning of the regular expression, it matches the beginning of the input (for example, `^[abc]` will only match input that begins with ‘a’, ‘b’, or ‘c’).
`-`	In a character class, `-` indicates a range of characters (for example, `[0-9]` matches any of the digits ‘0’ through ‘9’).
`?`	Indicates that the preceding expression is optional: it matches once or not at all (for example, `[0-9][0-9]?` matches ‘2’ and ‘12’).
`+`	Indicates that the preceding expression matches one or more times (for example, `[0-9]+` matches ‘1’, ‘13’, ‘666’, and so on).
`*`	Indicates that the preceding expression matches zero or more times.
`??`, `+?`, `*?`	Non-greedy versions of `?`, `+`, and `*`. These match as little as possible, unlike the greedy versions which match as much as possible. Example: given the input ‘<abc><def>’, `<.*?>` matches ‘<abc>’ while `<.*>` matches ‘<abc><def>’.
`( )`	Grouping operator. Example: `([0-9]+,)*[0-9]+` matches a list of numbers separated by commas (such as ‘1’ or ‘1,23,456’).
`{ }`	Indicates a match group. The actual text in the input that matches the expression inside the braces can be retrieved through the sequence `\0`, `\1`, etc.
`\`	Escape character: interpret the next character literally (for example, `[0-9]+` matches one or more digits, but `[0-9]\+` matches a digit followed by a plus character). Also used for abbreviations (such as `\a` for any alphanumeric character; see Table 18 below). If `\` is followed by a number *n*, it matches the *n*-th match group (starting from 0). Example: `<{.*?}>.*?</\0>` matches ‘<head>Contents</head>’.
`$`	At the end of a regular expression, this character matches the end of the input. Example: `[0-9]$` matches a digit at the end of the input.
`|`	Alternation operator: separates two expressions, exactly one of which matches (for example, `T|the` matches ‘The’ or ‘the’).
`!`	Negation operator: the expression following `!` does not match the input. Example: `a!b` matches ‘a’ not followed by ‘b’.

Table 17: Metacharacters in Regular Expressions.

Abbreviation	Matches
`\a`	Any alphanumeric character: `([a-zA-Z0-9])`
`\b`	White space (blank): `([ \\t])`
`\c`	Any alphabetic character: `([a-zA-Z])`
`\d`	Any decimal digit: `([0-9])`
`\h`	Any hexadecimal digit: `([0-9a-fA-F])`
`\n`	Newline: `(\r|(\r?\n))`
`\q`	A quoted string: `(\"[^\"]*\")|(\'[^\']*\')`
`\w`	A simple word: `([a-zA-Z]+)`
`\z`	An integer: `([0-9]+)`

Table 18: Abbreviations for regular expressions

When performing a replace operation on an input text, match groups are used in the output template. Match groups are delimited by curly braces containing a matching pattern. As an example the regular expression `{[0-9]?[0-9]}:{[0-9][0-9]}` contains two match groups. The first match group matches any combination of one or two digits. The second matches any two-digit combination. To make the replacement effective, the entire regular expression has to match. In the example the regular expression matches any one or two-digit combination followed by a ':' and by any two-digit combination.

Then the output is assembled by specifying an output template. In that output template, both literal text and references to the match groups can be specified. The first match group is denoted by **`\0`**, the second by **`\1`** and so on. Using the output template **`ref\0-\1`** on the example expression, the following input texts will produce these results:

- "ab1:22c" matches, replacement is "ref1-22",
- "foo22:11bar" matches, replacement is "ref22-11",
- "ab22:1c" does not match, no replacement results in "ab22:1c".

7 The LINX Configurator

This Chapter gives step-by-step instructions on how to configure and commission the L-DALI using the LINX Configurator PC software. This includes setup and commissioning of the devices and groups on the connected DALI channels, configuring the parameters of the L-DALI's light and sunblind applications, and setting up advanced alarming, scheduling and trending functionality. For LDALI-10X models (CEA-709) we show the configuration steps using LonMaker TE but other LNS-based network management tools can be used as well to install and configure the L-DALI. We also show how to configure the L-DALI without LNS.

7.1 Installation

7.1.1 Software Installation

The LINX Configurator can be used to commission the DALI network, configure the light application parameters, and to setup the data point configuration of the L-DALI. The Configurator is installed as a plug-in tool for all LNS-based network management tools as well as a stand-alone tool (for LDALI-20X models or for systems without LNS).

System requirements:

- LNS 3.1 SP8 U1, LNS 3.2 TE SP5, OpenLNS (for LNS mode),
- Windows Vista, Windows 7, Windows 8 (64 bit) or Windows Server 2003 (32 bit), Windows Server 2008, Windows Server 2012,
- Internet Explorer 9 or higher.

The LINX Configurator can be downloaded from the LOYTEC Web site http://www.loytec.com. When asked for the type of installation, there are two options to choose from. Select **Typical** to install the required program files. Select **Full** to install the LONMARK resource files along with the software. This option is useful, when the system does not have the newest resource files.

7.1.2 Registration as a Plug-In

If the L-DALI shall be configured using LNS-based tools (e.g., NL200 or LonMaker), the LINX Configurator needs to be registered as an LNS plug-in. In the following, the process is described for LonMaker TE. Otherwise, please refer to the documentation of your network management tool on how to register an LNS plug-in.

Note: *If you are using a LNS-based tool using an LNS version prior to LNS TE, registration of the LINX Configurator as a Plug-In may take more than 10 minutes.*

To Register in LonMaker TE

1. Open LonMaker and create a new network.

2. Click **Next** until the plug-in registration tab appears in the Network Wizard. Select the **LOYTEC LINX Configurator (Version X.Y)** from the list of **Not Registered** (see Figure 122).

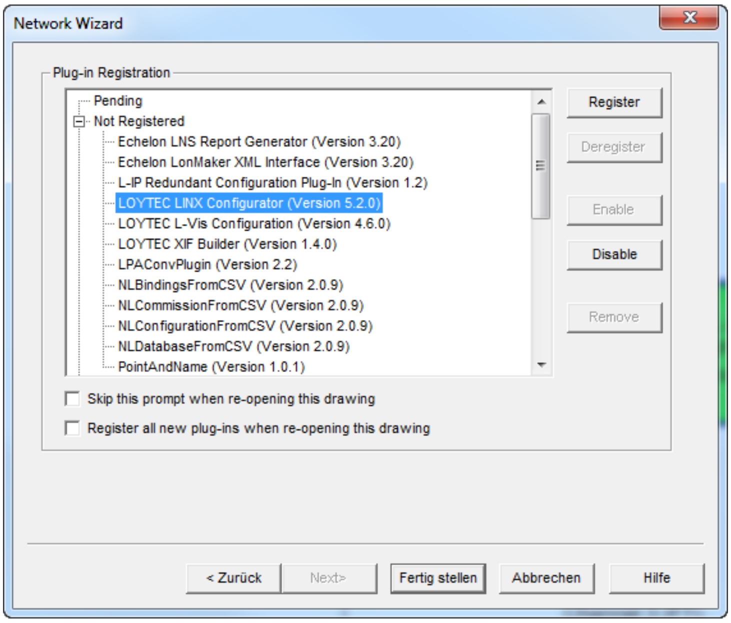


Figure 122: Select the Plug-in to be registered.

3. Click **Register**. The Configurator now appears in the **Pending** list.

4. Click **Finish** to complete the registration.

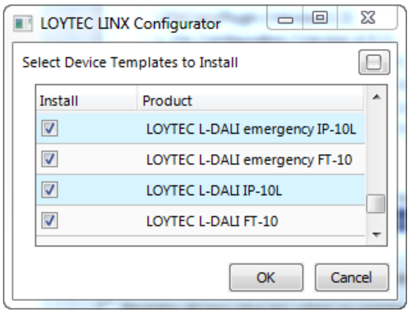


Figure 123: Select device templates for installation.

5. A dialog appears to optionally select the device templates to be installed. Deselect unneeded device templates to speed up registration. Click **OK** to continue.

6. The selected device templates are added automatically and XIF files are copied into the LNS import directory.

Note: *If you are using multiple databases (projects) make sure you have registered the plug-in in each project.*

7. Under **LonMaker → Network Properties → Plug-In Registration** make sure that the **LOYTEC LINX Configurator (Version X.Y)** shows up under **Already Registered**.

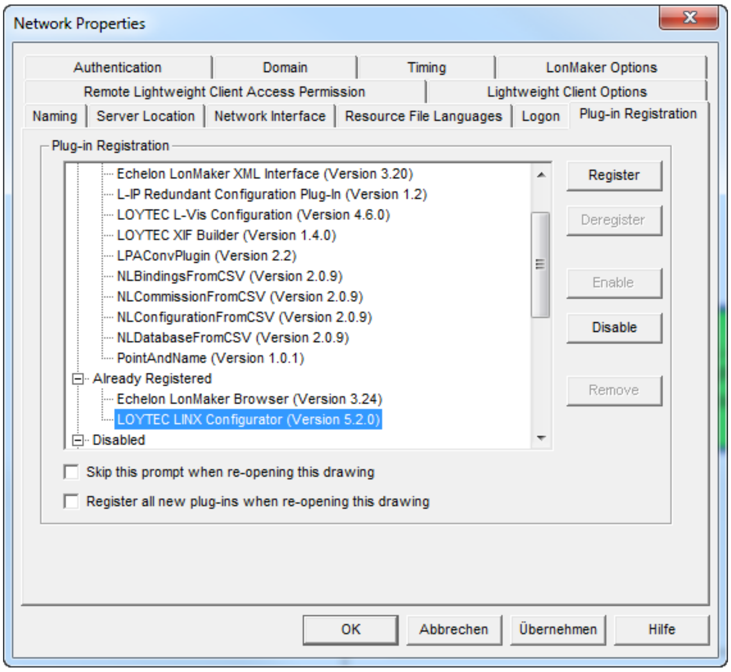


Figure 124: Check that the LINX Configurator is properly registered.

7.1.3 Operating Modes

The Configurator can be used in on-line, off-line, and stand-alone mode. On-line and off-line mode refers to the 2 operating modes of your LNS network management software.

- **On-line Mode**: This is the preferred method to use the configuration utility. The network management tool is attached to the network and all network changes are directly propagated into the network. This mode must be used to add the device, commission the device, scan the connected DALI channels, and download the configuration into the device.
- **Off-line Mode**: In off-line mode, the network management software is not attached to the network or the device is not attached to the network, respectively. This mode can be used to add the device using the device templates, setup DALI devices and groups, configure the parameters of the L-DALIs light and sunblind applications, and setup advanced alarming, scheduling and trending functionality.
- **Stand-alone Mode**: The Configurator can also be executed as a stand-alone program. This mode is useful for the engineer who doesn’t want to start the configuration software as a plug-in from within network management software (e.g., NL-220, LonMaker or Alex). Instead the engineer can work directly with the device when online or engineer it offline.

7.2 Workflows for the L-DALI

This section discusses a number of work flows for configuring the L-DALI in different use cases in addition to the simple use case in the quick-start scenario (see Chapter 3). The description is intended to be high-level and is depicted in flow diagrams. The individual steps refer to later sections, which describe each step in more detail. In principle, the LINX Configurator supports the following use cases:

- On-Line (see Section 7.2.2)
- Off-Line (see Section 7.2.3)

7.2.1 Involved Configuration Files

In the configuration process, there are a number of files involved:

- XIF file (LDALI-10X only): This is the standard file format to exchange the static CEA-709 interface of a device. This file can be used to create a device in the database without having the L-DALI on-line. There exists a standard XIF file for the FT port (“L-DALI_FT-10.xif”) and one for the IP-852 port (“L-DALI_IP-10L.xif”). In addition, there are XIF files for an L-DALI with enabled Emergency interface (“L-DALI_emergency_FT-10.xif” and “L-DALI_emergency_IP-10L.xif”, see Section 8.2) and for use with legacy mode (“L-DALI_legacy_FT-10.xif” and “L-DALI_legacy_IP-10L.xif”, see Section 7.3.4). For custom CEA-709 interfaces a XIF file can be generated (see Section 7.7.12).
- LINX Configurator project file: This file contains the DALI configuration for all ports, light and sunblind application parameters and the AST configuration. These files end with “.ldali”. It stores all relevant configuration data and is intended to be saved on a PC to backup the L-DALI’s configuration.

7.2.2 On-Line

The flow diagram in Figure 125 shows the steps that need to be followed in order to configure the L-DALI when the device and the DALI network including all DALI devices (e.g. ballasts, sensors, etc.) are available on-line.

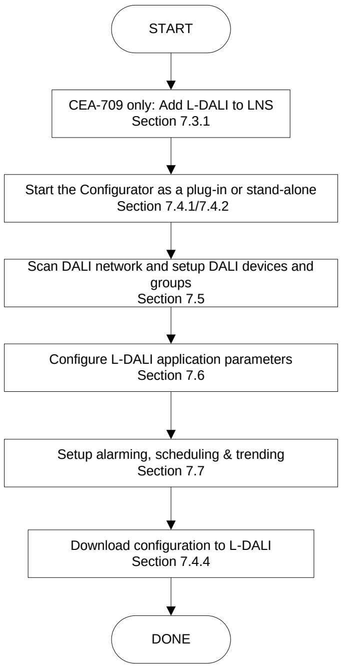


Figure 125: Basic on-line design-flow.

In case of LDALI-10x models in the first step the L-DALI device must be added to LNS (see Section 7.3.1). Then the LINX Configurator must be started to configure the L-DALI. Use Plug-In mode (see Section 7.4.1) is working with an LNS based tool (CEA-709 only) or stand-alone mode (see Section 7.4.2) otherwise. In the Configurator, the DALI network is scanned for DALI devices and the devices are setup and assigned to DALI groups (see Section 7.4.6). Then the parameters for the light application and the sunblind application can be configured (see Section 7.6). Optionally, alarming, scheduling and trending functionality can be set up (see Section 7.7). Finally, the configuration needs to be downloaded to the L-DALI (see Section 7.4.4). It is recommended to save the complete configuration to a file for being able to replace an L-DALI in the network. Additionally a backup should be created (see Section 5.2.17).

To add more DALI devices, change DALI group assignment or application parameters simply repeat the steps described above.

7.2.3 Off-Line

The flow diagram in Figure 126 shows the steps that need to be followed in order to configure the L-DALI off-line. In this scenario the first steps can be performed without the L-DALI and the DALI network being physically available. This allows to prepare the on-line commissioning and thus to speed up the time required for on-site installation. Further, some steps of the on-line commissioning part can be performed by less skilled personnel using the L-DALI Web Interface.

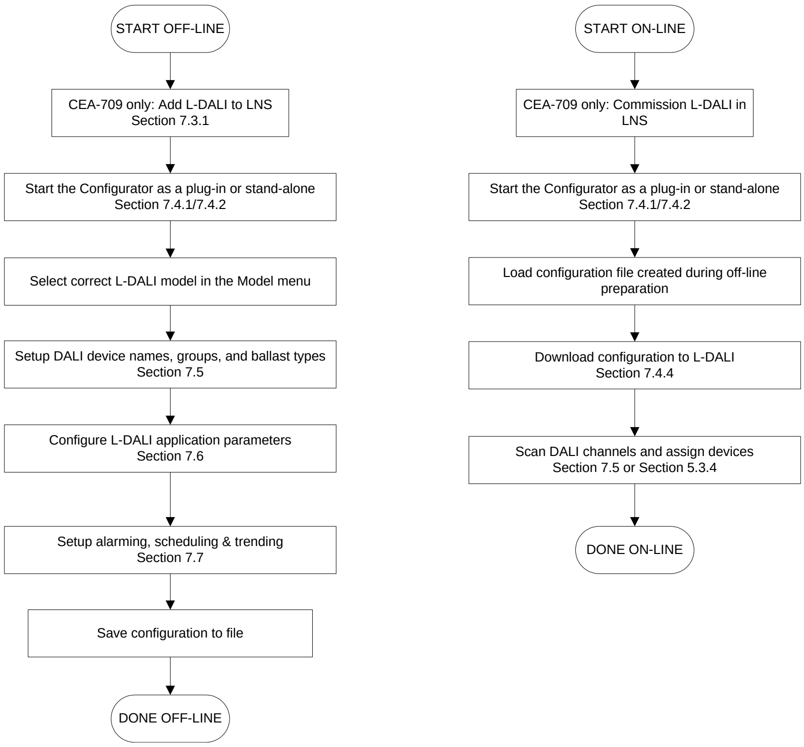


Figure 126: Basic off-line design-flow.

In case of LDALI-10x models in the first step the L-DALI device must be added to LNS (see Section 7.3.1). Then the LINX Configurator must be started to configure the L-DALI. Use Plug-In mode (see Section 7.4.1) is working with an LNS based tool (CEA-709 only) or stand-alone mode (see Section 7.4.2) otherwise. Note, that the device is off-line. Select the correct L-DALI model in the **Model** menu. Next setup names, groups and optionally device type for the DALI devices (see Section 7.4.6). The names assigned must allow identifying physical devices later on (e.g. “Room 301-1” for first ballast in room 301). The device type (e.g. emergency light) must be selected if any device type specific parameters shall be configured in the next step. Then configure the parameters for the light application and the sunblind application (see Section 7.6). Optionally, alarming, scheduling and trending functionality can be set up (see Section 7.7). Save the created configuration to a file.

When the L-DALI is physically available, in case of a LDALI-10X model, first commission all ports of the device in LNS. Then again start the Configurator in plug-in mode (see Section 7.4.1). Load the file created during the off-line preparation and download the configuration to the L-DALI (see Section 7.4.4). Now, scan the DALI channels, either using the Configurator (see Section 7.4.6) or the Web Interface (see Section 5.3.2) and assign the DALI devices found to the names entered during off-line preparation. Finally, it is recommended to upload and save the complete configuration to a file for being able to replace an L-DALI in the network. Additionally a backup should be created (see Section 5.2.17).

To add more DALI devices, change DALI group assignment or application parameters it is recommended to use the on-line work flow (see Section 7.2.2).

7.2.4 Replace an L-DALI

An L-DALI can be replaced in the network by another unit. This might be necessary if a hardware defect occurs. If a backup of the latest configuration is available, restoring the backup is the best option (see Section 7.4.6 and Section 5.2.17).

The work flow to restore a device from the corresponding LINX Configurator project file is depicted in Figure 127. First of all, the replacement L-DALI needs to be configured with the appropriate IP settings and all relevant system settings (BACnet device ID, CEA-852 device settings, etc).

Start the LINX Configurator software stand-alone and connect via direct method (see Section 7.4.2). Load the LINX Configurator project file from the directory, which has been saved when the original L-DALI has been configured or modified. Double-check, if the DALI configuration seems ok and all DALI devices in use are assigned, that is, have a valid short address assigned. Then download the configuration to the L-DALI (see Section 7.4.4).

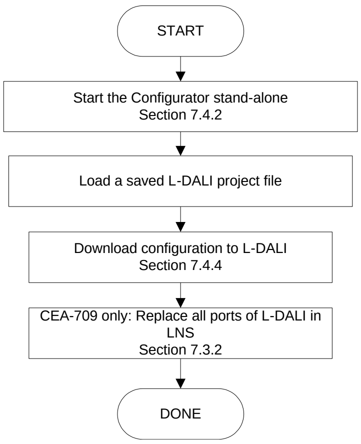


Figure 127: Basic work flow to configure a replacement device.

If using an LNS-based tool, all ports of the L-DALI device need to be replaced in that tool at some later point in time (see Section 7.3.2) as the NID has changed. If you are not using LNS, then refer to your network management tool's reference manual on how to replace a device.

7.3 CEA-709 Network Management/LNS Tools (LDALI-10X only)

7.3.1 Adding an L-DALI

To configure an L-DALI in your LonMaker drawing, the device needs to be added to the LNS database and commissioned. This Section refers to LonMaker TE and describes how to add an L-DALI to your database.

Note: *For the L-DALI versions supporting multiple DALI channels (LDALI-3E102 and LDALI-3E104) for each DALI channel used a LONMARK device must be added to the LNS database.*

To Add a Device to LonMaker TE

1. In your LonMaker drawing, drag a device stencil into the drawing. Enter an appropriate name as shown in Figure 128.

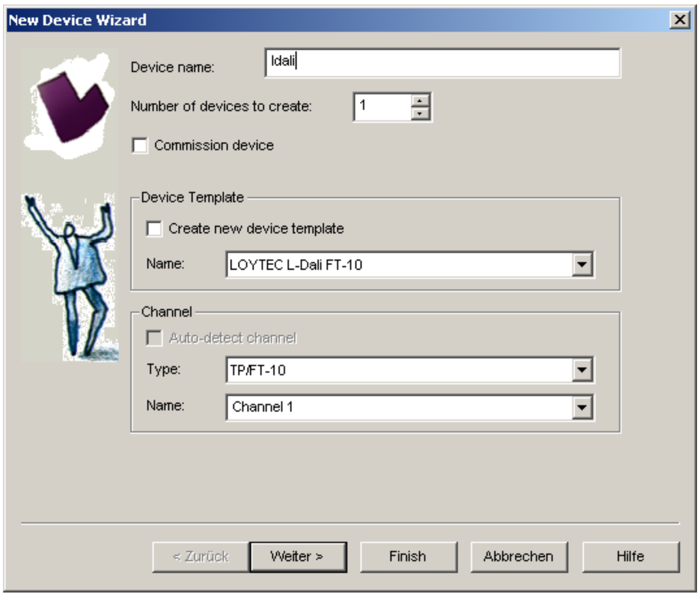


Figure 128: Create a new device in the drawing.

2. Select **Commission Device** if the L-DALI is already connected to the network.

3. In the **Device Template** group box select the existing device template of the L-DALI. Select “LOYTEC L-DALI FT-10”, if the L-DALI is configured to use the FT-10 interface, or “LOYTEC L-DALI IP-10L”, if the L-DALI is configured to be on the IP channel. For information on how to configure which port to use, refer to Section 5.2.8 for the Web UI. In addition, there are corresponding device templates for an L-DALI with enabled Emergency interface (“LOYTEC L-DALI emergency FT-10” and “LOYTEC L-DALI emergency IP-10L”, see Section 8.2). For custom CEA-709 interfaces a new device template must be generated. This can be done either by upload if the device is already on-line or by generating a XIF file if working off-line (see Section 7.7.12).

4. Select the channel, which the L-DALI is connected to and click **Next**.

5. The following dialog shown in Figure 129 appears, click **Next**.

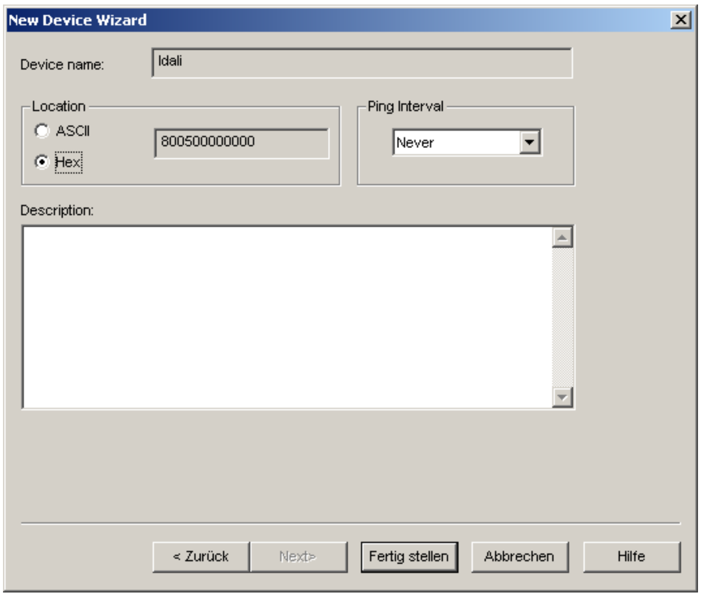


Figure 129: Leave defaults for Location.

6. Check Service Pin as the device identification method as shown in Figure 130 and click **Next**.

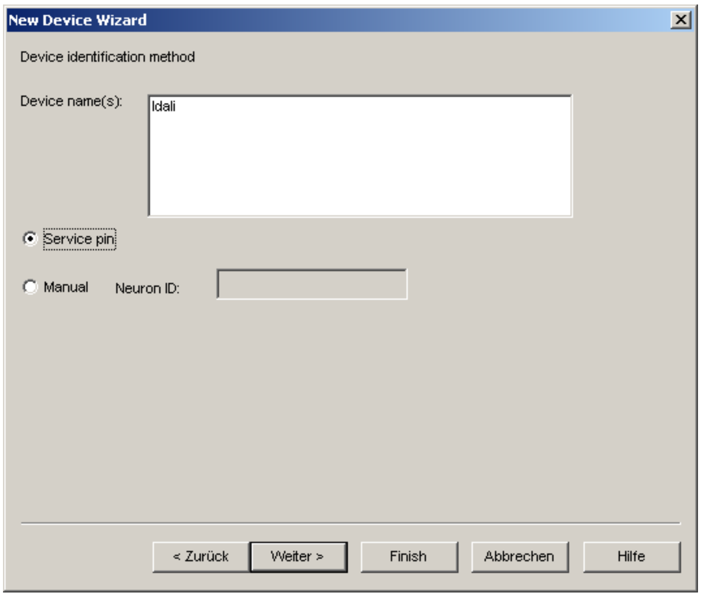


Figure 130: Use Service Pin.

7. Click **Next** in the following screens until you get to the final dialog shown in Figure 131.

8. If the device is already on-net, select **Online**. Also make sure to select **Current device values** for "Source of CP Values" (see Section 7.3.3 for details).

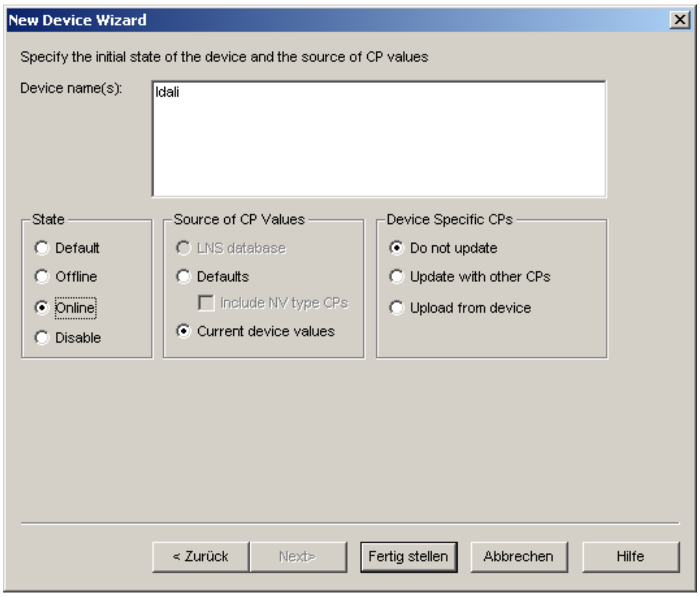
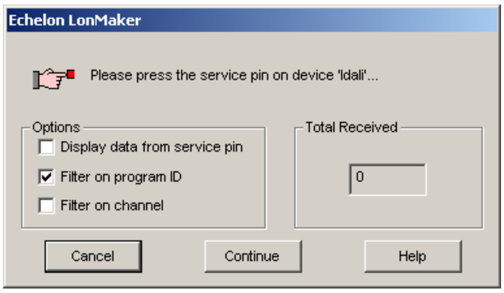


Figure 131: Final dialog.

9. Click **Finish**. A dialog will prompt to press the service pin.



10. Finally, you should get the device added to your drawing as depicted in Figure 132.

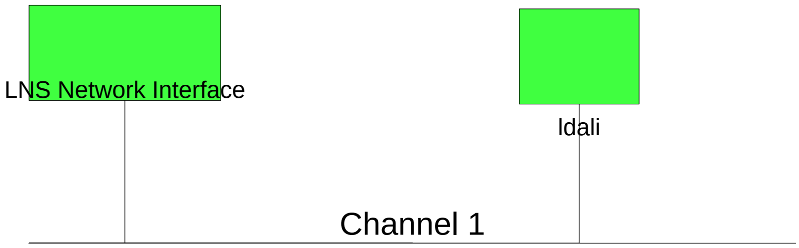


Figure 132: The L-DALI has been added to the drawing.

11. Repeat steps 1-10 for each CEA-709 node representing one DALI channel.

7.3.2 Replacing an L-DALI

This Section describes how to replace an L-DALI in your LNS database. The description refers to LonMaker TE. Let’s assume there is a device ‘ldali’ in the LNS database as shown in Figure 133.

Note: *For the L-DALI versions supporting multiple DALI channels (LDALI-3E102 and LDALI-3E104) for each DALI channel a LONMARK device representing that channel must be replaced in the LNS database.*

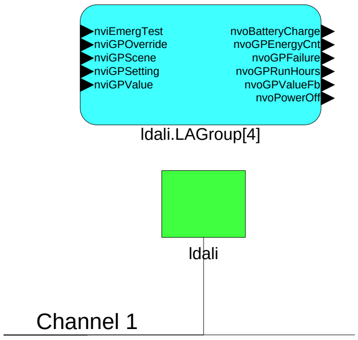


Figure 133: LonMaker drawing with one L-DALI.

Important!	***If using a non-standard network variable interface, make sure the appropriate data point configuration or backup was already loaded to the device using the IP connection!***

To Replace a Device in LonMaker TE

1. Select the device and right-click on the device shape.

2. Select **Commissioning → Replace…**. This opens the LonMaker Replace Device Wizard as shown in Figure 134.

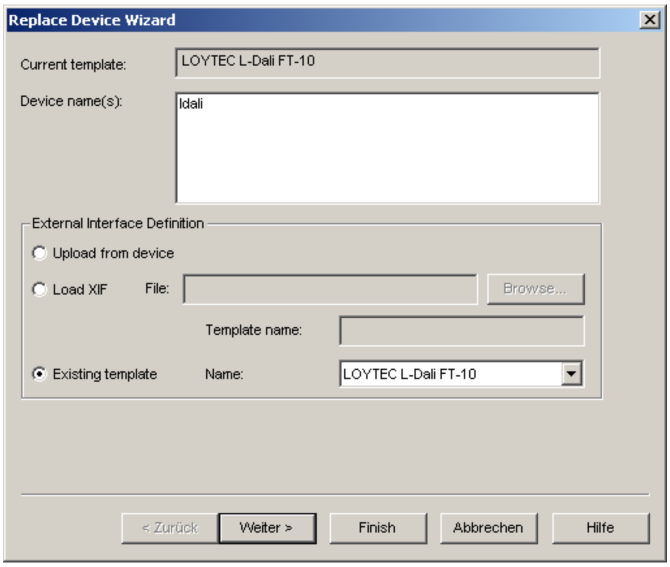


Figure 134: LonMaker replace device wizard.

3. Choose the existing device template and click **Next**.

4. In the following window shown in Figure 135 click **Next**.

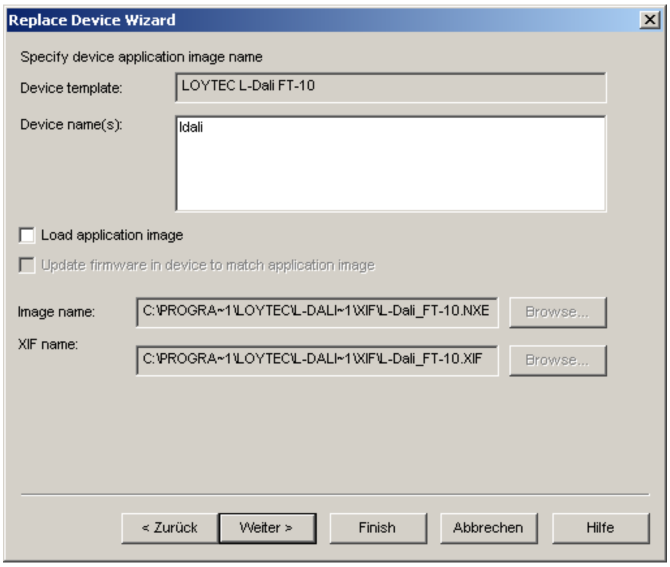


Figure 135: Click Next without loading an application image.

5. Then select **Online** for State and **New Device Values** for “Source of CP Values” as shown in Figure 136 and click **Next**.

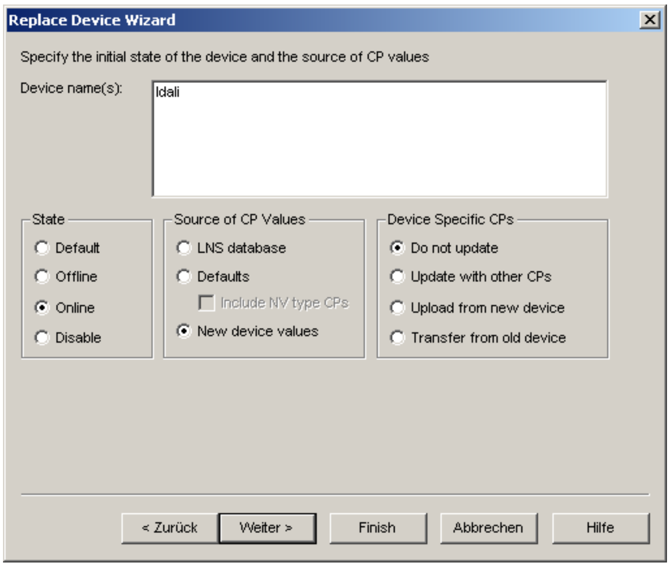


Figure 136: Select online state and source of CP values.

6. Select the **Service pin** method and click on **Finish** as shown in Figure 137.

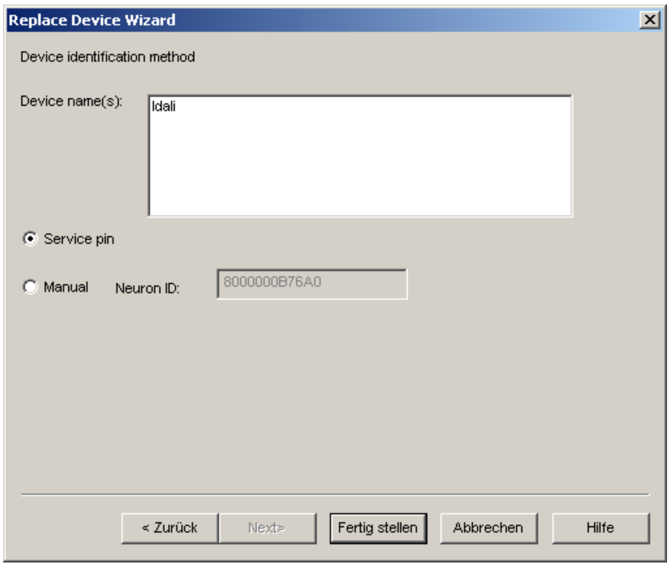
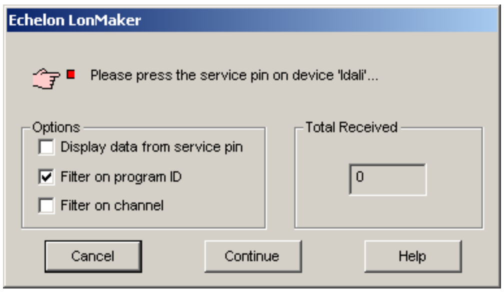


Figure 137: Select Service Pin and click Finish.

7. Then the service pin requestor opens. Press the service pin on the replacement L-DALI on the correct port. You can also send the service pin using the Web interface (see Section 5.1).



8. After the service pin has been received, LonMaker commissions the replacement device.

9. Repeat steps 1-8 for each CEA-709 node representing one DALI channel.

7.3.3 Working with Configuration Properties

On LDALI-10x models all application parameters which can be configured via the LINX Configurator (see Section 7.4.6) are available as LONMARK configuration properties. The LINX Configurator, on the other hand, directly modifies the values on the device when downloading its parameters (see Section 7.4.4). Or they were changed via the web interface. However, LNS based tools do not automatically read back CPs from the devices when browsing them. This can result in inconsistencies between the actual CP contents on the device and their copy in the network management tool. It is recommended to synchronize the CPs from the device into the LNS database before editing and writing them back.

Important!	***It is highly recommended to start the Configurator as LNS Plug-In when modifying and downloading parameters! In this case the Configurator will automatically synchronize the CP values to the LNS database.***

Note:	*Always choose* ***Current device values*** *or* ***New Device Values*** *for “Source of CP Values” when commissioning or replacing a device!*

To Synchronize CPs in NL220

1. Double-click on the device object in the device tree

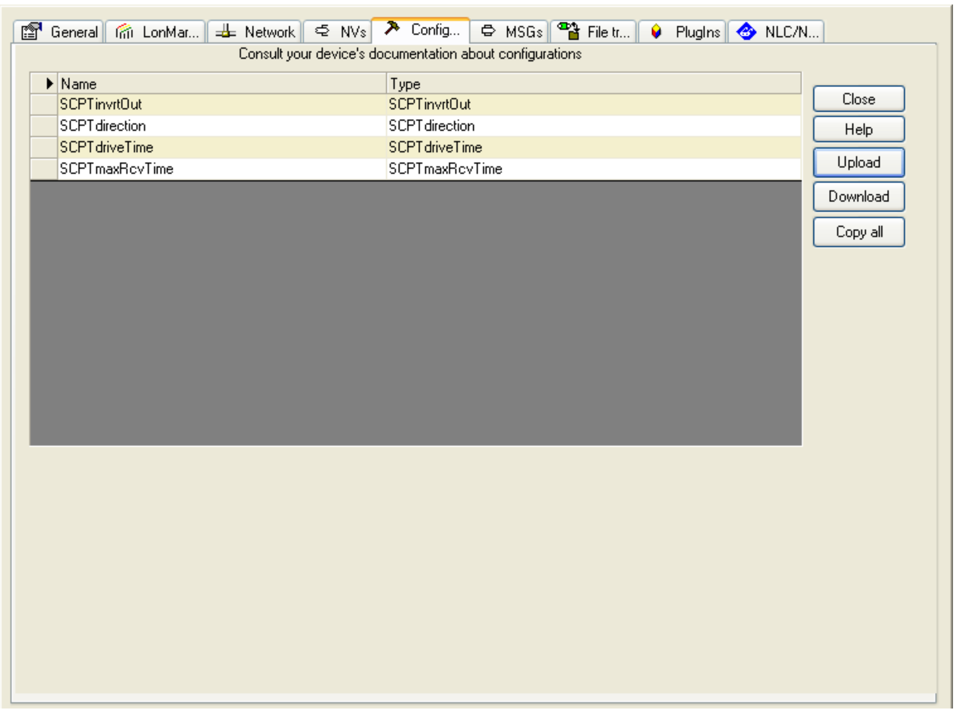


Figure 138: Configuration Tab for Configuration Properties in NL220.

2. Press the **Upload** button on the Configuration tab of the device properties (see Figure 138).

To Synchronize CPs in LonMaker TE

1. Right-click on a device object and select **Commissioning → Resync CPs…** from the context menu.

2. This opens the dialog shown in Figure 139.

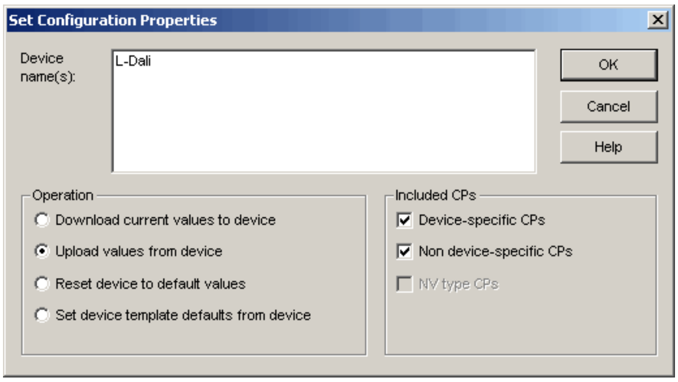


Figure 139: Set Configuration Properties in LonMaker TE.

3. In this dialog select the radio button **Upload values from device** in the **Operation** group box. To use the current settings of the device as default values for new devices, select **Set device template defaults from device**.

4. Execute the operation by clicking the **OK** button.

7.3.4 Enable Legacy NM Mode

For network management tools, which do not support the ECS (extended command set) network management commands, the legacy network management mode must be configured. Please contact the tool's vendor for information whether ECS is supported or not. Note, that changing to legacy network management mode changes the static interface of the device and thus requires different XIF files.

The legacy mode can be enabled using the Web Interface (see Section 5.2.11) or the **System Settings** tab in the **Project Settings** dialog of the Configurator Software (see Section 7.16.2).

7.4 Using the LINX Configurator

7.4.1 Starting as an LNS Plug-In (LDALI-10X only)

In LonMaker the plug-in is started by right-clicking on the L-DALI device shape and selecting **Configure…** from the pop-up window.

In NL-220, the Plug-in is started by right clicking on the L-DALI node, then selecting the Option **LOYTEC LINX Configurator** in the **PlugIns** sub menu.

In Alex, the Plug-in is started by right clicking on the L-DALI device and selecting the **LOYTEC LINX Configurator** in the **Starte PlugIn** sub menu.

A window similar to what is shown in Figure 140 should appear.

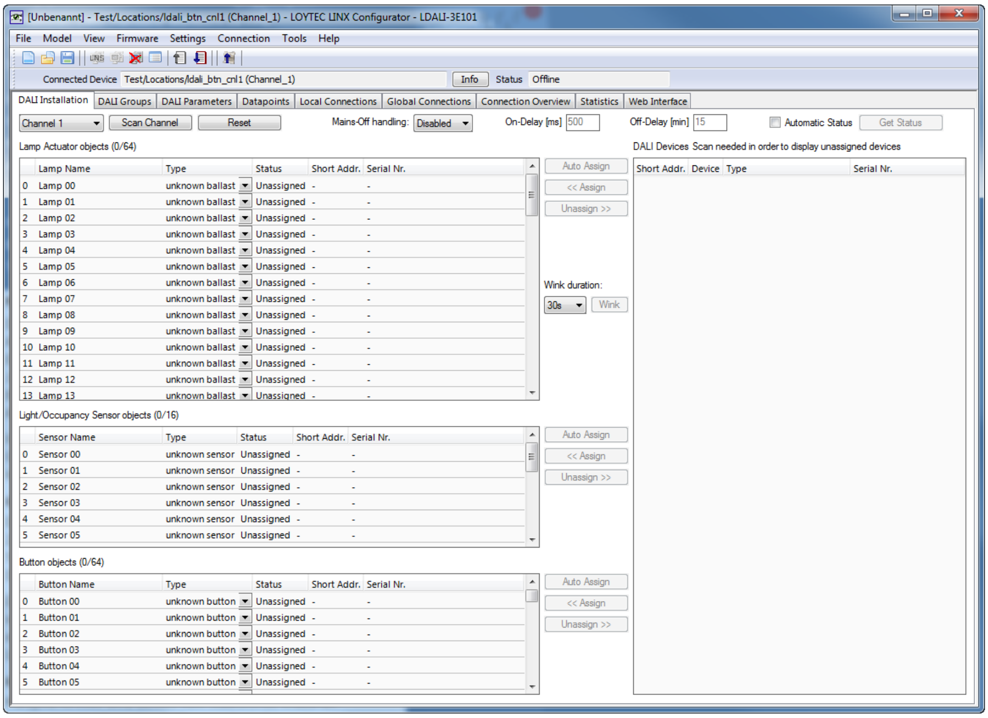


Figure 140: LINX Configurator main window.

Note: *It is recommended to start the LINX Configurator as LNS-Plug-In, whenever using a LDALI-10X with an LNS based network management tool (LonMaker, NL220, etc.)! This allows the Plug-In to keep the device configuration in-sync with the LNS database.*

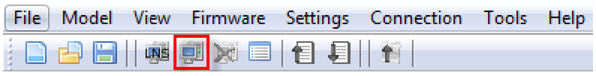
7.4.2 Starting Stand-Alone

The L-DALI can also be used without LNS-based tools. In this case, the LINX Configurator needs to be started as a stand-alone application. Go to the Windows **Start** menu, select **Programs**, **LOYTEC LINX Configurator** and then click on **LINX Configurator**. This starts the LINX Configurator.

If the L-DALI is not yet connected to the network, go to the **Model** menu and select the L-DALI model to be configured. If the L-DALI is already connected to the network it is recommended to connect the configuration software to the L-DALI.

To Connect to an L-DALI Stand-Alone

1. Select the direct connection method by clicking on the **Connect to device** button



in the tool bar of the main connections window. The connect dialog as shown in Figure 141 opens.

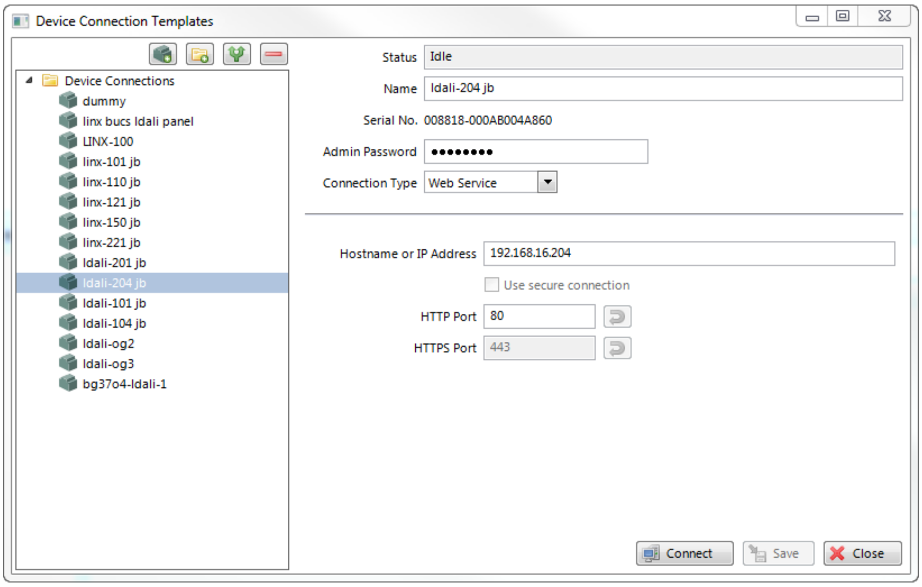


Figure 141: Connection dialog.

2. To add a new device connection, click on the **New Device Connection** button  or select an existing connection in the tree on the left-hand side and click on the **Duplicate** button .

3. Enter a name for the connection.

4. Enter the admin password. The default password is 'loytec4u' (older firmware versions used 'admin').

5. Choose a **Connection Type**. Possible options are:

   a. **Web service** (recommended): Firewall friendly connection using HTTP.

   b. **TCP/IP**: This uses the IP protocols FTP and Telnet to connect to the device.

c. **CEA-709 (NIC)**: Connection via a LOYTEC CEA-709 network interface.

6. For IP based connection methods enter the host name or IP address of the device. If your device is located behind a NAT router or firewall, you may change the FTP, Telnet, or HTTP ports respectively to your needs for accessing the device.

7. For a CEA-709 based connection the CEA-709 address information must be entered and a LOYTEC network interface has to be selected.

8. Click on **Save** to store that connection.

Note: *If you connect without having the connection settings saved, a dialog asks whether to use the changed settings temporarily for this connection only. In this case the existing connection is not altered.*

9. Organize device connections into folders. To add a new folder, click on the **New Folder** button  and enter a folder name. Drag the new device connection onto the new folder.

10. Click on **Connect**. This establishes the connection to the device.

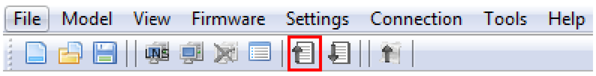
7.4.3 Uploading the Configuration

To get the configuration of the L-DALI it needs to be uploaded. This allows uploading the entire configuration from the L-DALI, including DALI configuration, light and sunblind application parameters, system settings, AST configuration, and schedules. Optionally, parts of the configuration can be omitted from upload.

Note: *To speed up configuration upload and download LOYTEC recommends using an IP connection. If an IP connection is available the LINX Configurator will try to use it, even if the software was started as LNS Plug-In.*

To Upload a Configuration

1. Click on the **Upload Configuration** button



in the tool bar. The configuration upload dialog opens up as shown in Figure 142.

2. Choose which configuration items shall be uploaded:

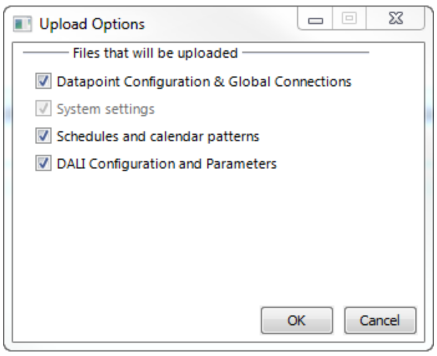


Figure 142: Upload Options Dialog.

- **Datapoint Configuration**: Contains everything configured on the Datapoints tab (see Section 7.7), the Local and Global connections tabs (see Section 7.9) and in the project settings except for the system settings (see Section 7.16). Typically this includes local alarm, scheduler, calendar, and trendlog objects, e-mail templates, math objects and user registers.

- **System settings**: Contains the configuration setup in the System Settings tab of the Project Settings (see Section 7.16.2)

- **Scheduler and calendar patterns**: Contains the schedule and calendar configuration ("contents" of scheduler and calendar objects).

- **DALI Configuration and Parameters**: The DALI Configuration contains the configuration of the DALI network, including device names, device types, device assignment (DALI short address), group names and group assignment as configured on the DALI Installation, DALI Groups and DALI Channel tabs (see Section 7.4.6). The Parameters contain the parameters of the L-DALI light and (if available) sunblind applications, including CLC Bindings and button functions as configured on the Parameters tab (see Section 7.6).

3. Click on the button **Upload Configuration…** to start the transfer. This will upload the selected configuration items. The configuration items not selected for upload will remain as before the upload.

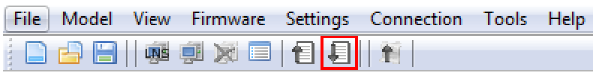
7.4.4 Configuration Download

After a configuration was created or changed, it needs to be downloaded to the L-DALI. For doing so, the L-DALI must be online. If the L-DALI is not yet connected to the network, the configuration can be saved to a project file on the local hard drive.

Note: *To speed up configuration upload and download LOYTEC recommends using an IP connection (FTP). If an IP connection is available the LINX Configurator will try to use it, even if the software was started as LNS Plug-In.*

To Download a Configuration

1. In the main connections window, click on the **Download Configuration** speed button



in the tool bar of the main connections window. This will open the configuration download dialog as shown in Figure 143.

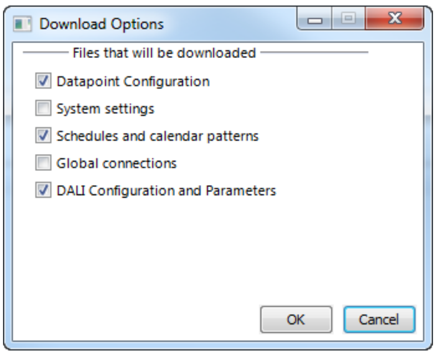


Figure 143: Download Options Dialog

2. Choose which configuration items shall be:

   - **Datapoint Configuration**: Contains everything configured on the Datapoints tab (see Section 7.7), the Local connections tab (see Section 7.9) and in the project settings except for the system settings (see Section 7.16). Typically this includes local alarm, scheduler, calendar, and trendlog objects, e-mail templates, math objects and user registers.

   - **System settings**: Contains the configuration setup in the System Settings tab of the Project Settings (see Section 7.16.2)

   - **Scheduler and calendar patterns**: Contains the schedule and calendar configuration ("contents" of scheduler and calendar objects).

   - **Global connections**: Contains everything configured in the Global connections tab (see Section 7.9.7). Do not download Global connections using the LINX Configurator if using LWEB-900 software to configure and manage global connections!

   - **DALI Configuration and Parameters**: The DALI Configuration contains the configuration of the DALI network, including device names, device types, device assignment (DALI short address), group names and group assignment as configured on the DALI Installation, DALI Groups and DALI Channel tabs (see Section 7.4.6). The Parameters contain the parameters of the L-DALI light and (if available) sunblind applications, including CLC Bindings and button functions as configured on the Parameters tab (see Section 7.6).

Tip! *To speed up your work flow only download the configuration items changed. In applications where no AST features are used it is* ***sufficient to download DALI Configuration and Parameters****!*

3. After clicking on **OK** the **Configuration Download** dialog appears as shown in Figure 144. Click **Start** to start the download. Each of the actions is displayed in the **Task List** section of the dialog. The current progress is indicated by the progress bar below.

Note: *As new DALI devices are commissioned during the download of the DALI configuration this operation may take some time depending on the number of DALI devices to be commissioned.*

4. When the download process has finished, a notification window appears, which has to be acknowledged by clicking **OK**. If an error occurred details are available when clicking on **Show Details**.

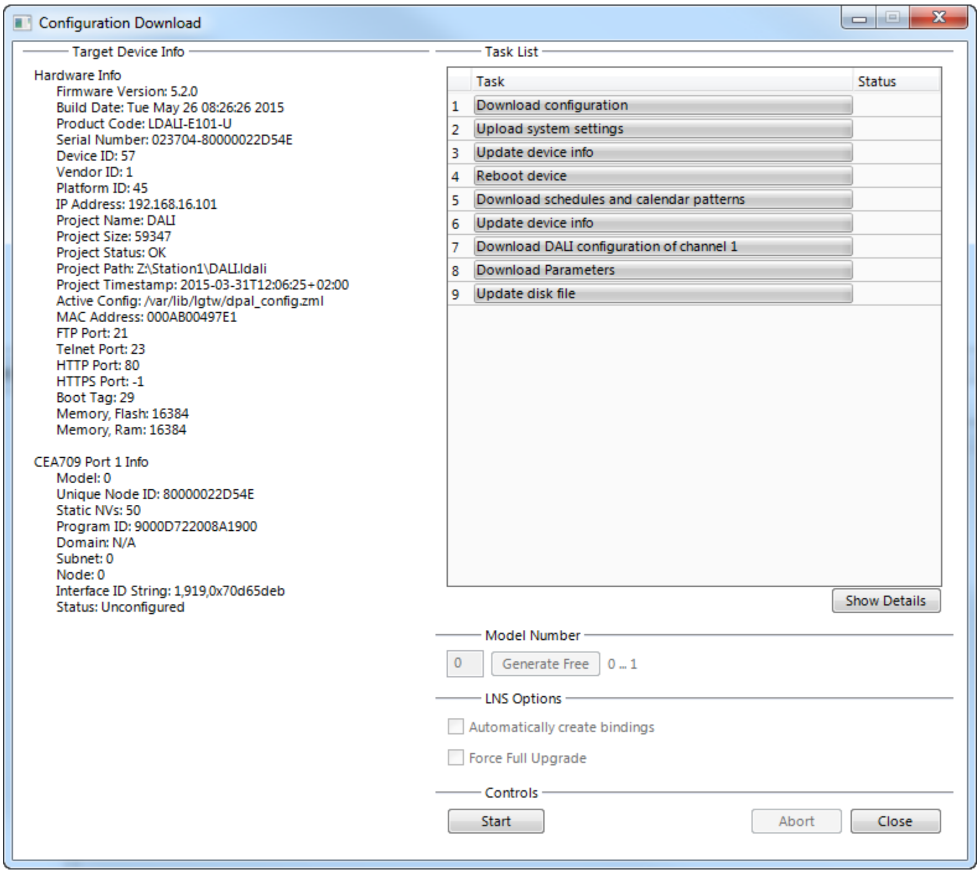


Figure 144: Configuration Download Dialog.

If the dialog shown in Figure 145 appears, the software has detected a version mismatch between the DALI configuration in the device and the one in the Configurator.

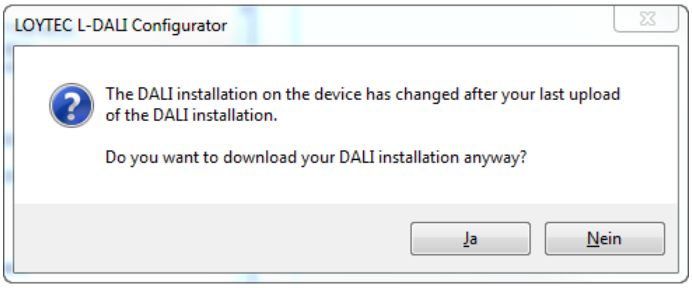


Figure 145: DALI configuration mismatch.

Possible reasons for this message are:

- The DALI configuration on the device was changed using the web interface.
- An old configuration file version was loaded to the LINX Configurator software.
- No configuration file was loaded to the LINX Configurator software but the device contains a DALI configuration.

If you are sure, you are loading a matching DALI configuration to the device, you can answer the question by clicking on **Yes**. In any case the log (“Show details”) will contain the message “Syncro counter did not match” for each affected channel.

If errors occurred during the configuration download, clicking on the button **Show Details** opens a Log-window. The Configuration Download Details log window as shown in Figure 146 opens up.

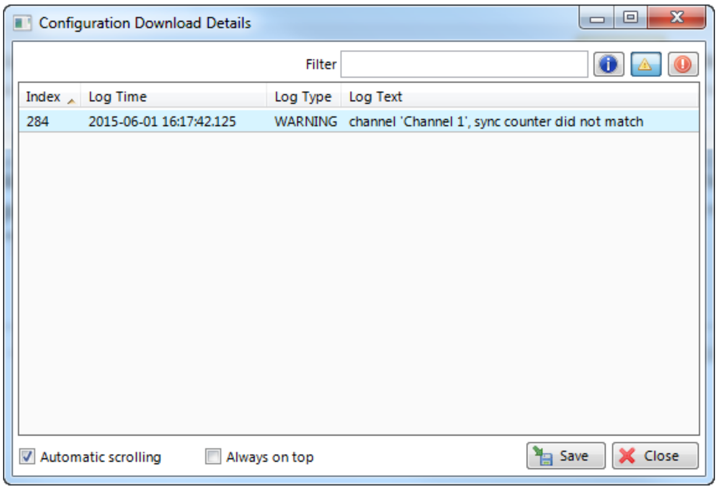


Figure 146: Configuration Download Details Log.

To filter out info messages use the button . Similar warning messages can be filtered by pressing the button and error message by pressing the button . Press the **Save** button to store the log file on your harddrive (e.g. for sending in to LOYTEC support for analysis).

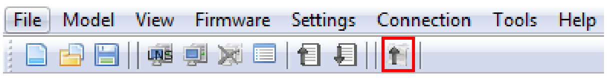
7.4.5 Upload the System Log

The system log on the device contains important log messages. Log messages are generated for important operational states (e.g., last boot time, last shutdown reason) or errors at run-time. This file is important for trouble-shooting and is available on the Web UI (see Section 5.5.1). The file can also be uploaded from the device with the LINX Configurator.

To Upload the System Log

1. Connect to the L-DALI via the IP or LNS method (see Section 7.4.2).

2. Click on the **Upload system log** button



in the tool bar. The upload system log dialog as shown in Figure 147 opens showing the upload progress.

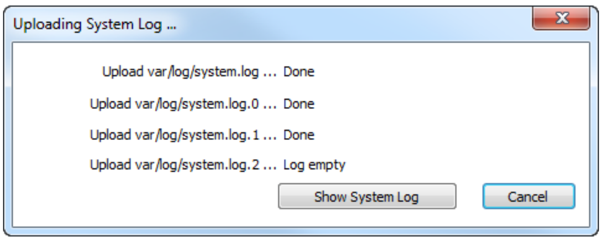


Figure 147: Upload system log dialog.

3. When the upload is finished, click on **Show System Log**. The system log window appears as shown in Figure 148.

4. Click on **Save** to store the system log into a file on your local hard drive.

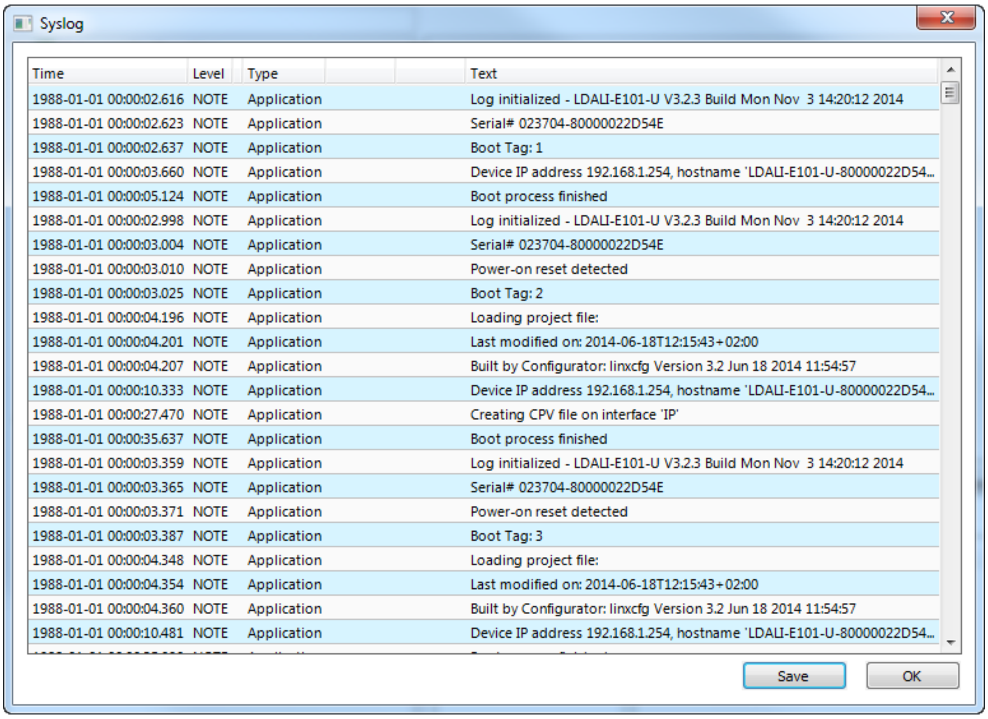


Figure 148: System log window.

7.4.6 Backup and Restore

The Configurator provides a backup and restore function for the connected device. It is highly recommended to create a device backup once the device configuration has been completed. This backup can be used in the case a device needs to be replaced in the network.

To Create a Backup

1. Connect to the device (see Sections 7.4.1 and 7.4.2).
2. Choose the menu **Tools** → **Backup Device Configuration …**.
3. A file requestor opens. Choose a location to store the ZIP file of the device backup. The suggested file name contains device IP address and creation date.
4. Click on **Save**. The backup is being uploaded.

To Restore a Backup

1. Choose the menu **Tools** → **Restore Device Configuration …**.
2. In the file requestor choose a backup ZIP file and click **Open**.
3. The Configurator restores and reboots the device. The process is complete when the device has finished rebooting.

7.5 DALI Installation

To install and manage the DALI channels connected to the L-DALI the tabs **DALI Installation**, **DALI Groups**, and **DALI Channels** are used:

- The **DALI Installation** tab is used to assign DALI ballasts to Lamp Actuator objects, DALI sensors to Light and Occupancy Sensor objects, and DALI buttons to Button objects. Additionally, a name can be assigned to each ballast, sensor, and button. For details see Section 7.5.1.

- The **DALI Groups** tab is used to assign ballasts to DALI groups. Additionally, a name can be assigned to each group. For details see Section 7.5.2.

- The **DALI Channels** tab can be used to virtually connect two DALI channels (“bridging”). Additionally, a name can be assigned to each channel. For details see Section 7.5.3.

7.5.1 DALI Installation Tab

The DALI Installation tab is shown in Figure 149. It is used to scan a DALI channel and assign DALI devices to Lamp Actuator, Light/Occupancy Sensor objects, and Button objects. The dialog is divided into three sections:

- The channel selection and functions section (number 1 in Figure 149).
- The DALI network database with the device assignment (number 2 in Figure 149).
- The DALI network scan results (number 3 in Figure 149).

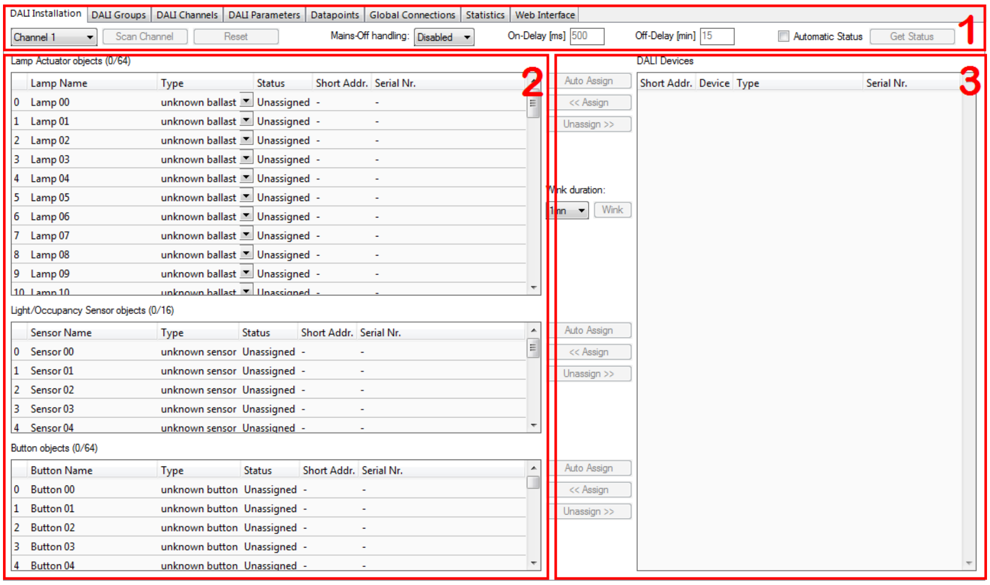


Figure 149: DALI Installation tab.

7.5.1.1 Channel Selection and Functions Section

The following functions are available in on-line and off-line mode:

- The drop down box to the left of this section allows you to choose the DALI channel to work with. The default selection is **Channel 1**. The number of channels available depends on the L-DALI model.

- In the middle of this section the drop down box **Mains-Off handling** and the parameters **Mains-On delay** and **Mains-Off delay** are found. This function allows saving the standby energy consumed by the ballasts, by switching off the ballast mains when all ballasts on a channel are off. For further details see Section 8.4.7.

- The **Export** and **Import** buttons on the right side allow exporting and importing the DALI configuration of the selected channel to/from an XML file. This includes DALI device assignment, device names, group assignment, group names, and the channel configuration.

If you are online the following additional functions are available:

- The **Scan Channel** button starts a scan of the DALI channel. When pressing the button a window showing the scan progress will be shown (see Figure 150). After the scan the devices found will be listed in the DALI network scan results section (number 3 in Figure 149) of the DALI Installation tab.
- Pressing the **Reset** button will clear the DALI configuration of the selected channel including all DALI device assignments, device names, group assignment, group names, and the channel configuration. If on-line it will also reset all DALI devices on the channel to factory defaults.

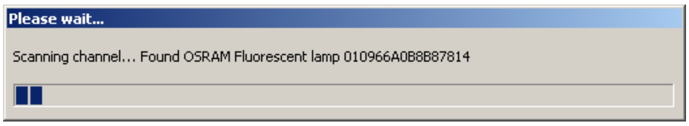


Figure 150: DALI channel scan progress.

- The **Automatic Status** check box allows enabling and disabling the periodic update of the status of the devices in the DALI network database. Pressing **Get Status** allows updating the device status manually if automatic status update is disabled.

Note: *If there is no IP connection available, disable automatic status update in case there is no need to monitor the status of DALI devices as the status update uses network bandwidth.*

7.5.1.2 DALI Network Database Section

This section allows naming DALI devices (ballasts, sensors, and buttons) and assigning them to the Lamp Actuator, Light/Occupancy Sensor, and Button objects (see Section 8.2 and 8.3). Optionally the DALI device type can be configured for unassigned ballasts and buttons. The upper table contains 64 entries, one for each Lamp Actuator of the channel, the middle table contains 16 entries, one for each Light and Occupancy Sensor of the channel, and the lower table contains 64 entries, one for each Button object of the channel.

For each entry the tables contain the following columns:

- **Lamp/Sensor/Button name**: Contains the name of the DALI device. Double click on the name to modify it. The name should be chosen in a way which allows identifying the lamp (e.g. room number/ballast number). This is specifically important in off-line work flow (see Section 7.2.3) for the installer to pick the correct lamp during the final on-line phase and to identify a lamp if it reports an error.

- **Type**: Show the type and optionally the make of the DALI ballast, sensor, or button assigned. For assigned and online ballasts this information is read from the ballast. In case of unassigned ballasts and buttons, the DALI device type can be chosen using a drop-down box or **Set device type…** from the context menu (can be used with multi-select). Selecting the device type allows to configure DALI device type specific parameters during off-line preparation including the button functions. Further, this information is used during device assignment when on-line to ensure only a matching device type can be assigned. For information on the device type reported by a ballast please refer to the documentation provided by the vendor of the ballast.

- **Status**: Shows the status of the DALI device if a DALI ballast, sensor, or button is assigned and the DALI configuration was downloaded. If the DALI device is online the status is **OK**, if it is not reachable via the DALI network it is **Offline**. The device is marked **modified** if its configuration/assignment was changed, but not yet downloaded to the L-DALI. If **Automatic Status** is checked the status is periodically updated. Press **Get Status** to manually trigger an update of the Status column.

- **Short Addr**.: Shows the DALI short address of the device. The DALI short address can be in the range 0 to 63.

- **Serial Nr.**: Shows the serial number of the device, if available. If the serial number of the device is known during preparation phase of off-line work flow (see Section 7.2.3) it can be entered by double clicking on this field. In this case the assignment is automatically performed once the DALI configuration is downloaded to the L-DALI.

The list can be sorted by clicking on one of the column headers.

A device can be winked by selecting the corresponding row and clicking on the **Wink** button located between the DALI network database section and the DALI scan result section.

7.5.1.3 DALI Network Scan Result Section

If a DALI network scan has been performed by pressing the **Scan Channel** button, the unassigned devices found during the scan are displayed in the table in the DALI network scan result section. In case an error occurs see Section 11.4 for a description of the error codes and possible reasons.

For each entry the tables contain the following columns:

- **Short Addr.**: Shows the DALI short address of the device. For devices which do not contain a short address L-DALI automatically assigns one during network scan.

- **Device**: Can be **Ballast**, **Sensor**, **Button**, or **Unknown**. Unknown devices are devices not supported by the L-DALI (e.g. switches). See Section 8.4.3 for details on the different DALI device types.

- **Type**: Show the type and optionally the make of the DALI device.

- **Serial Nr**.: Shows the serial number of the device, if available.

The list can be sorted by clicking on one of the column headers.

Devices found during a DALI network scan can be assigned to Lamp Actuator, Sensor, or Button objects either one by one or using the **Auto Assign** button.

To manually assign a device select one device in the scan results list on the right side and an unassigned entry in the Lamp Actuator, Sensor, or Button objects lists on the left side and press the **Assign** button. Alternatively you can drag a device from the scan results list and drop it to an unassigned entry in the object lists. DALI ballasts must be assigned to Lamp Actuator objects. DALI sensors must be assigned to Light/Occupancy Sensor objects. DALI buttons must be assigned to Button objects.

To remove an assignment from an object select the corresponding entry and press the **Unassign** button. The entry in the object list will become unassigned again and the DALI device will be moved to the scan results list to the right.

To identify a device in the scan results list select the corresponding row and click on the **Wink** button. The duration of the **Wink** action can be chosen in the drop down box besides the button. When a device winks it dims between its minimum and maximum value for the specified wink duration.

Tip! *Use auto assign if you do not care about the assignment of a specific device. This is typically the case if you are using the on-line work flow (do not prepare DALI installation off-line) and you do not use the Lamp Actuator objects to control the lights (rather use Group or Channel Actuator objects).*

7.5.2 DALI Groups Tab

The DALI Groups tab is shown in Figure 151. It is used to assign DALI ballasts to groups and setup group names.

Similar to the DALI Installation tab the drop down box in the top left corner of the tab allows you to choose the DALI channel to work with. Below there is a table which contains one row for each DALI ballast (64) and one column for each DALI group (16). To add/remove a device to/from a group check/uncheck the check box where the corresponding row and column are crossing. The default name of a group (e.g. “Group 00”) can be changed by double clicking on it.

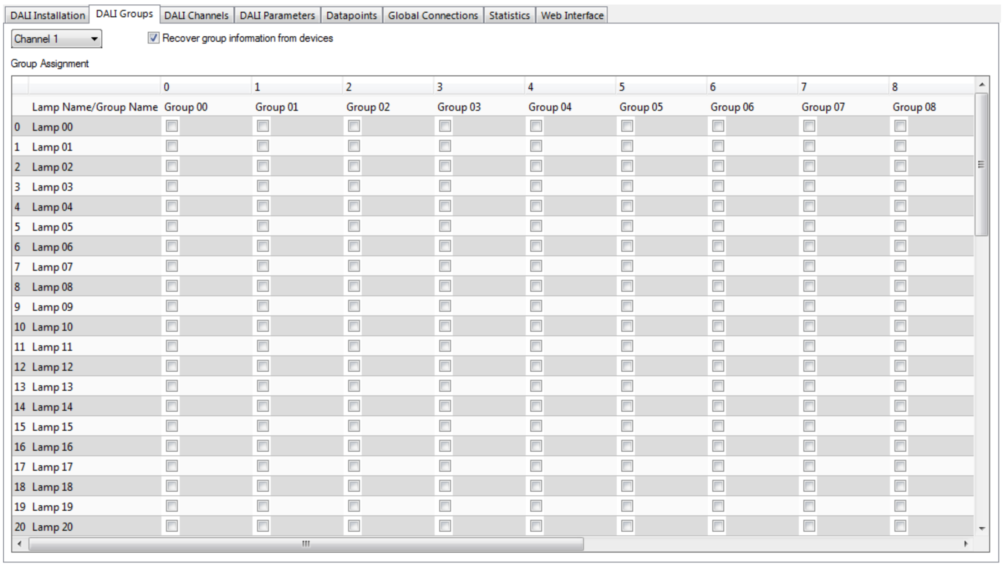


Figure 151: DALI Groups tab.

The checkbox **Recover group information from devices** allows keeping the group configuration in the devices (ignoring the group configuration defined in the LINX Configurator). It is automatically unchecked if the group configuration is changed with the LINX Configurator.

7.5.3 DALI Channels Tab

The DALI Channels tab is shown in Figure 152. It is only available if the device is equipped with more than one DALI channel. This tab is used to change channel names and to configure DALI channel bridging.

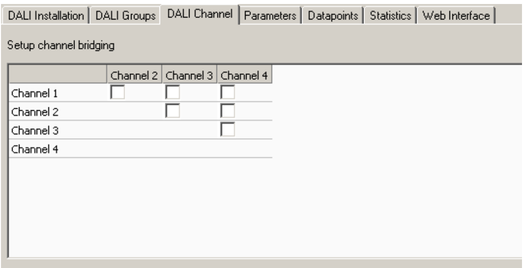


Figure 152: DALI Channels tab.

The default name of a channel (e.g. “Channel 1”) can be changed by double clicking on it in the first row of the table.

To connect two channels by channel bridging, check the check box where the corresponding row and column are crossing. For more details on the DALI channel bridging function refer to Section 8.4.6.

7.6 Parameter Configuration

The parameter configuration is located on the Parameters tab as shown in Figure 153. The dialog is divided into the following sections:

- The tree view (number 1),
- the LONMARK object list (number 2),
- the parameter view (number 3),
- the constant light controller bindings and button functions view (number 4) and
- buttons for special functions (number 5).

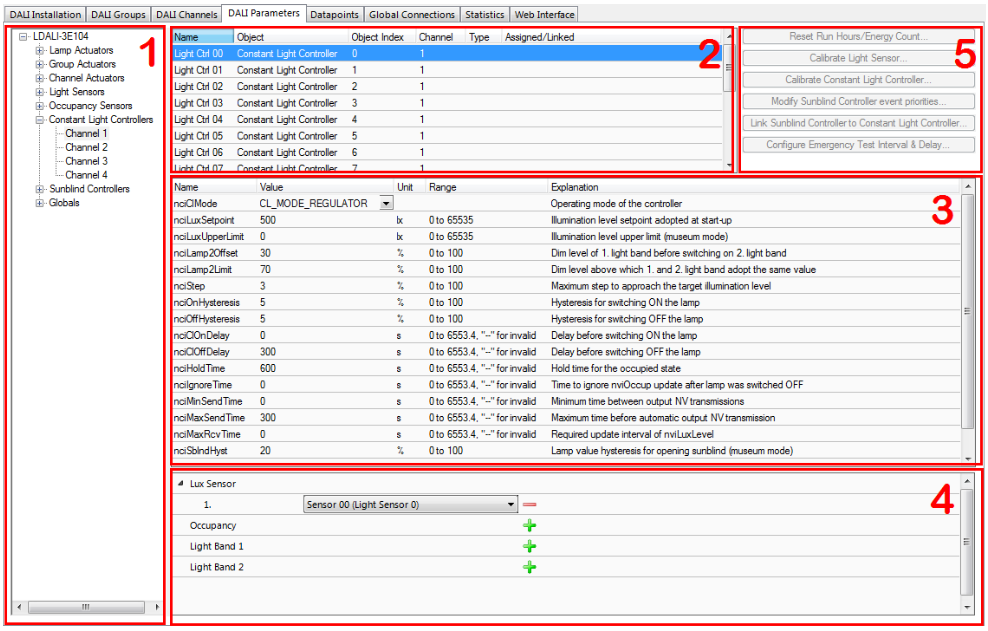


Figure 153: Parameter Configuration Dialog

- The selected node in the tree view (1) determines which LONMARK objects are displayed in the list view (2). If the root node (**LDALI-3E104**) is selected, all LONMARK objects are displayed, if a branch is selected, the LONMARK object in or below this branch are displayed. For example, selecting **LDALI-3E104/Lamp Actuators** displays all lamp actuator objects on all DALI channels, whereas selecting **LDALI-3E104/Lamp Actuators/Channel 1** displays only lamp actuators of DALI channel 1.

In the LONMARK object list (2) one or multiple objects can be selected. The properties of the selected object(s) are displayed and can be edited in the parameter view (3). In this way it is possible to change a parameter for multiple objects at once. All parameters are described in Section 8.2 and 8.3, respectively.

In case a single Constant Light Controller is selected the constant light controller bindings section (4) becomes active and allows determining which sensors (occupancy & lux) are used as inputs to and which light groups are controlled by the selected constant light controller instance.

Click on the to add an input or output. Use the drop down box to select a sensor (input) or a light group (output). Click on to remove an input or output. When adding the first manual CLC binding on a channel all automatic bindings of that channel will be disabled. Automatic bindings can be re-enabled in the CEA-709 Settings tab of the Project Settings (see Section 7.16.2).

For each constant light controller one lux level sensor, but up to 16 occupancy sensor inputs can be configured. Whenever at least one occupancy sensor reports occupancy the area controlled by the controller instance is considered occupied.

Similar for each of the two light bands up to 16 DALI groups can be selected as outputs. The primary light band is near the inside of the building, the secondary light band is near the window front. Depending on the outdoor light intensity the primary light band has to be brighter than the secondary light band to illuminate the room evenly. NV bindings can be used in parallel to the internal CLC bindings.

In case one or more Button objects supporting button functions are selected the section for button function configuration (4) becomes active (see Figure 154). It allows the configuration of the function executed when a button is operated.

Note: *Not all DALI button device types are configurable. Therefore, a button device type supporting button functions must be selected or assigned on the DALI installation tab before the corresponding button object can be configured.*

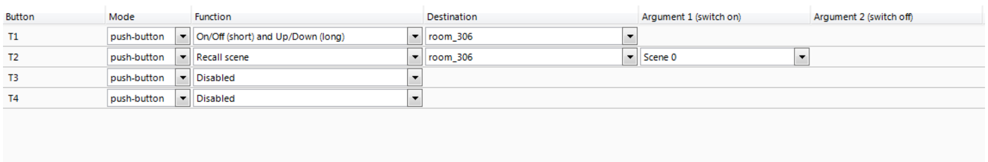


Figure 154: Button function configuration.

For DALI push-button coupler devices the **Mode** drop down box allows to select whether a switch or a push button is connected to the button input. The **Function** drop down box allows selecting the function to be performed when the button is pressed. See Table 11 for a list of available function. If the function is used to control light, the **Destination** column allows selecting the DALI group or channel to be controlled. If the function requires additional arguments (e.g. dim values or scene numbers) they are configured in the columns **Argument 1** and **Argument 2**.

Multi-select different button objects to change the configuration of multiple buttons at once. Only changes are applied to the selected objects. The rest of their configuration remains as is. This allows to quickly change selected values (e.g. button function) for multiple button objects.

If DALI buttons are used with Constant Light Controller applications to allow manual and automatic operation please see Section 8.2.8.9 (LONMARK) and 8.3.5.8 (BACnet) on how DALI buttons can interact with the Constant Light Controller application.

Depending on the selected objects the following special functions (5) are available:

- **Lamp Actuators**: Reset run hours and Energy Count
- **Light Sensors**: Calibrate light sensor
- **Constant Light Controllers**: Calibrate constant light controller
- **Sunblind Controllers**: Modify sunblind controller event priorities, link sunblind controller to constant light controller

7.6.1 Reset Run Hours and Energy Count

To reset the run hours and/or energy count of a lamp actuator click on the **Reset Run Hours/Energy Count…** button. A dialog is displayed which allows you to specify what you want to reset (see Figure 155).

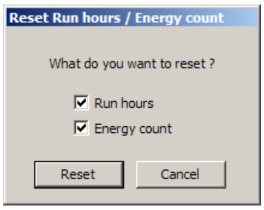


Figure 155: Reset Run Hours and Energy Count

7.6.2 Calibrate Light Sensor

To calibrate the light sensor click on the **Calibrate Light Sensor …** button. The dialog shown in Figure 156 is opened.

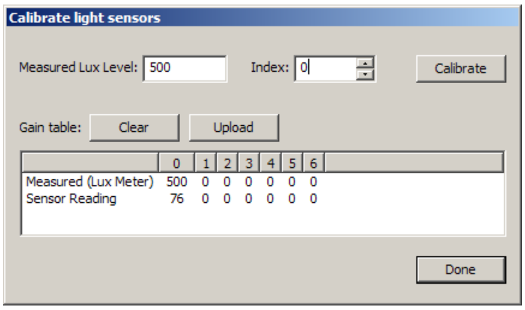


Figure 156: Light Sensor Calibration

To counter any non-linearity the sensor can be calibrated for up to seven different daylight illumination levels. However, in most cases it is sufficient to calibrate the sensor only once. The best result is achieved if the illumination level is near the setpoint. To calibrate the sensor perform the following steps:

1. Measure the current lux level at the reference area (e.g. desk) using a luxmeter.
2. Enter the measured lux level in the input field **Measured Lux Level**.

3. Select an unused index.

4. Click on the **Calibrate** button.

5. If necessary repeat steps 1.– 4. For different daylight illumination levels.

6. Press the **Done** button.

The current gain table can be uploaded from the device by pressing the Upload button. It can be cleared with the Clear button.

7.6.3 Calibrate Constant Light Controller

To calibrate the constant light controller click on the **Calibrate Constant Light Controller …** button. The dialog shown in Figure 157 is opened.

Artificial light calibration

Artificial light calibration is required if

1. the artificial light in the controlled area/room is dimensioned outside the range of 500lx to 700lx. In a typical office room artificial light is dimensioned for around 600lx (500lx + 20%). If in the rooms artificial light is dimensioned dimmer or brighter the artificial light factor is used to parameterize the amount of lux provided by the rooms artificial light.

2. the difference in sensor sensibility for artificial light vs. for natural light must be considered. While the natural light comes through the window, the artificial light is provided by luminaires mounted on the ceiling. Thus, the reflection and light distribution is different depending on the light source. This results in a different ratio of the lux level measured by the sensors mounted on the ceiling to the lux level on the reference area (e.g. desk). In most use cases this difference is negligible. However, if it is not the artificial light factor is used to parameterize this influence.

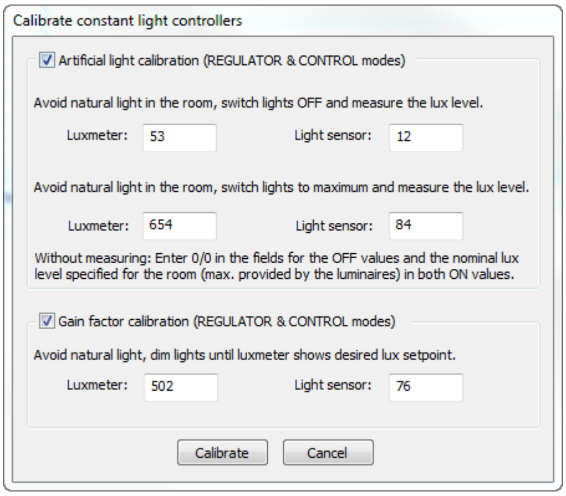


Figure 157: Calibrate Constant Light Controller.

To measure the amount of light provided by the rooms artificial light the following steps have to be performed:

1. If possible darken the room.

2. Switch the lamp off. Measure the lux value on the reference area (e.g. desk) with a luxmeter and enter it in the first **Luxmeter** input field. Enter the value measured by the light sensor in the first **Light Sensor** input field.

3. Switch the lamp on. Measure the lux value on the reference area (e.g. desk) with a luxmeter and enter it in the second **Luxmeter** input field. Enter the value measured by the light sensor in the second **Light Sensor** input field.

4. Click on the **Calibrate** button.

If the lighting output of the rooms artificial lighting is known (in lux on the reference area) you can use those values as an approximation instead of measuring:

1. Enter 0 in both “OFF” fields (Luxmeter and Light sensor).

2. Enter the lux value corresponding to the lighting output of the room in both “ON” fields (Luxmeter and Light sensor).

3. Click on the **Calibrate** button.

Gain factor calibration

The gain factor has to be calibrated only if

1. a light sensor connected via CEA-709 or BACnet is used and this sensor cannot be calibrated.

2. the constant light controller is operated in CONTROL mode to define the ratio between the measured outdoor lux level and the resulting indoor lux level.

To calibrate the gain factor perform the following steps:

1. Switch the light off and measure the current lux level at the reference area (e.g. desk) using a luxmeter.

2. Enter the measured lux level in the input field **Luxmeter**.

3. Enter the value measured by the light sensor in the field **Light sensor**.

4. Click on the **Calibrate** button.

7.6.4 Modify Sunblind Controller Event Priorities

To modify the priority of event which influence the sunblind controller click on the **Modify Sunblind Controller Event Priorities …** button. The dialog shown in Figure 158 is opened. For a description of the different events refer to Section 8.2.9.9.

The button **Higher** increases the priority of the selected event; the button **Lower** decreases the priority.

Events can be disabled with the **Remove** button. Previously deleted events appear in the drop-down list and can be enabled by pressing the **Add** button.

Per default an event which is activated with identical parameters multiple times in succession will be executed multiple times. For some events (e.g. local control) this default behavior can be changed by pressing the **toggle re-evaluation** button. The text “not re-evaluated” will appear beside the event and successive identical commands will be ignored.

To store the modified event priorities press the **Modify** button. The event priorities can be reset to the default values by pressing the **Reset to default** button.

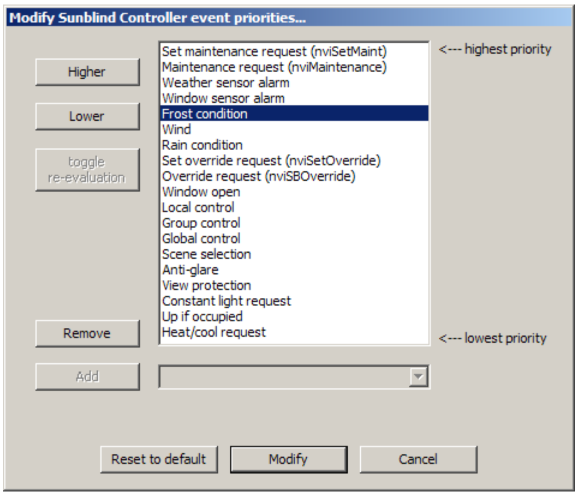


Figure 158: Modify Sunblind Event Priorities

7.6.5 Link Sunblind Controller to Constant Light Controller

To synchronize the lights and sunblinds of a room the sunblind controllers of a room have to be linked to the room's constant light controller. When clicking on the **Link Sunblind Controller to Constant Light Controller…** button the dialog shown in Figure 159 is opened. Select a constant light controller and press the **Link** button.

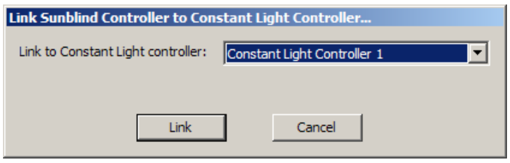


Figure 159: Link Sunblind Controller to Constant Light Controller.

7.6.6 Configure Emergency Light Auto-Test Calendar

This function is only available if the Configurator is connected to an L-DALI and one or more Lamp Actuator objects are selected, which are assigned to a DALI ballast of type emergency light. Further, the DALI network has to be commissioned by downloading the DALI configuration.

To configure the Auto-Test Calendar of a DALI emergency light click on **Configure Emergency Test Interval & Delay…**. The dialog shown Figure 160 in is opened.

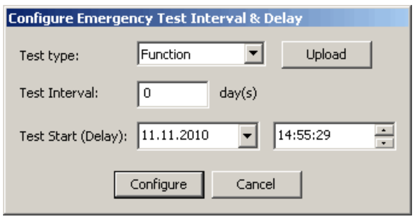


Figure 160: Configure Emergency Test Interval & Delay.

For both tests – function and duration test – a test interval in days and the time and date of the next execution of the test can be specified. A Test Interval of 0 disables auto-test of the selected test. Click **Configure** to store the new values in the selected devices. Click on **Upload** to read the currently configured values from the device.

Note: *The resolution of the duration test interval is 7 days, the resolution of the delay to the next test execution is 15 minutes. In both cases the value entered will be rounded to the next appropriate value.*

Test results will be stored in the appropriate emergency light test log (see Section 8.1.4).

7.7 Data Point Manager

The Configurator uses a central concept to manage data points. The data point manager is located on the **Datapoints** tab as shown in Figure 161. It is used to select, create, edit and delete user data points, calendar, scheduler, and trendlog objects, e-mail templates, alarm logs, and math objects. Further it allows selecting those data points on the device, which shall be exposed as OPC tags.

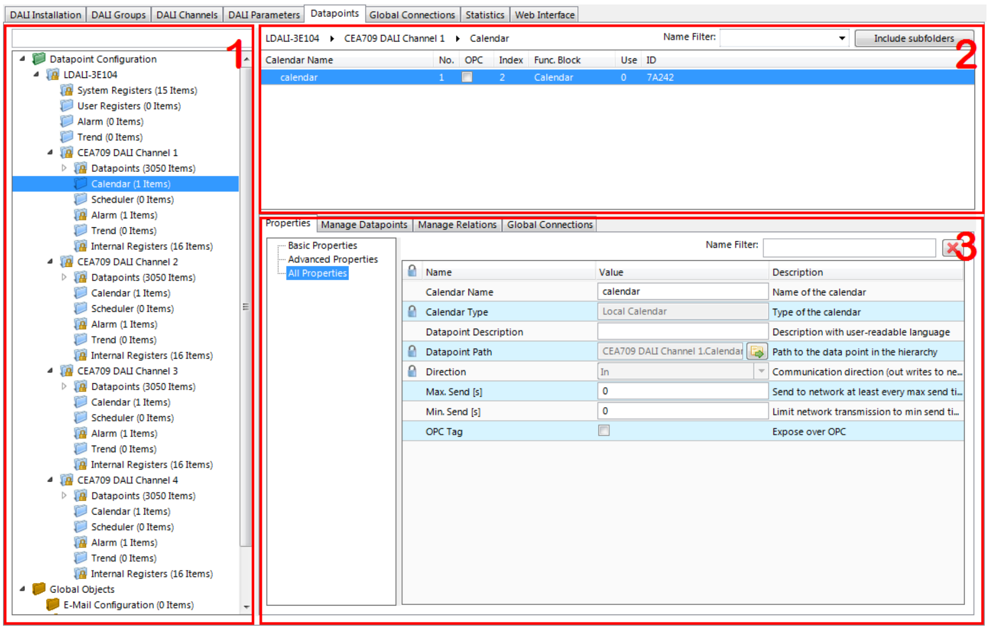


Figure 161: Data Point Manager Dialog.

The dialog is divided into three sections:

- The folder list (number 1 in Figure 161),
- The data point list (number 2 in Figure 161),
- And a property view (number 3 in Figure 161).

7.7.1 Folder List

At the left is a list of folders which is used to sort the available data objects by their category. There are a number of predefined folders available:

- **Imported**: This folder has a number of sub-folders for different import methods:
    - **BACnet Network Scan (LDALI-20X only)**: This folder is used to display data points retrieved by an online scan of the BACnet network.
    - **BACnet EDE File (LDALI-20X only)**: This folder is used to display data points imported from an EDE file.

Data objects in the import folder are not stored on the device when the project is downloaded. They represent data objects which are available on remote devices and are shown here as templates to create suitable data objects for use on the device by selecting the **Use on Device** option.

- **L-DALI**: This is the device folder of the L-DALI. It contains all the necessary data points which constitute to the L-DALI's port interface definition. These data points are created on the L-DALI when the configuration is downloaded. The subfolders represent
    - **Favorites**: This folder contains freely configurable symbolic links to data points, which may reside anywhere in the folder structure. This folder represents a way to assemble an alternate logical view to the data point hierarchy. This folder is also available on the Web UI or the LCD UI.
    - **System Registers**: This folder contains system registers, which provide information on the device itself.
    - **User Registers**: This folder holds user-definable registers. These registers are not visible on the underlying network and are intended for internal usage.
    - **Scheduler**: This folder contains generic scheduler and calendar objects. These provide technology-independent scheduling functionality. Any data point can be alarmed using a generic scheduler object.
    - **Alarm**: This folder contains generic alarm servers. These provide technology-independent alarms. Any data point can be alarmed using a generic alarm server.
    - **Trend**: This folder contains generic trend log objects. These can record historical values for any data points.
    - **CEA709 DALI Channel X (LDALI-10X only)**: These folders contain data points, schedulers, calendars, trend logs, and alarm servers of the CEA-709 network technology representing the different DALI channels. There is one subfolder of this type for each DALI channel available on the L-DALI device. See Section 7.7.2.
    - **BACnet Port (LDALI-20X only)**: This folder contains data points, schedulers, calendars, trend logs, statistics, and remote data points of the BACnet network technology. See Section 7.7.2.
- **Global Objects**: This top-level folder contains sub-folders that organize specific application objects that operate on data points.
    - **E-Mail Configuration**: This folder contains e-mail templates. An e-mail template defines the destination address and text body of an e-mail, which is triggered by data points and may contain data point values or file attachments. To create an e-mail template, select the folder and use the context menu (see Section 7.7.12).
    - **Math Objects Configuration**: This folder contains math objects. Math objects are used to perform a predefined calculation on a number of input data points and write the result to a defined set of output data points. Each math object contains one formula. To create a math object, select the folder and use the context menu (see Section 6.2).
    - **Alarm Log Configuration**: This folder contains the alarm log objects. Each alarm log object creates a historical log of alarm transitions of one or more alarm objects (alarm server or client). To create an alarm log, select the folder and use the context menu (see Section 7.12.4).

Using the context menu on a folder, sub-folders may be created to organize the available objects. If new objects are created automatically, they are usually placed in the base folder and can then be moved by the user to any of his sub-folders. Note, that the folder structure described above cannot be changed by adding or deleting folders at that level.

7.7.2 Network Port Folders

Each network port folder on the device has the same structure of sub folders. These sub folders are:

- **Datapoints**: This folder holds all data points, which are allocated on the network port. As the L-DALI comes with a static interface matching its fixed application most of the data points in this folder are protected. That is, new data points cannot be created and existing data points cannot be deleted or modified. If the model supports the creation of additional custom data points a separate **User** sub-folder is available to hold these custom data points.
- **Calendar**: This folder is used to hold a locally available calendar object with its calendar patterns (definitions of day classes like holiday, maintenance day, and so on). Current devices allow one local calendar object. To create a calendar, select the folder and use the context menu (see Section 7.10).
- **Scheduler**: This folder is used for local scheduler objects. Each of these objects will map to a local scheduler on the device’s network port. Configuring schedules through these objects actually transfers *schedule configuration data* to the underlying scheduler objects on the network port. To create a scheduler, select the folder and use the context menu (see Section 7.10).
- **Alarm**: This folder is used for local alarm server objects. Current devices allow one local alarm server, which is already pre-allocated and cannot be deleted.
- **Trend**: This folder is used for local trend log objects. Each of these objects will be able to trend data points over time and store a local trend log file. To create a trend log object, select the folder and use the context menu (see Section 7.13).
- **Statistics**: This folder contains registers, which provide communication statistics specific to the network port.
- **Remote Devices**: This folder is used to collect all remote calendars, schedulers, trend logs, and alarm client objects, which were created from network scan data. For each remote device, a subfolder will be created where the objects referencing this device are collected.

7.7.3 Data Point List

At the top right, a list of all data objects which are available in the selected folder is shown. From this list, objects may be selected (including multi-select) in order to modify some of their properties. Click on the **Include Subfolders** button to show all data points of the selected data point folder and all its sub-folders. This can be a convenient way for multi-select across folders.

To filter for data point names, enter a search text into the **Datapoint Name Filter** text box and hit *Enter*. A drop-down list holds the previously used filters available. Filters can also specify name patterns for sub-data points using a dot. Typing the first dot expands all filtered data points to the first sub-level. Continuing typing after the dot then filters on names of those sub-data points. For example, typing “sw.val” selects all data points having “sw” in their name, then expanding to their first sub-level and filtering all sub-data points on that level having “val” in the name. For complex name filters, regular expressions may be used (see Section 6.7).

The list can be sorted by clicking on one of the column headers. For example, clicking on the **Direction** column header will sort the list by direction. Other columns display **Datapoint Name**, **NV name**, **SNVT**, **Object Name**, object **Type** and **Instance** number. To apply the current sort order as the new data point order on the device, right-click on the column header and select **Renumber Datapoints**. Alternatively, select from the menu **Tools → Renumber Datapoints.**

The **OPC** column provides check boxes for each data point. If checked, the respective data point is exposed to OPC on the device. Deselect the check box, if a data points shall not be exposed to OPC. Note, that deselected data points do not add to the OPC tag limit.

The **Param** column provides check boxes for each data point. If checked, the respective data point is exposed to the parameter file.

New objects may be created in the selected folder by selecting **New Datapoint** command in the context menu of the data point list. A plus ⊞ sign in the list indicates that the data point contains sub-points. Clicking on the plus ⊞ sign expands the view.

If an entry in the data point list denotes a reference to another data point (e.g., a scheduled value reference beneath a scheduler object), right-click on that item and choose **Go to related data point** in the context menu. This will navigate your selection to the referenced data point.

For the alarming, scheduling, trending (AST) features, there are columns, which display icons for data points that are attached to an AST function. See Table 19 for details.

Icon	Data Point Usage
	Data point is scheduled
	Data point has an active alarm condition
	Data point has an inactive alarm condition.
	Data point has sub-data point with alarm condition.
	Data point is a trigger for e-mails.
	Data point used for trending.
	Data point is a link.
	Data point has sub-data points, which are links.

Table 19: Icons for used data points in the data point list view.

The data points in this list are color coded to make general information visible to the user at-a-glance. The color coding is described in Table 20.

Color	Coded Information
ao1 (yellow)	Data point is user-created and can be downloaded to the device.
MAC (blue)	Data point is fixed on the device and cannot be edited, e.g. system registers.
dark red	Data point is user-created and its technology object (e.g. dynamic NV) has been deleted. The data point, however, is still used in the configuration. It will not work on the device until the technology object is created again.

Table 20: Color coding used for data points in the data point list view.

7.7.4 Property View

When one or multiple data points are selected, the available properties are displayed in the property view. Properties which are read-only are marked with a lock  sign. When applying multi-select, only those properties common to all selected data points are displayed. According to the frequency of usage, different views for the properties are provided. **Basic Properties** list the most frequently used ones. **All Properties** is a list of all available properties for the data point. Depending on the network technology and data point class, different properties may exist.The **Name Filter** in the properties tab allows quick filtering of properties that have a matching text in their name. For example, type “OPC” to filter the OPC Tag property.

Data point properties common to all technologies:

- **Datapoint Name**: This is the technology-independent data point name. This name may be longer than and different to the name of the native communication object (i.e., network variable). Data point names must be unique within a given folder. The maximum length is limited to 64 ASCII characters.
- **Datapoint Path**: This informational property specifies the entire path of the data point within the data point hierarchy. The maximum length is limited to 64 ASCII characters.
- **Datapoint Description**: This is a human-readable description of the data point. There are no special restrictions for a description.
- **OPC Tag**: If enabled, the data point will be exposed to OPC. If not enabled, this data point does not contribute to the limit of OPC tags.
- **Parameter**: If enabled, the data point will be exposed to the parameter file. Those data points are visible to the LWEB-900 parameter view [5]. A parameter data point is also persistent. See Section 6.1.5.
- **Use Pollcycle value as**: For input data points, this property defines whether the input shall use a receive timeout or be constantly polling. See Section 6.1.2.
- **Poll on Startup**: For input data points this property defines, whether the data point shall be polled once at start-up. Poll-on-startup can be enabled independently of the poll cycle. See Section 6.1.2.
- **Pollcycle**: For input data points, this property defines the poll cycle in seconds. Set this property to 0 to disable polling. See Section 6.1.2.
- **Receive Timeout**: For input data points, this property defines the receive timeout in seconds. Set this property to 0 to disable polling. See Section 6.1.2.
- **Min Send**: For output data points, this property defines the min send time in seconds. See Section 6.1.2.
- **Max Send**: For output data points, this property defines the max send time in seconds. See Section 6.1.2.
- **Send-on-delta**: For output data points this property defines, if value updates shall be sent only once they meet the COV condition of the data point. For analog data points the analog COV increment is used. If not checked, updates are always transmitted according to min and max send times. See Section 6.1.7.
- **Use Linear Scaling**: If this property is enabled, the analog values are pre-scaled from the technology to the data point. This scaling is in addition to any technology-specific scaling factors. If enabled, the properties **Custom Scaling Factor** and **Custom Scaling Offset** accept the scaling factors. See Section 6.1.7.
- **Custom Scaling Factor, Custom Scaling Offset**: These properties only exist, if linear scaling is enabled. They apply to analog data points only. See Section 6.1.7.
- **Only notify on COV**: This property assists for binary and multi-state input data points. It defines, if a data point shall trigger an update only when the value changes or on every write. If this is enabled, consecutive writes with the same value do not trigger an update. If you want to convey every write, disable COV on the data point.
- **Persistent**: This property defines, if the last written value shall be stored as a persistent value. Persistent data points restore that value after a restart from the persistent storage. See Section 6.1.4.
- **Default Value**: This property defines a default value (see Section 6.1.3). Enter a default value to enable this feature in the data point. Delete the value entirely to remove

the default value. If no default value is defined, this property reads “N/A”. The default is no default value.

- **Historic Filter**: This property allows defining historic filters for the scalar data point. See Section 6.4.6.
- **Point Type**: This is the base data point type, e.g., “Analog Datapoint”.
- **Direction**: This is the data point direction. Use input, output or value as directions.
- **Unit**: For analog data points this property contains the definition of an engineering unit of the scalar value, e.g., “kg”. A human-readable text for the engineering unit is displayed and can be entered. If the unit is known by the Configurator as a convertible unit, it is displayed with a green checkmark (see Section 6.1.13).
- **Unit for display**: If the data point has a convertible unit, an alternative unit for display can be chosen. The data point value on the Web UI is converted to and displayed in the display unit (e.g., °F instead of °C). For a non-convertible unit this option is not available.
- **Analog Datapoint Max Value**: For analog data points this property contains the upper limit of the supported value range. Note that this does not define an alarm limit.
- **Analog Datapoint Min Value**: For analog data points this property contains the lower limit of the supported value range. Note that this does not define an alarm limit.
- **Analog Datapoint Precision**: For analog data points this property defines the number of decimals. ‘0’ specifies an integer value. Display units may use this to format the floating point value accordingly.
- **Analog Datapoint Resolution**: For analog data points this property defines the smallest possible value increment.
- **Analog Point COV Increment**: This property is valid for analog input data points. It specifies by which amount the value needs to change, before an update is generated. If every write shall generate an update even when the value does not change, specify 0 as the COV increment. If any value change shall generate an update, delete the value, which results in **Any**.
- **Active Text**: For binary data points this property defines a human-readable text for the active state (true).
- **Inactive Text**: For binary data points this property defines a human-readable text for the inactive state (false).
- **Current State Map**: For multi-state data points this property defines the multi-state map. It must be set to a valid multi-state map or it points to User/UndefinedStates. Click on to assign a state map.
- **State Count**: For multi-state data points this property defines the number of discrete states.
- **State Text**: For multi-state data points this property defines a human-readable state label for each state.

7.7.5 Tracking Data Point Usage

Data points can be used by other objects, such as connections, schedulers, math objects and many more. To get an overview on data point usage, the data point list contains the column Use. This column provides a usage counter of the individual data points. If usage is ‘0’ the data point is not used by any other object.

If the usage count is larger than zero, choose the item **Show Usage …** from the data point context menu. This opens up a report window showing the objects referring to the selected data point. An example is shown in Figure 162.

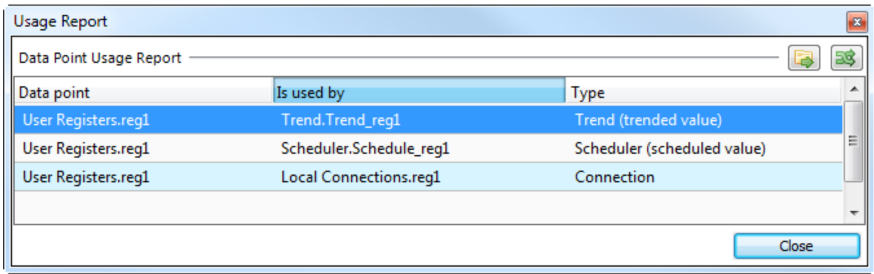


Figure 162: Data point usage report.

Each line reveals an object using the data point. Select a line and click on the **Go to data point** button. This will navigate yours selection to the reported object.

7.7.6 Managing Multistate Maps

Multistate data points have a descriptive set of state texts for their state IDs. To manage those state IDs and state texts among many multistate data points, they refer to *multistate maps*. Some technologies have a fixed set of such multistate maps others have freely configurable multistate maps (e.g, user registers). Editing a multistate map affects all multistate data points, which are using that particular map. It is not necessary to edit each data point individually. For managing multistate maps go to the menu **Tools → Manage State Maps …** .

To Edit a Multistate Map

1. Click on the **Configure** button in the **Current State Map** property of a multistate data point. This opens the multistate map manager as shown in Figure 163.

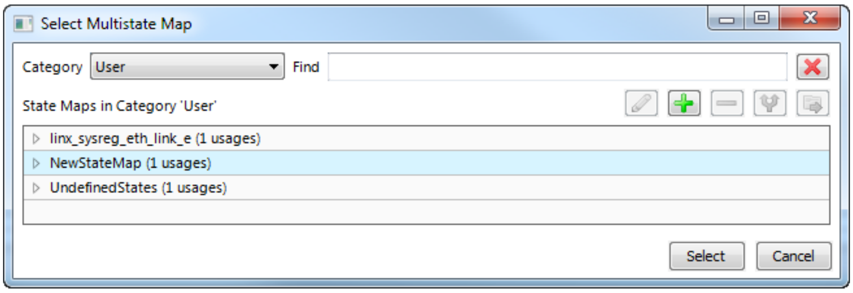
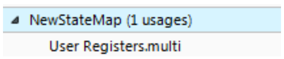


Figure 163: Select multistate maps in the multistate map manager.

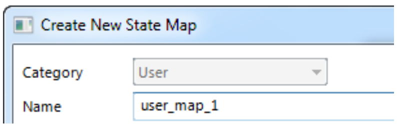
2. Select a **Category** and an existing state map in the **State Maps** list and click on **Select**. Maps that are fixed and cannot be changed are marked with a lock symbol.

3. Expand a state map line to see where this state map is used. Select the usage and click the **Go to Data Point** button. This navigates to the data point.

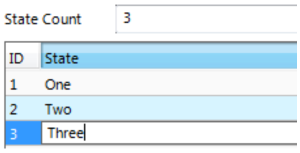


4. If a new multistate map shall be created, click the **Create State Map** button.

5. In the **Create New State Map** dialog enter a state map Name.



6. Then enter the desired number of states and edit the state texts as needed and click into the list of states. Edit state IDs and texts to your needs. Pressing Enter goes to the next line. Finally click the **Save** button.

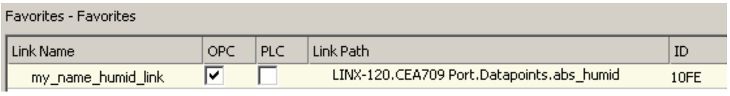


7. Select the newly created multistate map and click the **Select** button. The selected map is now assigned to the data point.

7.7.7 Organizing Favorites

There is a special **Favorites** top-level folder in the device data point folder hierarchy. This folder contains freely configurable symbolic links to data points, which may actually reside anywhere in the folder structure. This folder represents a way to assemble an alternate logical view to the data point hierarchy.

To configure favorites, select any data point from any location in the data point folder hierarchy and drag it onto the favorites folder. This will create a data point link, which is displayed in the data point list:



The **link name** can be edited to something different that the original data point name. The contents of this folder are also available on the Web UI or the LCD UI. The link names are displayed there. The data point links can also be individually exposed to the OPC notwithstanding if their original source is exposed or not.

Furthermore, the user can create sub-folders in the favorites folder and beneath to build a hierarchy of data point links. Folder links are, however, not allowed.

For editing links of favorites for a large number of data points, the **Manage Favorites** tab in the property view area of the data point manager provides a fast solution. The data point tree is replicated in that tab as shown in Figure 164. Select a folder and optionally click on the button  to include favorites in sub-folders. Enter a filter expression to **Filter**, which applies to the favorite name. For example, enter ‘room1’ to display all favorites that contain this as a sub-string.

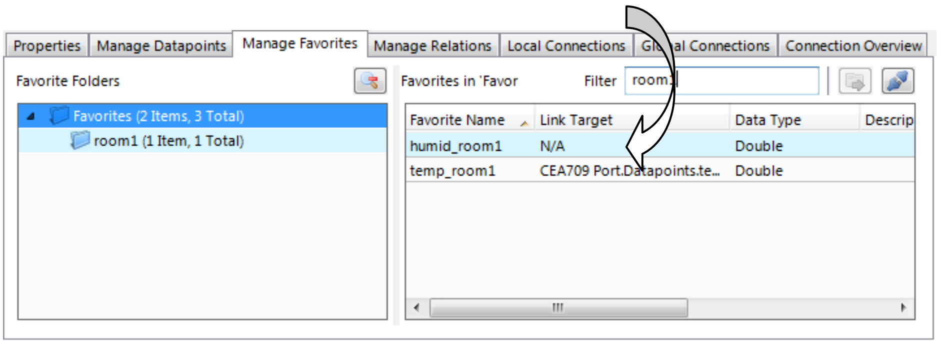


Figure 164: Manage favorites tab.

To link favorites to other data points using the manage favorites tab, navigate to the desired folder in the main folder tree of the data point manager and select the data point to be linked. Drag the data point onto the **Link Target** column. For detaching links, use multi-select on the desired favorites and click the detach button . On linked favorites you may click on the button , which jumps to the linked data point in the data point list.

7.7.8 Managing Property Relations

Property relations can be linked to data points in various user dialogs (e.g. alarm condition dialog). For editing links of property relations for a large number of data points, the **Manage Relations** tab in the property view area of the data point manager provides a fast solution. The data point tree is replicated in that tab as shown in Figure 165. Select a folder and optionally click on the button to include property relations of data points in sub-folders. Enter a filter expression to **Filter**, which applies to both the data point name and relation type. For example, enter 'feedback' to display all feedback value property relations.

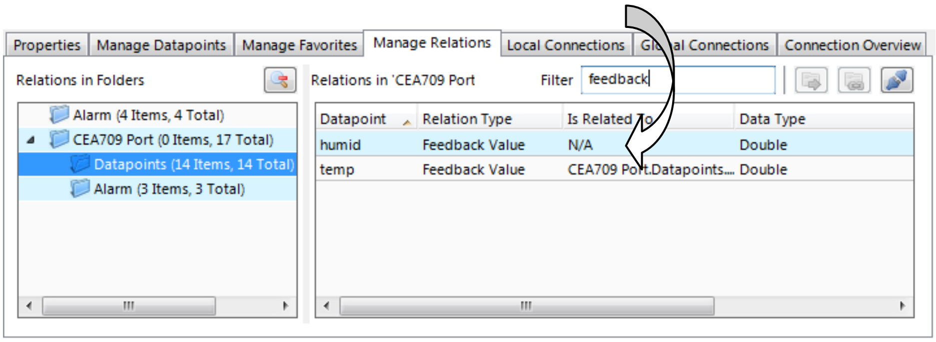


Figure 165: Manage relations tab.

To link property relations to other data points using the manage relations tab, navigate to the desired folder in the main folder tree of the data point manager and select the data point to be linked. Drag the data point onto the **Is related To** column. For detaching links, use multi-select on the desired property relations and click the detach button . On linked property relations you may click on the button , which jumps to the linked data point in the data point list.

7.7.9 CEA-709 Properties

Apart from the common data point properties discussed in Section 7.7.3 the data points of the CEA-709 technology have additional properties. Depending on whether a NV is local or external (remote), the properties may vary.

- **NV Allocation**: This property defines how a data point shall be allocated on the device. Choices are "Static NV", "Dynamic NV", and "External NV". If the allocation type cannot be changed, this property is locked.
- **SNVT**: This property defines the SNVT of the NV, e.g., "lux (79)".
- **Invalid Value**: This property defines the "invalid value" for the NV. If set, this specific value will be interpreted as "invalid" in the data point. If known by the SNVT, the invalid value is filled in. Otherwise, the user can specify an invalid value.
- **CEA-709 Mapping Information**: This information is derived from the SNVT. It defines how the NV contents are mapped to the data point.
- **NV Scaling A, B, C**: These are the scaling factors known from the SNVT table. The scaling factors are applied to translate a raw NV value into the scalar representation of the data point.
- **Data Type**: This is the basic NV data type. This is usually filled in from the SNVT definition.
- **Local NV Member Index**: This property specifies the NV member index within a given functional block. This must be a unique index in the functional block, which identifies the NV after other NVs have been added or removed from the interface.

- **Local/Remote NV Index**: This property specifies the NV index. For local, static NVs this is the NV index of the static NV. For external NVs, this is the NV index of the NV on the remote device.
- **Local/Remote NV Name**: This property specifies the programmatic name of the NV. For local, static NVs this is the programmatic name of the static NV. For external NVs, this is the programmatic name of the NV on the remote device.
- **Local/Remote Functional Block**: This property specifies the programmatic name of the NV. For local, static NVs, one of the reserved functional blocks can be selected.
- **Local/Remote NV Flags**: This property specifies the NV flags. For local (static or dynamic) NVs, the flags can be configured. For external NVs, these flags are only informational.
- **Remove NV Information**: For external NVs, this property contains the information on the remote device and the NV selector on that device.
- **Remote Device ID**: For external NVs, this property contains information on the remote device by listing the program ID and location string.
- **Remote Device Address**: For external NVs, this property contains the CEA-709 network addressing information to access the node, i.e., subnet, node, and NID.
- **Retry Count**: For external NVs, this property defines the retry count. The default is 3.
- **Repeat Timer**: For external NVs, this property defines the repeat timer in milliseconds. The default is 96 ms.
- **Transmit Timer**: For external NVs, this property defines the transmit timer in milliseconds. The default is 768 ms.
- **LNS Network Path**: If available from an LNS scan, this property specifies the LNS network path of the device where the given NV exists.
- **LNS Channel Name**: If available from an LNS scan, this property specifies the LNS channel name of the device where the given NV exists.

7.7.10 BACnet Properties

Apart from the common data point properties discussed in Section 7.7.3 the data points of the BACnet technology have additional properties. Depending on whether a server object and/or a client mapping exists, the properties may vary.

- **Engineering Units**: For analog BACnet server objects, this property defines the engineering units from the BACnet standard. One of those units can be chosen from a drop-down box, if this property is not locked.
- **Server Object Type**: This property defines the BACnet object type of the underlying BACnet server object. It can be changed within the class, i.e., for an analog data point, the server object type analog input, analog output, or analog value can be chosen.
- **Commandable**: This property defines, if the underlying BACnet server object is commandable. For BACnet value objects (AV, BV, MSV) this property can be edited to create commandable or non-commandable BACnet value objects.
- **Relinquish to invalid value**: This property defines whether the data point maintains the Relinquish_Default value, if the priority array is empty or assumes the invalid value. By default, this property is false and the Relinquish_Default value is used. Setting this property to true can be beneficial when used in a connection to withdraw a value in another technology.
- **Server Object Name**: This property defines the object name of the underlying BACnet server object. It must be unique among all server objects. It can be up to 64 characters.

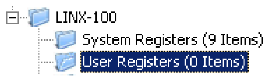
- **Server Object Instance No**: This property defines the object instance number of the underlying BACnet server object.
- **Server Object Description**: This property defines the object description of the underlying BACnet server object. It can be left blank.
- **Server Object Device Type**: This property defines the object device type of the underlying BACnet server object. It can be left blank.
- **Server Accumulator Offset Correction**: This property is available for accumulator objects only. It is activated as a default. It ensures that the written data point value is corrected in the Present_Value as soon as the Value_Set property is written and requests a certain Present_Value. That correction offset is then added to the written data point value, e.g., when received over a connection.
- **Get Active Priority**: Set this check box to let the data point reflect the active priority of the local or remote BACnet object. The priority is a number between 1 and 16. This property is only applicable for commandable BACnet objects.
- **Allocate Server Object**: This Boolean property defines, if a server object shall be allocated for the data point. This option is useful, when a local server object shall be allocated for a client mapping.
- **Allocate Client Mapping**: This Boolean property defines, if a client mapping shall be allocated for the data point. This option is always set, if at least on client mapping is attached.
- **Client Map Count**: This property defines the number of client mappings attached to a data point. A data point can have one read client map or *n* write client mappings.
- **Client Map [n]**: This is a list of client mappings. The property shows a summary of the client mapping parameters.
- **Client Confirmed COV**: This Boolean property defines, if a client map subscribes with the confirmed COV service. If not enabled, the unconfirmed COV is used.
- **Client Map Type**: This property determines the type. It can be one of the following: Poll, COV, Auto, Write, or Value (see Section 6.6.1).
- **Client Write Priority**: For a write or value client map, this property defines which priority is used for writing.
- **Remote Instance Number**: This property specifies the object instance number of the remote server object. The object type cannot be edited.
- **Value Read Mode**: For value client mappings, this property defines the read mode: Poll, COV, or Auto.

7.7.11 User Registers

User registers are data points on the device that do not have a representation on the network. Thus, they are not accessible over a specific technology. A register merely serves as a container for intermediate data (e.g., results of math objects). Since a register has no network direction, it can be written and read. Therefore, it is created as a value data point. For back-ward compatibility, it is still possible to generate two data points for each register, one for writing the register (output) and one for reading the register (input).

To Create a User Register

1. Select the **User Registers** folder under the device folder.



2. Right-click in the data point list and select **New Datapoint…** in the context menu. This opens the register creation dialog as shown in Figure 166.

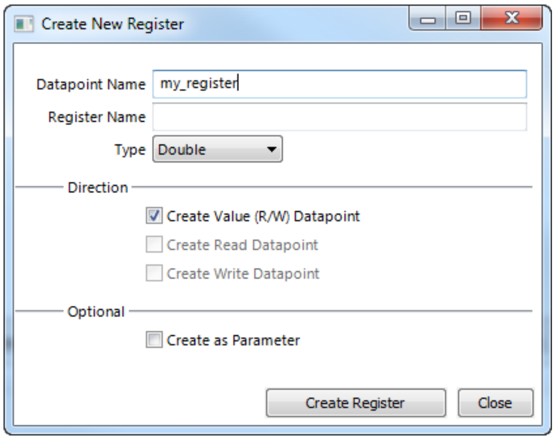


Figure 166: Create a user register.

3. Enter a **Datapoint Name** for the register. You may leave the **Register Name** blank to give the underlying register the same name as the data point.

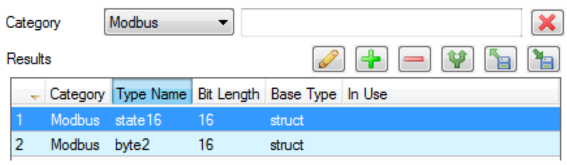
4. The standard setting is a value data point for read/write. Optionally, deselect the value option and select the read data point or write data point check box. This is necessary when configuring for an older firmware version.

5. Select **Create as Parameter** if needed. In this case, the register will be a persistent parameter. It can be done later in the data point properties also.

6. Select a **Type**. Available are “Double”, “Boolean”, “Signed Integer”, “String”, “Variant” or “User”.

7. When selecting **User**, a register with a user-defined structure is going to be created. Click on ... next to **Custom Type** in order to select a structure type.

8. In the dialog **Select Custom Type** choose the type **Category** in the drop-down list and the desired user type. Then click the button **Select**.



9. Click **Create Register**.

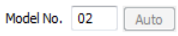
10. If needed create additional registers from the dialog. Finally, click **Cancel** to exit the dialog.

7.7.12 Build XIF for CEA-709 Interface

When using a non-standard static CEA-709 interface on the device, the Configurator can export a new XIF file for the changed static interface. Before exporting the XIF for the interface it is recommended to download the configuration into the device. In this case, the Configurator can verify that the model number of the port is correct.

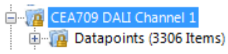
To Create a XIF File

1. Make sure the **Model No** will match the final model number of the port. If not, enter an appropriate model number in the toolbar of the **Datapoints** tab.



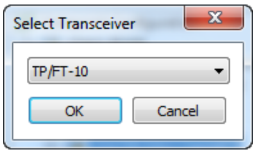
   A custom model number must be in the range 2 to 255 (0 and 1 are used by the standard and the emergency interface respectively).

2. Select the **CEA-709 Port** folder



3. Right-click on that folder and in the context menu select **Build XIF …**.

4. This opens a dialog, which allows selecting, which transceiver (FT-10 or IP) is used.



5. Selecting the appropriate transceiver and clicking on **OK** opens a file requestor where the XIF file name needs to be entered. Select a useful name to identify the device, e.g., as "LDALI-10X_button_02.xif".

7.8 BACnet Configuration (LDALI-20X only)

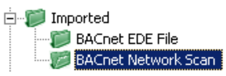
7.8.1 Scan for BACnet Objects

LOYTEC devices also support an online network scan on the BACnet network. In this scan the device searches for other devices on the BACnet network and pulls in the BACnet object information of these devices. These BACnet objects can then be used on the device as the basis for client mapping.

To Scan for BACnet Objects

1. Go to the **Datapoints** tab.

2. Select the folder **BACnet Network Scan**



3. Right-click on that folder and select **Scan BACnet Network…**. This opens the BACnet Network Scan dialog as shown in Figure 167.

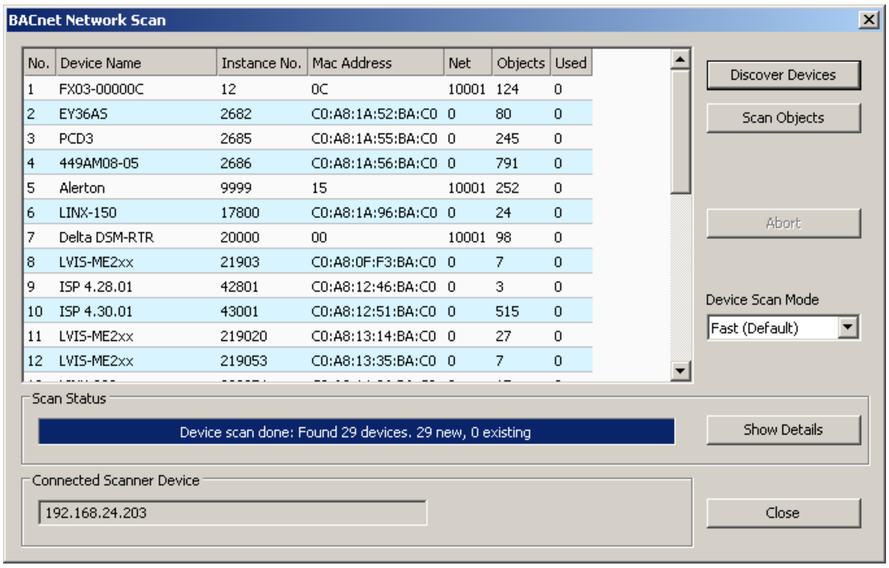


Figure 167: BACnet network scan dialog.

4. Click on the button **Discover Devices**. This starts a network scan. The results are put in the device list box. A progress bar below indicates how many devices are being scanned.

5. Select a device in the device list and click the button **Scan Objects**. This scans the BACnet objects on the selected device and adds them to the **BACnet Network Scan** folder as a separate sub-folder for the device.

6. If the scan does not give the expected results, change the **Device Scan Mode** to normal or slow and try again. With this setting the scanner uses simpler but slower protocol features.

7. Click **Close** when all devices needed have been scanned.

Note: *If proprietary properties access on a remote device is required, support for proprietary properties must be enabled in the BACnet tab of the Project Settings dialog (see Section 7.16.5).*

7.8.2 Import from EDE File

If the device is engineered offline or some of the required BACnet devices are not yet online in the network, the engineering process can be done by importing a device and object list from a set of EDE files. These objects also appear in the import folder and can be later used on the device as client mappings.

There are a set of EDE files. Select the main EDE file, e.g. *device.csv*. The EDE import will also search for the other components, which must be named *device-states.csv*. Which components are expected, please refer to Section 8.3.9. Example EDE files can be found in the ‘examples’ directory of the LOYTEC Configurator software installation directory.

To Import BACnet Objects from an EDE File

1. Go to the **Datapoints** tab.

2. Select the folder **BACnet EDE File**

3. Right-click and select **Import File**. In the following file selector dialog, choose the EDE import file and click **OK**.

4. Now the **BACnet EDE File** folder is populated with the imported BACnet objects.

7.8.3 Use Imported BACnet Objects

After BACnet objects have been imported (with a network scan or by importing from an EDE file) the user can select the BACnet objects that the device shall access. When executing the **Use on device** the configuration software allocates client mappings on the device. These client mappings will read or write values from the BACnet objects in the network.

In an additional step, there can be also server objects allocated on the device. These server objects can be created automatically from converting a client mapping to a server object. This is usually done, if the imported BACnet objects shall also be directly modified over the BACnet network on the device itself.

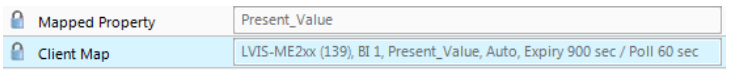
To Use Imported BACnet Objects on the Device

1. Open the data point manager dialog and select the desired BACnet objects in one of the import folders.

2. Use the multi-select feature by holding the *Shift* or *Ctrl* keys pressed.

3. Click on the button **Use on Device** in the tool bar.

4. This creates data points in a remote device sub-folder of the BACnet Port/Datapoints folder. All data points in that folder will be created as client mappings. No server object is created automatically in this case.



5. To also create server objects select the data points in question using the multi-select feature. Then activate the property **Allocate Server Object** in the section **Advanced**.

6. For editing the client mapping, you may multi-select client map data points and edit the corresponding data point properties **Client Confirmed COV**, **Client COV Expiry**, **Client Map Type**, **Client Write Priority**, **Remote Instance Number**.

7.8.4 Create a Client Mapping

The client mapping information can also be created manually. Usually, this is done to create client data points without importing information from EDE or scanning online.

To Create a Client Mapping

1. Select the **Datapoints** folder under the **BACnet Port** folder.

2. Right-click in the data point list and select **New Datapoint…** in the context menu. This opens the **Create New BACnet Point** dialog as shown in Figure 168.

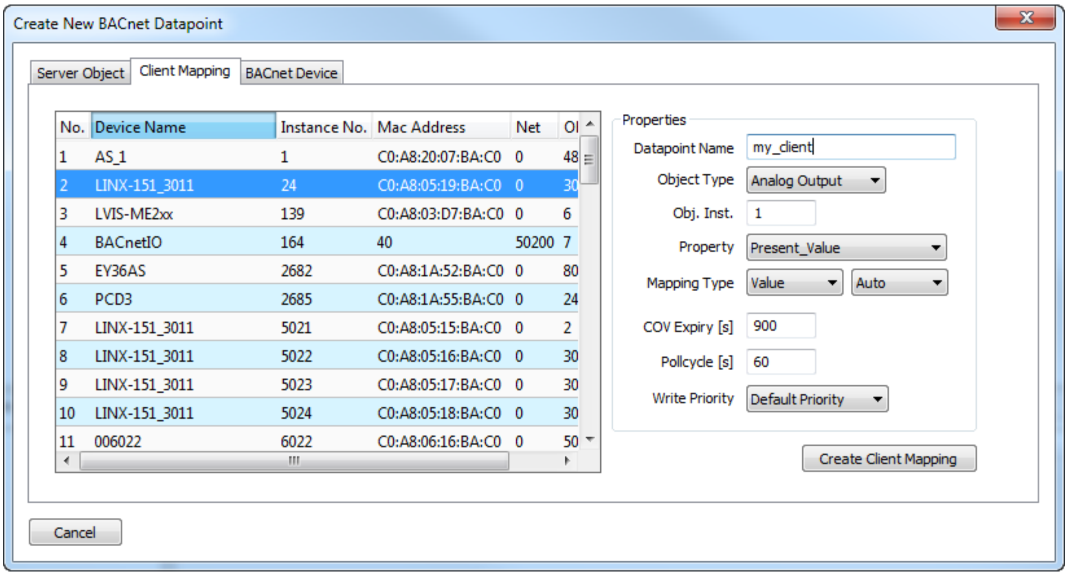


Figure 168: Create Client Mapping Dialog.

3. Select the tab **Client Mapping**.

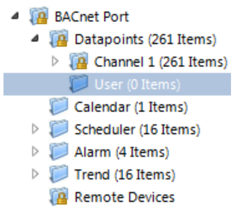
4. Choose a target device in the list of known devices. Enter a **Data Point Name**, choose an **Object Type**, and edit the target object instance number. Then select the **Mapping Type**. For read client mappings edit the **COV expiry** or **Pollcycle** setting. For write client maps edit the **Write Priority**. For value client maps edit both. When finished click **Create Client Mapping**.

7.8.5 Create Server Object

On the BACnet port server objects can also be created manually. These BACnet objects are visible on the BACnet network and can be modified by other devices. They appear as data points in the **BACnet/Datapoints** folder.

To Create Server Objects Manually

1. Select the **User** sub-folder of the **Datapoints** folder under the **BACnet Port** folder.



2. Right-click in the data point list and select **New Datapoint…** in the context menu. This opens the **Create New BACnet Point** dialog as shown in Figure 169.

3. In the **Mandatory Properties** enter a **Datapoint Name** and an **Object Type**. Optionally, update the **Instance No** and select the **Commandable** check box for value objects, if the value object shall be commandable from the network.

4. In the **Optional Properties** you may select **Engineering Units** for analog objects. For all object types you can enter the **Description**. The **Device Type** can be left empty. For multi-state objects you have to select a multi-state map.

5. Click **Create Server Object**. The BACnet data point is created and appears in the data point list.

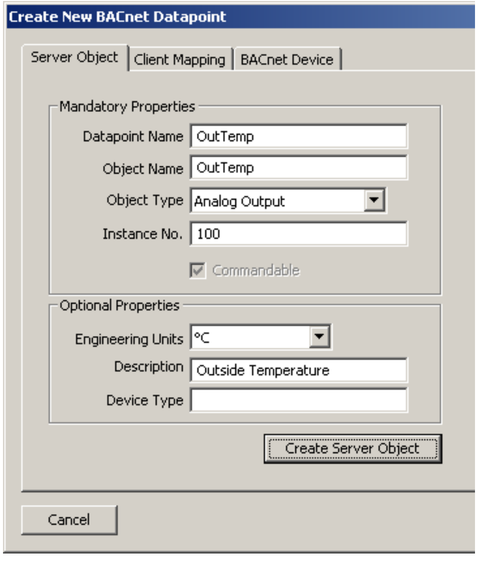


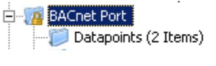
Figure 169: Create a Server Object manually.

7.8.6 Export Server Objects to an EDE File

When engineering offline it can be beneficial to hand out the server object configuration of the device to other parties electronically. For doing so you may export the server object configuration to a set of EDE files. The set of EDE files consist of the main EDE file, e.g. *myDevice.csv*. This file contains the list of all objects and refers to state texts that are exported to a second file named *myDevice-states.csv*. For which components are exported in an EDE file, please refer to Section 8.3.9.

To Export an EDE File

1. Select the **BACnet Port** folder.



2. Right-click and select **Export EDE …** in the context menu. This opens the EDE export dialog to enter the EDE header information as shown in Figure 170.

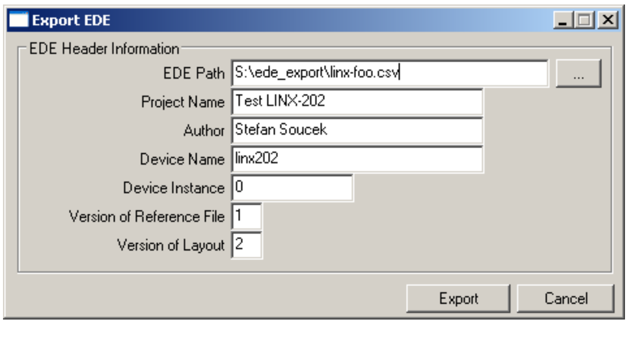


Figure 170: EDE Export Dialog.

3. Click the  button to select the EDE file export location.

4. Specify the **Device Name** and **Device Instance**. The device instance will be used by other tools to configure their BACnet clients for accessing the exported device.

5. Optionally fill in project name, author to document that information in the EDE file.

6. Click **Export**.

7.8.7 Import Server Objects from an EDE File

It is also possible to import a BACnet server object interface from EDE files. In this use case, the device is configured to resemble a the device of the EDE file. If conflicts in instance numbers or object names arise with already existing server objects, the imported objects are re-assigned.

There are a set of EDE files. Select the main EDE file, e.g. *device.csv*. The EDE import will also search for the other components, which must be named *device-states.csv*. Which components are expected, please refer to Section 8.3.9. Example EDE files can be found in the 'examples' directory of the LOYTEC Configurator software installation directory.

To Import BACnet Server Objects from an EDE File

1. Select the folder **BACnet Port**

2. Right-click and select **Import Server Objects from EDE…**. In the following file selector dialog, choose the EDE import file and click **OK**.

3. Now a folder for the device in the EDE file is generated and a report is displayed, informing about the imported objects and possible reassignments.

7.8.8 Map other Properties than Present_Value

When creating a BACnet server object, the Present_Value property is mapped by the created data point. That means writing and reading on the data point reads or writes the Present_Value. If other properties shall be accessed, they must be added to the BACnet server object's data point.

To Add other BACnet Properties

1. Select the BACnet server object for adding properties.

2. Right-click on the data point and select **Add/Remove BACnet properties …** from the context menu. The dialog appears as shown in Figure 171.

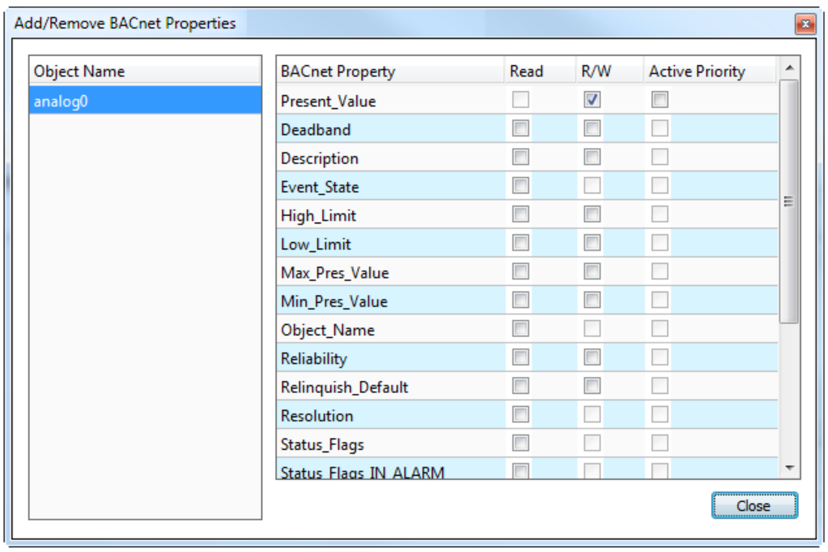
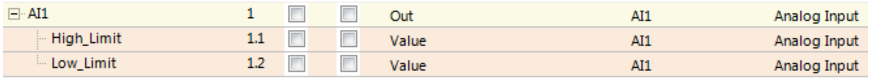


Figure 171: Dialog for adding/removing BACnet properties.

3. Check the additional properties. Checking the **Read** box will add an input data point, checking the **R/W** box will add a value data point.

4. Click **Close**. The selected data point can now be expanded with the plus icon and show its additional properties as sub-data points.



5. To remove properties perform the same steps and uncheck the corresponding check-boxes. Alternatively, select the property (or more) and press the *Delete* key.

7.8.9 Enable International Character Support

By default BACnet objects on the device contain ASCII strings in properties such as object name, description, active/inactive text, state texts. This is the setting most third-party tools are interoperable with. To support international character sets, the device can be configured to expose strings as ISO-8895-1 (for most Western European languages) or UCS-2 (for Unicode character sets such as Japanese).

To Enable International Character Support

1. In the Configurator software menu go to **Settings → Project settings …**. This opens the **Project Settings** dialog (see also Section 7.16.5).

2. Click on the tab **BACnet**.

3. Put a check mark either on **ASCII/UTF-8** (default), **UCS-2** (Unicode, e.g., for Japanese), or **ISO-8859-1** (for Western European languages).

4. Click **OK.**

5. Download the configuration to activate the change.

7.8.10 Read the Active Priority

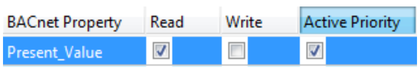
In BACnet objects that possess a priority array the effective value depends on the used write priority slots. A special data point on the device can be added that allows reading out the active priority of such an object, giving a value between 1 and 16.

To Read the Active Priority of a Local Object

1. Select the BACnet server object for adding properties.

2. Right-click on the data point and select **Add/Remove BACnet properties …** from the context menu. The dialog for mapping BACnet properties to data points appears.

3. For the Present_Value select the additional box Active Priority.



To Read the Active Priority of a Feedback Value

1. Select a write client mapping.

2. Right-click on the data point and select **Create Priority Feedback Point** from the context menu.

3. A new data point is created, which is a feedback client mapping that reads the active priority out of the remote object.

7.8.11 Write and Read with Priority

In BACnet objects that possess a priority array the effective value depends on the used write priority slots. As default, data points for those objects have the direction input, which is intended to read the resulting value. Additional priority write output data points can be added for writing to the BACnet object. For those data points, a write priority between 1 and 16 can be defined. The default write priority is defined by the project settings.

To read back the value at a given priority slot, additional priority read input data points can be added. They reflect the value at the configured priority slot between 1 and 16. If the priority slot is NULL the data point stays at invalid value.

To Create a Priority Write Data Point

1. Select the commandable BACnet server object.

2. Right-click and select **Create Priority Write Point…** from the context menu.

3. A dialog prompts for the write priority. Note that the write priority can also be changed later.

4. The new priority write data point appears below the original BACnet server object data point.

Datapoint Name	No.	OPC	Param	PLC in	PLC out	Direction
AO1	1	☑	☐	☐	☐	Value
AO1_pri4_Write	2	☑	☐	☐	☐	Out

To Create a Priority Read Data Point

1. Select the commandable BACnet server object.

2. Right-click and select **Create Priority Read Point…** from the context menu.

3. A dialog prompts for the read priority. Note that the read priority can also be changed later.

4. The new priority read data point appears below the original BACnet server object data point.

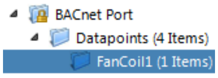
Datapoint Name	No.	OPC	Param	PLC in	PLC out	Direction
AO1	1	☑	☐	☐	☐	Value
AO1_pri4_Write	2	☑	☐	☐	☐	Out
AO1_pri4_Read	3	☑	☐	☐		In

7.8.12 Duplicate BACnet Devices with Data Points

When importing BACnet devices via network scan or EDE file, the resulting client mappings are used on the device. For each BACnet device a sub-folder is created which organizes the client mapping data points for that device. The BACnet device itself appears in the BACnet device manager. After editing the client mapping data points as appropriate, entire device folders can serve as templates for duplication. The created copies of the data points are pointing to an unassigned device, which can be commissioned later on the Web interface.

To Duplicate BACnet Devices

1. Select a folder created for a scanned/imported BACnet device.



2. Right-click and choose **Duplicate** in the context menu.

3. The **Duplicate data points and set naming rules** dialog opens as shown in Figure 172.

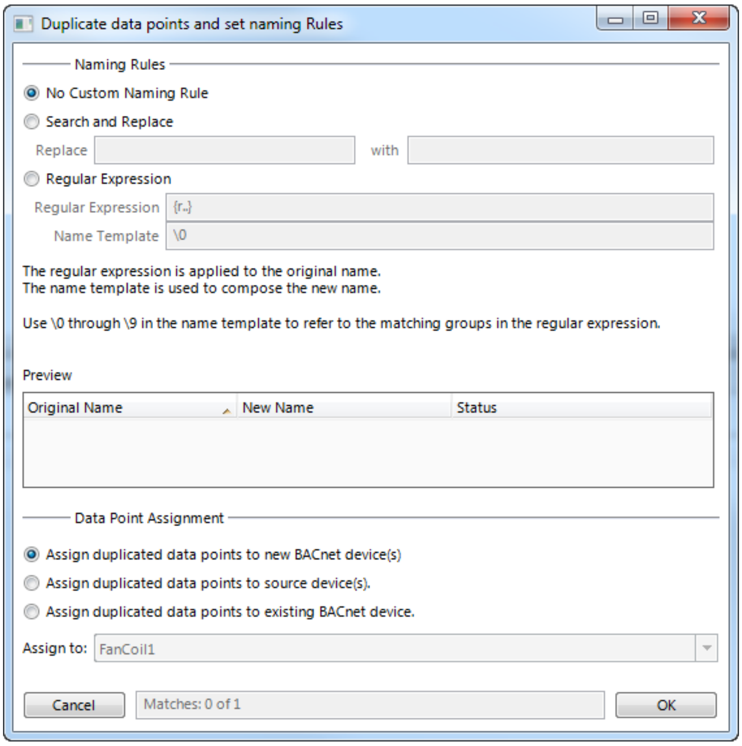


Figure 172: Duplicate BACnet devices.

4. In the **Data Point Assignment** section choose to assign the duplicated data points to a new BACnet device instance and click **OK**.

5. A new BACnet device folder is created with the duplicated client mapping data points. The original BACnet device is also duplicated, leaving the actual device instance number empty and marking the device to be commissioned later on the Web interface as described in Section 5.3.1.

6. The **BACnet Device Manager** shows the created devices as depicted in Figure 173.

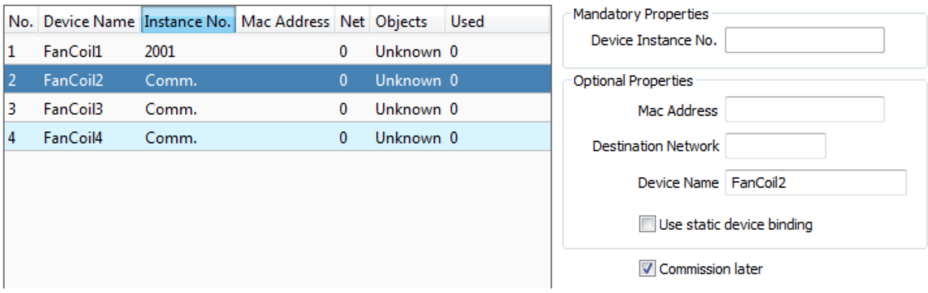


Figure 173: Duplicated BACnet devices for later commissioning.

7.9 Connections

7.9.1 Create a New Connection

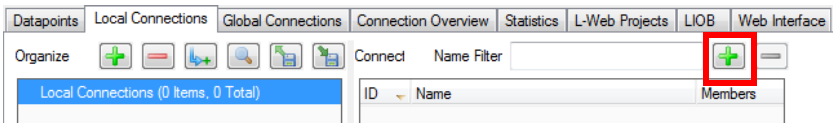
After having configured the device's network ports with data points, internal connections between those data points can be created. Usually, the manual method to create a connection is used to create connections between different named data points.

A connection is an internal mapping in the device between input and output data points. A connection always consists of one or more data points. A value update from an input data point (sender) is distributed to all output data points (receivers). A status change of a receiver data point is propagated back to all sender data points. All data points in the connection must be of a compatible type or use an adaptor.

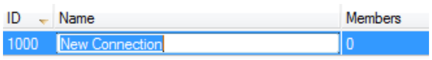
By adding data points as sender and receiver to the same connection, they transfer values in both directions. Doing so with connected data points, bi-directional connections can be built.

To manually create a new connection

5. Click on the **Local Connections** tab



in the main window and press the speed button **Create new Connection**. In the menu choose **Standard Connection**. A new connection is added to the connection list. Rename the connection if you want to do so.



6. Over the list **Datapoints in connection** on the right-hand side click on **Attach Data Point** to add data points for this connection. This opens a list of all available data points. Select one and press **OK**. You may use multi-select to select more than one data point at a time.

Note: *By default only compatible data points are displayed. Sometimes compatible data points are available as member points (e.g., a SNVT structure member). Click on ⊞ to expand the data point and select the desired member point.*

7. Now the connection tab contains the new connection and below the list of data points in that connection as shown in Figure 174.

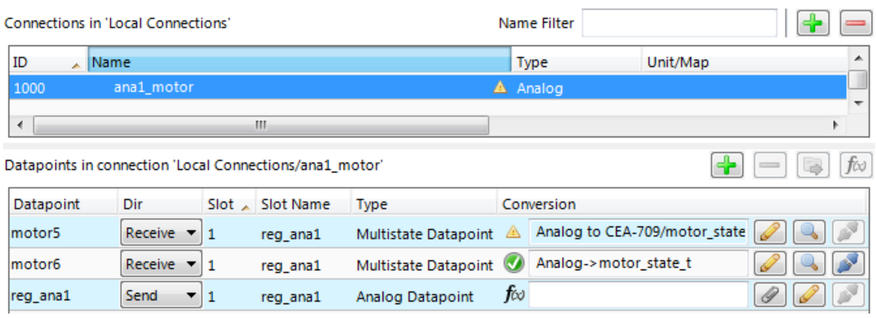


Figure 174: Connection tab with a connection and data points.

8. Change the direction by modifying **Send** or **Receive**. For changing multiple data points use multi-select. Optionally, select **Disable** to temporarily exclude this data point from communication in the connection.

9. If the attached data point needs a conversion, the item displays a yellow exclamation mark and the default conversion (e.g. 'Analog to CEA-709/motor_state_t'). By clicking on the button to view the current conversion.

10. To add a new conversion to this item, click on the button. A dialog opens, which displays the matching adaptors already available in the library as shown in Figure 175.

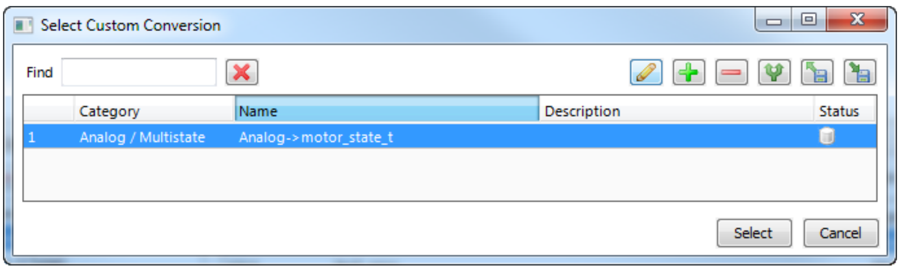


Figure 175: Choose a custom conversion.

11. Select an existing conversion, click the plus button to create a new conversion, or click the edit button to modify an existing conversion.

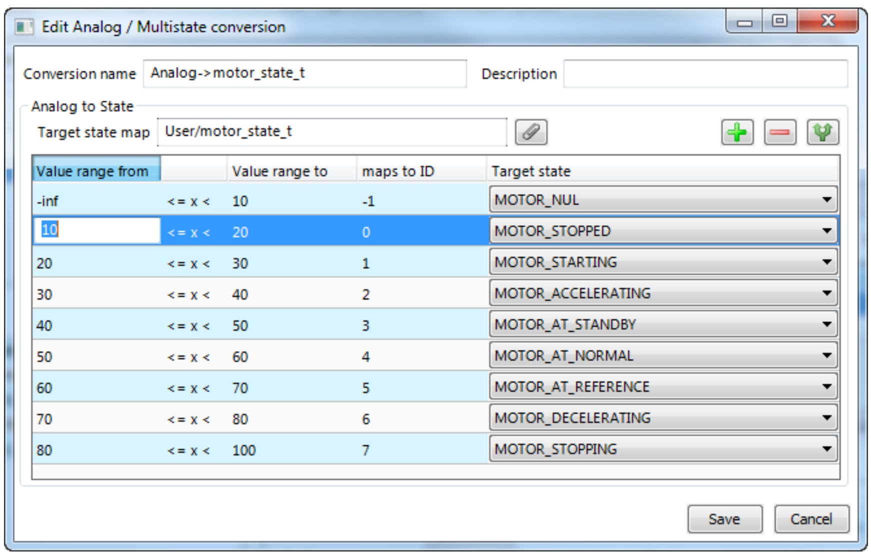


Figure 176: Edit an analog to multi-state conversion.

12. An example for editing an analog to multi-state value conversion is shown in Figure 176. Enter a **Conversion name**, then edit the **Value range from** column and select the desired **Target state** mapping.

13. Click **Save** and then choose the newly created conversion by clicking **Select**.

14. The items with an assigned value conversion appears with a green checkmark.

To Create a Connection via Drag-and-Drop

1. Change to the **Datapoints** tab of the main window and navigate to the data point that shall be put into the new connection.

2. In the properties view below the data point list click on the **Local Connections** tab as shown in Figure 177.

3. Then simply drag a data point from the data point list and drop it onto an empty area in the connections list as shown in Figure 177.

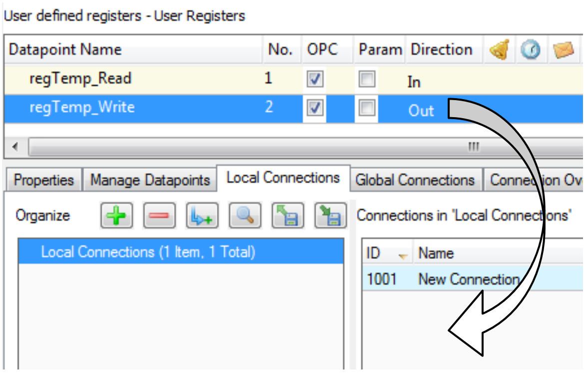


Figure 177: Create a connection with drag-and-drop.

4. To add data point to that connection, drop the new data point into the empty area in the connected data points list below.

To Create a Bi-Directional Connection

1. Start the connection by adding a data point. A value data point is added as **Send** to a new connection.

2. Then add the same data point a second time and do not create a multi-slot connection. This time it is added as **Receive**. This makes the data point send to and receive values from the connection.

3. Continue by adding the data point, which shall be connected. A value data point is added as **Receive** to an existing connection.

4. Then add the same data point a second time to that connection. This time it is added as **Send**.

5. Now both data points send values to and receive values from the connection. This synchronizes value changes back and forth between the involved data points. Update loops are suppressed by the connection. It is not necessary to set a COV on any of the involved data points. An example is shown in Figure 178.

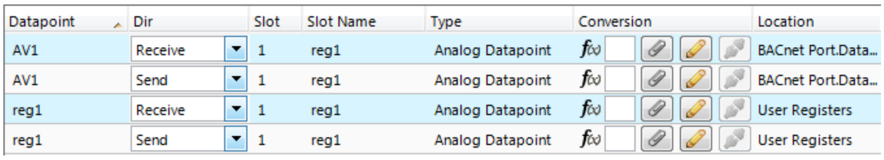


Figure 178: Bi-directional connection.

7.9.2 Create Connections from a CSV File

A quick way to perform batch edit on connections is to export and import connections from the connections CSV file. Each line in the connections CSV file identifies a connection. The first column is the connection name. The second column specifies the hub data point. The full path to the data point must be specified using the dot '.' as the folder separator. The third and following columns specify the target data points.

To Create Connection from a CSV File

1. Select the menu **Tools → Export Local Connections …**

2. Select an appropriate file name and export.

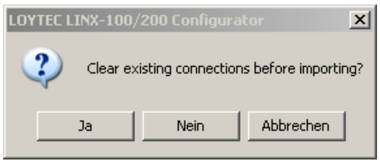
3. Edit the connections CSV file. An example is shown in Figure 179.

```
#connection_csv_ver,1
#ConnectionName,HubDPName,TargetDPName
Ai0,LINX-200.BACnet Port.ai0,LINX-200.User Registers.abs_humid1
Ai1,LINX-200.BACnet Port.ai1,LINX-200.User Registers.abs_humid2
Ai2,LINX-200.BACnet Port.ai2,LINX-200.User Registers.abs_humid3
Ai3,LINX-200.BACnet Port.ai3,LINX-200.User Registers.abs_humid4
```

Figure 179: Example Connection CSV File.

4. Select the menu **Tools → Import Local Connections …**

5. If connections that are not part of the connection CSV file shall be deleted, click **Yes** when prompted. Click **No** if the other connections shall be left as is.



6. Choose the file to import and click **Ok**.

7. When the import has completed, optionally view the log to check, which connections have been added, modified, and deleted.

7.9.3 Modify Connections

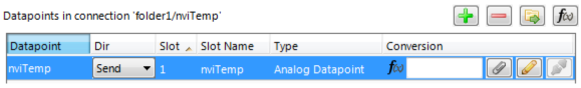
Connections can be edited and deleted. This is also done in the **Connections** tab of the main window. Editing connections does not influence the data point configuration. This means, when deleting a connection or adding/removing data points to/from a connection, the data points are not deleted.

To Edit a Connection

1. Change to the **Local Connections** tab of the main window.

2. Select the connection to edit. Then follow the steps as applied when creating a connection.

3. To detach a data point from the selected connection, select the data point and click on the button **Detach selected data points** over the connection member list.



4. Change the direction of a data point in the connection by choosing one from the **Dir** drop-down. You may select **Disable** to temporarily exclude this data point from the connection altogether.

To Add Data Points via Drag-and-Drop

1. Change to the **Datapoints** tab of the main window and navigate to the data point for being added.

2. In the properties view below the data point list click on the **Connections** tab as shown in Figure 180.

3. Select an existing connection.

4. Drag the selected data point and drop it into the empty area of the **Datapoints in connection** list as shown in Figure 180. This adds the data point to the selected connection.

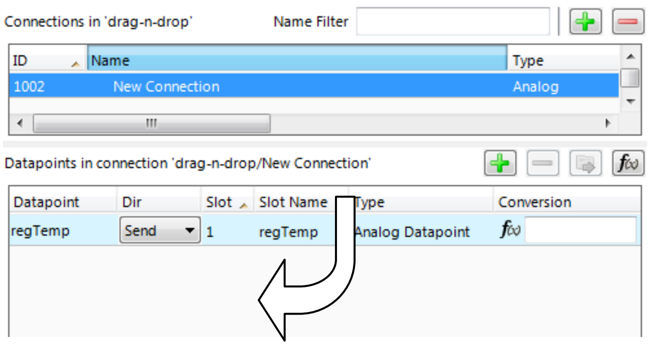


Figure 180: Modify connections in the properties view.

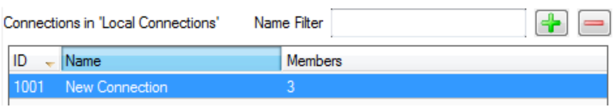
5. To replace a data point in a connection drop the new data point right onto an existing data point in the connection that shall be replaced.

To Delete a Connection

1. Change to the **Local Connections** tab of the main window.

2. Select the connection for removal. Use multi-select to select more than one connection.

3. Click on the button **Delete Connection** over the connections list.



7.9.4 Create a Multi-Slot Connection

A multi-slot connection can be used to connect a number of different data points together under one umbrella. It can be considered as a cable with many wires, each wire represented by a slot with a label. For example, one can create a multi-slot connection for a structured data point, connecting each sub-data point to another technology. All those slots appear under the same connection. But data point only exchange data, if they are added to the same slot.

To Create a Multi-Slot Connection

1. Change to the **Datapoints** tab of the main window and navigate to the data point to be connected.

2. Select a structured data point and drag it into the connections list to create a new connection.

3. A dialog prompts the user, whether to create a multi-slot connection or use the user data point as a single entity in the connection. Choose **Yes** to create a multi-slot connection.

4. A multi-slot connection is created as shown in Figure 181. The multi-slot connection can be collapsed or expanded. In the expanded view it shows all slots. Select the top-level multi-slot connection to view all data points in the connection. Select a single slot beneath to view only those data points in that slot.

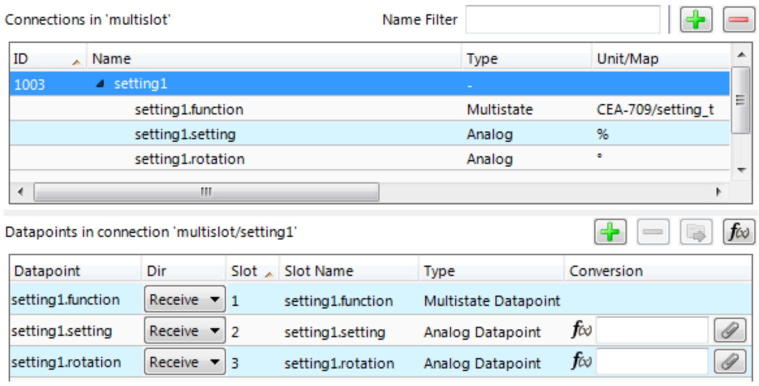


Figure 181: Multi-slot connection.

5. Add more slots by dropping data points onto the top-level multi-slot connection ‘setting1’.

6. Connect other data points to the slots by dropping them onto the slots. For example connect the register ‘regRotation’ by dropping it onto ‘setting1.rotation’ as depicted in Figure 182.

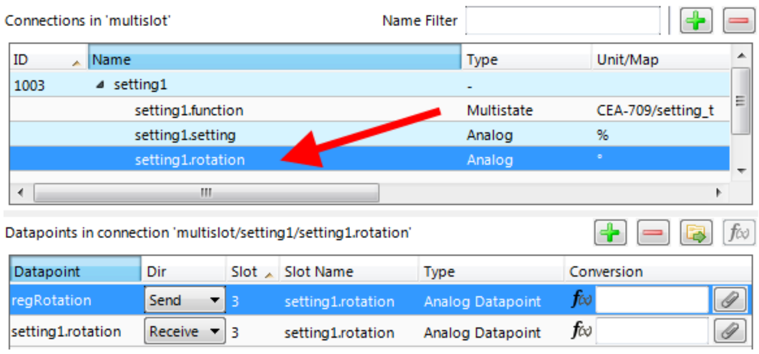


Figure 182: Add data point to connection slot.

7.9.5 Create a Math Block Adaptor

When connecting structured data points the multi-slot connection can be used. If a simple mapping of the sub-data points is not possible and a more advanced mathematical conversion is required, a math block adaptor can be created. This math block is based on a multi-slot connection with *n* inputs and *m* outputs (see Section 6.3.2).

To Create a Math Block Adaptor

1. Create a multi-slot connection from a structured data point, e.g., the input data point.

2. Add output slots to the multi-slot connection, e.g. by adding a structured output data point. An example is shown in Figure 183.

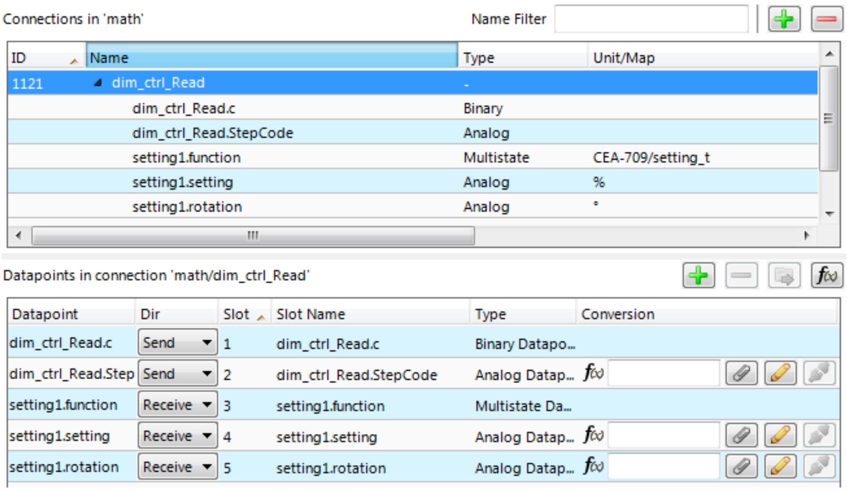


Figure 183: Multi-slot connection for math block adaptor.

3. Click on the **Create math adapter from connection** button. The dialog **Edit Multi-Slot Math Adaptor** opens as shown in Figure 184.

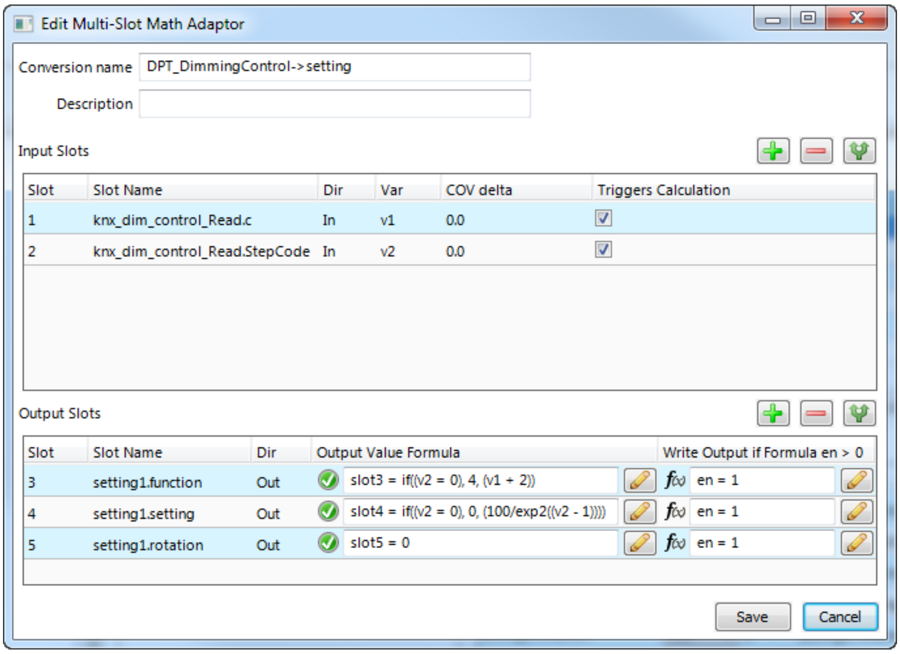


Figure 184: Edit a math block adaptor.

4. Enter a **name** and **description** for the adaptor.

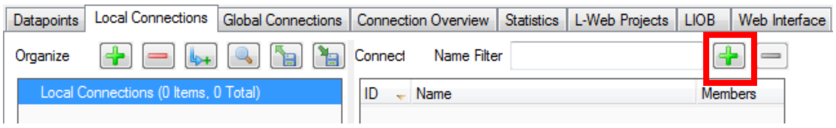
5. For each output enter an **Output Value Formula**. This can be done by directly typing the formula or by clicking the edit button .

6. Optionally, etner an enable formula into **Write Output if Formula en > 0**. As a default enable is ‘1’.

7. When finished with the math block click **Save**.

To Use an Existing Math Block Adaptor

1. Click on the **Local Connections** tab



in the main window and press the speed button **Create new Connection**. In the menu choose **Connection with Math Adaptor**.

2. In the dialog **Select Multi-Slot Math Adaptor** select an existing adaptor and click **Select**. A new multi-slot connection is added to the connection list with empty slots as depicted in Figure 185.

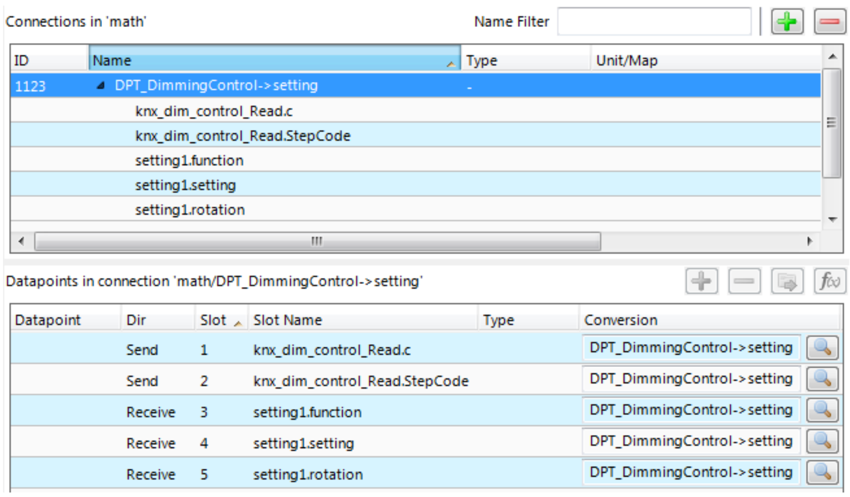


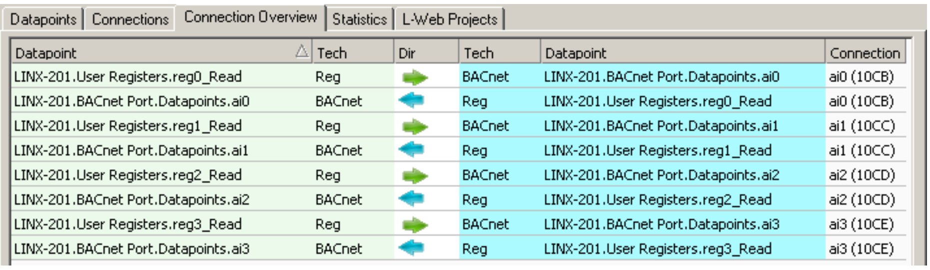
Figure 185: Created new multi-slot connection from math adaptor.

3. Now connect data points by dragging and dropping them onto the empty slots in the data point list below.

4. To view the math conversion click the magnifier button .

7.9.6 Connection Overview

Select the **Connection Summary** tab to get a graphical representation of all connections. It represents the two connected data points, their technology they are based on and the direction of the connection. An example for the overview is shown in Figure 186.



Datapoint	Tech	Dir	Tech	Datapoint	Connection
LINX-201.User Registers.reg0_Read	Reg		BACnet	LINX-201.BACnet Port.Datapoints.ai0	ai0 (10CB)
LINX-201.BACnet Port.Datapoints.ai0	BACnet		Reg	LINX-201.User Registers.reg0_Read	ai0 (10CB)
LINX-201.User Registers.reg1_Read	Reg		BACnet	LINX-201.BACnet Port.Datapoints.ai1	ai1 (10CC)
LINX-201.BACnet Port.Datapoints.ai1	BACnet		Reg	LINX-201.User Registers.reg1_Read	ai1 (10CC)
LINX-201.User Registers.reg2_Read	Reg		BACnet	LINX-201.BACnet Port.Datapoints.ai2	ai2 (10CD)
LINX-201.BACnet Port.Datapoints.ai2	BACnet		Reg	LINX-201.User Registers.reg2_Read	ai2 (10CD)
LINX-201.User Registers.reg3_Read	Reg		BACnet	LINX-201.BACnet Port.Datapoints.ai3	ai3 (10CE)
LINX-201.BACnet Port.Datapoints.ai3	BACnet		Reg	LINX-201.User Registers.reg3_Read	ai3 (10CE)

Figure 186: Connections Summary.

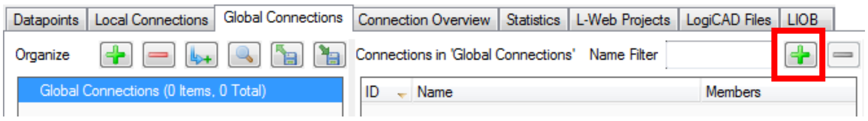
7.9.7 Create a Global Connection

Global connections are an easy way to publish or subscribe to global data, which is distributed among devices. To configure such communication, a device needs to be member of a CEA-852 channel. Once it is member of that channel, global connections need to be configured in the data point configuration. This is similar to creating local connections and most of the configuration steps apply also for global connections. In addition, also network timing parameters can be configured for global connections. For a description see Section 6.3.4.

If other devices already have global connections configured that publish data on the network, their definition can be exported and imported into the new configuration. That will make all the global connection names available. When creating manually, the connection names can be edited.

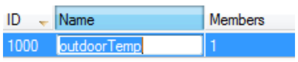
To Create a Global Connection

1. Click on the **Global Connections** tab



in the main window and press the speed button **Create new Connection**. A new connection is added to the connection list.

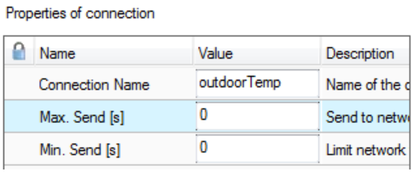
2. Define a name for the global connection. This name is required to be unique on the network. Data will be published or subscribed to under this name.



3. Add data points to the global connection as described in Section 7.9.3. As a default, output data points will be added as sending, input data point as receiving data under the global connection.

4. Change the direction by modifying **send** or **receive**. For changing multiple data points use multi-select.

5. Define timing parameters for a global connection that is sending out data. On the global connections tab of the main window the connection properties are listed below the data point member list. In the tab of the property area click the button .



6. To export the definitions of the created global connections, click the button **Export connections to disk** and choose the XML format.

7. To use those definitions, click the button **Import connections from disk** and choose an exported connections XML file. This creates the global connections structure with connection names but without any data points. Data points can then be added, for example, via drag-and-drop.

7.9.8 Automatic Generation of Connections

The *smart auto-connect* feature of the Configurator provides a quick way to automatically generate target data points out of a source data point selection and generate connections to them. Using this feature a gateway interface is generated with a few mouse clicks.

To Auto-generate Data Points and Connections

1. Go to the **Datapoints** tab.

2. Select those data points of a given port folder that shall be mapped to another technology. The methods innclude sub-folders, data point name filter and multi-select may be used for doing this.

3. Click on the speed button **Generate and connect selected** in the tool bar.

4. Alternatively, you can select the port folder or any sub-folder and click the speed button **Folder-wide Generate points and auto-connect** in the tool bar. This generates target data points and connections for all data points in the folder.

5. The auto-generate preview dialog opens as shown in Figure 187. Choose the target technology. The preview results show a list for each source type found how it will be created as a target type. The **Type Name** column provides a drop-down to modify the result. This choice is remembered and will be applied the next time again. You may click the **Restore defaults** button to revert all custom settings.

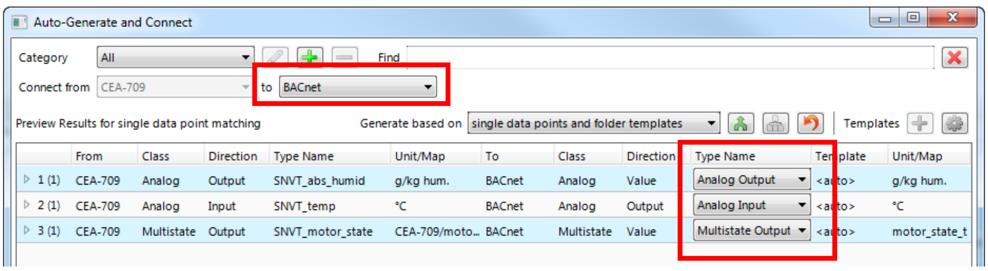
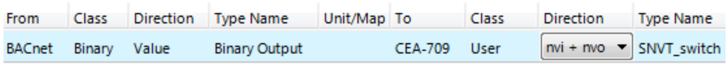


Figure 187: Auto-generate and connect preview.

Note: *The respective port may have to be enabled first in the project settings to have the target technology available in this dialog.*

6. If the selected target technology offers choices on the direction to create, choose one of the offered directions.



7. Structured data points will be flattened in some target technologies. To prevent this from happening, click the **Don't expand structured data points** button. Note, that this may require an auto-generate template, which defines how to map this structured data point.

8. Select an auto-generate template in the drop-down box of the **Template** column as shown in Figure 188.

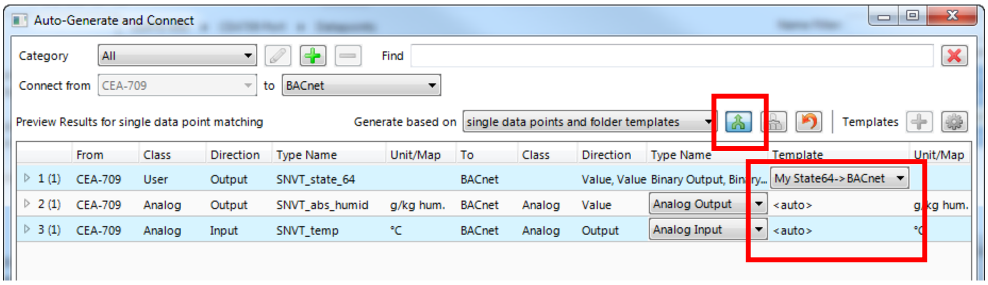


Figure 188: Auto-generate and connect with auto-generate template.

9. After having reviewed all types, click the **Generate** button.

Note, when auto-creating the target data points, the Configurator initializes their properties with default values derived from the properties of the source data points. In particular, the data point name, description, minimum and maximum value, and engineering units are generated. If the default properties do not have the desired values, the user can edit them in the target folder. The user can also craft an auto-generate template where those properties are pre-set.

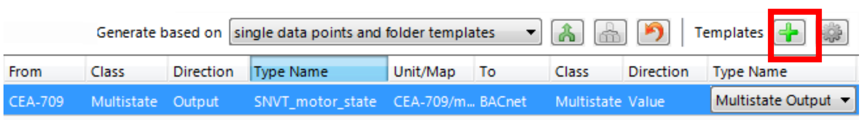
7.9.9 Create an Auto-Generate Template

If the implicit options for generating target types are not sufficient for the envisioned task, a specialized auto-generate template can be crafted. This template contains copies of the source data points as a starting point. The user can then create instances of the desired target data points and use one or more connections (e.g., a multi-slot connection for sub-data points of a structured source). The target data point name and description can contain variable placeholders, which expand to the name and description of the actual sources, which the auto-generate template will be applied on. If required, one can add conversion adaptors including math blocks. The entire configuration serves as a template on how to generate the target data points and apply the appropriate connections. Once saved in the template library, it will be available for selection in the preview dialog.

To Create an Auto-Generate Template

1. Select the source data points and invoke auto-generation as described in Section 7.9.8.

2. In the preview dialog select the source type for which a new template shall be created and click the **Create template for selected source** button.



3. The auto-generate template editor opens as shown in Figure 189. It contains two data point lists. The left-hand (denoted ‘1’) list is pre-filled and contains the selected source data points. They are locked and cannot be modified. The right-hand list (denoted ‘2’) is intended for creating the desired target data points.

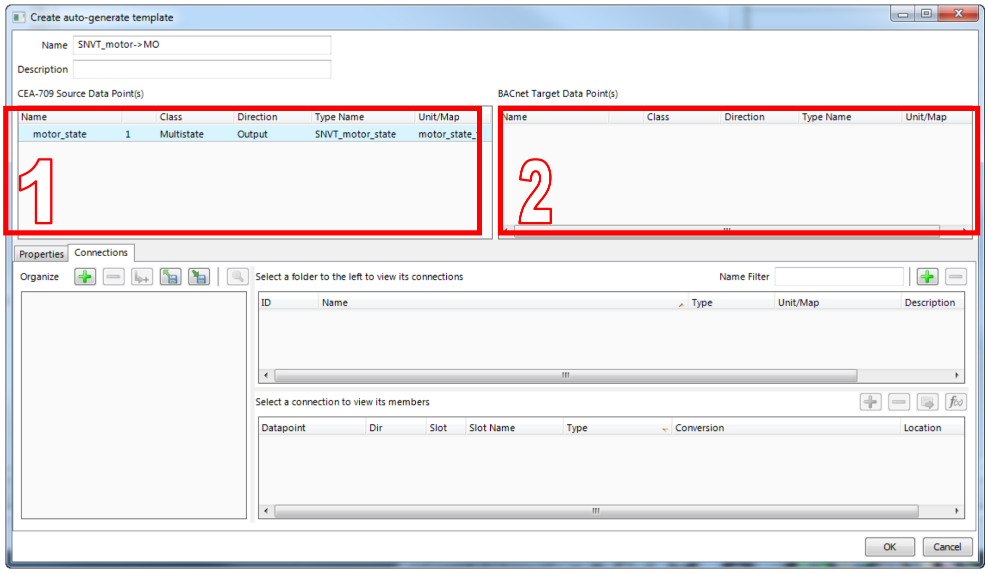


Figure 189: Auto-generate template editor.

4. Enter Name and Description, which is used later to select the auto-generate template.

5. Right-click in the target data point list and select **Create Data Point …**. The creation dialog of the respective target technology opens. Create the desired target data points, e.g., a BACnet ‘MO’ data point with a custom multi-state map.

6. Edit the target data point name to use a variable placeholder for the target data point name such as **`%{name}`**. Insert the placeholder by selecting the desired entry of the **add var** drop-down list. When applying the auto-generate template, the placeholder is expanded to the name of the actual source data point. Choose **`%{path}%{name}`** to flatten the folder tree of the source data point and include the path in the target data point name.

7. Drag and drop the source and target data points into the **Connections** tab below to create the needed connections. Add custom conversions to the connection items as needed.

8. Click **OK** to store the auto-generate template.

9. From now on it can be selected in the **Template** column and applied to the source.

From	Class	Direction	Type Name	Unit/Map	To	Class	Type Name	Template	Unit/Map
CEA-709	Multistate	Output	SNVT_motor_state	motor_state_t	BACnet	Multistate	Multistate Output	SNVT_Motor->MO	bac_fan_states

7.9.10 Create a Complex Auto-Generate Template

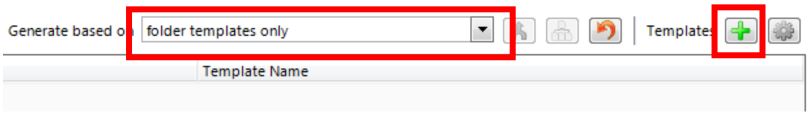
Simple auto-generate templates as described in Section 7.9.9 are based on single data point sources. These can be scalar or structured data points. In any case the decision, which auto-generate template applies, is based on that single data point. If a given set of source data points shall generate another specific set of target data points, so-called *complex* auto-generate templates can be used. These are based on folders that contain the described data points (i.e., name and types must match). With a complex auto-generate template the entire folder is used as a source and an entire target folder will be generated with the target data points defined by the complex auto-generate template.

For example, there are device folders of similar devices on the BACnet port. These shall serve as the connection source. One such device folder contains three analog data points named ‘TempComfort’, ‘TempNight’, ‘RoomTemp’ that have ‘°C’ as engineering units. These shall generate two target data points, one a structured data point with the setpoints ‘TempComfort’ and ‘TempNight’ connected (e.g. a SNVT_temp_setpts) and one data point with the ‘RoomTemp’ connected (e.g., a SNVT_temp). A complex auto-generate template is created based on one of the device folders.

To Create a Complex Auto-Generate Template

1. Select the source folder and invoke auto-generation as described in Section 7.9.8 using the button **Folder-wide Generate points and auto-connect** in the tool bar.

2. In the preview dialog choose generation based on **folder templates only**. The list will then be empty as no such template yet applies. Then click the **Create template for selected source** button.

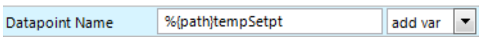


3. The auto-generate template editor opens as shown in Figure 189. It contains two data point lists. The left-hand (denoted ‘1’) list is pre-filled and contains the selected source data points. They are locked and cannot be modified. The right-hand list (denoted ‘2’) is intended for creating the desired target data points.

4. Enter **Name** and **Description**, which is used later to select the auto-generate template.

5. Right-click in the target data point list and select **Create Data Point …**. The creation dialog of the respective target technology opens. Create the desired target data points, e.g., a SNVT_temp_setpts and a SNVT_temp.

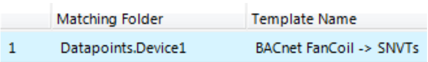
6. Edit the target data point name and local NV name. Since the source data point names will be equal for all source folders, an additional component may be added to the target data point name. Use the variable placeholder `%{path}` in the name to make it unique. Insert the placeholder by selecting the desired entry of the **add var** drop-down list. When applying the auto-generate template, the path placeholder is expanded to the actual folder path relative to the **Datapoints** folder.



7. Drag and drop the source and target data points into the **Connections** tab below to create the needed connections. Add custom conversions to the connection items as needed.

8. Click **OK** to store the auto-generate template.

9. From now on it can be selected in the **Template** column and applied to matching source folders.



7.9.11 Managing Connection Resources

All described resources for connections, including connection adaptors and auto-generate templates are part of the configuration project. When opening the project file on another PC, all project resources will be merged into the local resource repository. After opening a project file, all its resources are therefore available to new projects.

The Configurator provides a resource manager, that can be used to view, edit, import and export connection resources. Select the menu **Tools → Manage Connection Adaptors …** or **Tools → Manage Auto-Generate Templates …** to open the resource manager dialogs.

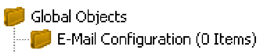
7.10 E-mail Templates

7.10.1 Create an E-mail Template

E-mail templates are used to assemble and transmit e-mails when certain trigger conditions occur. The e-mail template contains the destination e-mail address, the subject, and text. Variable parameters can be added to the text by using data point sources. The transmission of an e-mail is triggered by one or more trigger data points. For setting up e-mails, the e-mail account information has to be configured on the device, e.g., on the Web UI (see Section 5.2.10).

To Create an E-mail Template

1. Under the **Global Objects** folder, select the **E-mail Configuration** sub-folder.



2. Right-click and select **New E-mail Template …** from the context menu.

3. In the **Configure E-mail Template** dialog, which is shown in Figure 190 enter the **To** address and the **Subject**. Optionally, **Cc** and **Bcc** addresses can be specified.

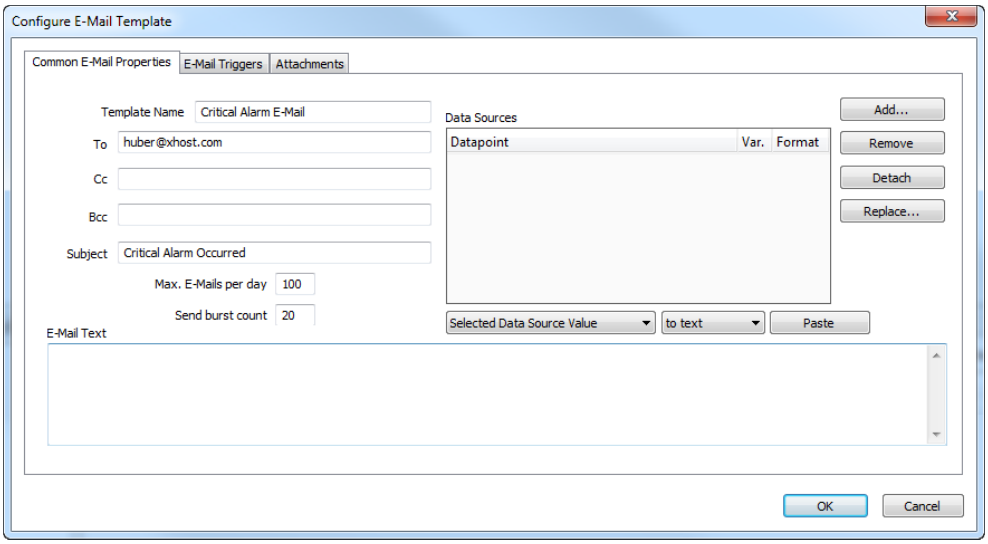
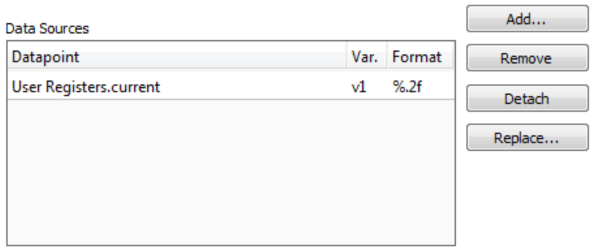


Figure 190: Configure E-Mail Template Dialog.

4. Enter text in the **E-mail Text** multi-line field.

5. If the e-mail text shall contain values of data points, add data points to the **Data Sources** list by clicking the **Add…** button.

6. A data point selector dialog opens. Select one or more data points and click **OK**. The selected data point appears in the **Data Sources** list.



7. If necessary, edit the **Format** string. The default '%.2f' will format the value as a floating point with 2 decimals.

8. Select the data point in the **Data Sources** list. In the drop-down box underneath select **Selected Data Source Value**, in the next drop-down select **to text** click the **Paste** button. Variables can also be pasted to the subject line or any of the address lines.

9. A place holder `%{v1}` for the data point value appears now in the e-mail text.

10. To replace an existing data source select the data point in the **Data Sources** list and click the **Replace…** button. This opens a data point selector dialog for choosing the replacement data point.

7.10.2 Trigger E-mails

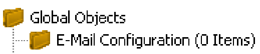
E-mail templates are used to assemble and transmit e-mails when certain trigger conditions occur. For an e-mail template, one or more trigger conditions can be defined. The e-mail will be sent, when one of the trigger conditions is activated. Depending of the trigger data point type, the trigger conditions can be refined.

Note, that the behavior of the trigger data point is influenced by the COV properties of the data point. If the **Only notify on COV** property is checked, the data point triggers only if its value changes to the value of the trigger condition. If that property is not checked, the data point triggers on every write with a value that matches the trigger condition.

The trigger for sending an e-mail can be enabled or disabled altogether by using an *enable* data point. This data point must be of type *binary*. If the value of that enable data point is TRUE, the trigger conditions are evaluated. If the value of the enable is FALSE, no e-mails are be triggered.

To Create an E-mail Trigger

1. Under the **Global Objects** folder, select the **E-mail Configuration** sub-folder.



2. Right-click and select **Configure E-mail Template …** from the context menu.

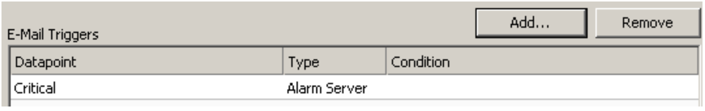
3. Change to the **Mail Triggers** tab.

Note: *Of course, you can also change directly to the **Mail Triggers** tab when creating an e-mail template.*

4. Click the **Add…** button. A data point selection dialog opens.

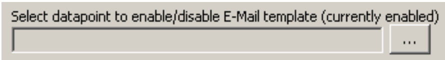
5. Select one or more data points and click **OK**.

6. The triggers appear now in the **E-Mail Triggers** list. The data points that serve as e-mail triggers also appear with the e-mail icon in the data point list.



7. In the **Manage Trigger Conditions** you can setup the trigger condition depending on the trigger data point class.

8. If the trigger condition is depending on the value of an enabling data point, you can add an enable data point by clicking on the **…** button.



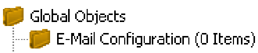
9. To remove such a trigger enable, click the **Remove Enable Trigger** button.

7.10.3 Attachments

E-mail templates can be configured to have file attachments. Basically, any file of the device can be specified as an attachment.

To Configure Attachments

1. Under the **Global Objects** folder, select the **E-mail Configuration** sub-folder.



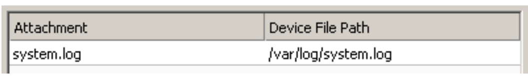
2. Right-click and select **Configure E-mail Template …** from the context menu.

3. Change to the **Attachments** tab.

Note: *Of course, you can also change directly to the* ***Attachments*** *tab when creating an e-mail template.*

4. Select an available file from the **Attach File** drop-down box.

5. Click the **Add** button. The file appears in the **Attachments** list.



6. To remove an attachment, select the attachment file in the **Attachments** list and click the button **Remove**.

7.10.4 Limit E-mail Send Rate

The transmission of e-mails is triggered by the configured trigger conditions. It is not predictable, how often the trigger condition will cause the transmission of an e-mail. The e-mail template can be configured to limit the number of transmitted e-mails. This is done in the Configure E-mail Template dialog.

To configure an E-mail Rate Limit, configure the settings:

- **Max. E-mails per day**: This setting defines how many e-mails can be sent on average per day. The actual number of transmitted e-mails on a specific day may be slightly higher than this setting, depending on burst rates. The default is 100 e-mails per day. This results in an average interval of one e-mail per 14 minutes.
- **Send burst count**: This setting defines how many e-mails may be transmitted shortly after each other not limited by the above average interval. After the burst count, the average mails per day limit takes effect. The default is a maximum of 20 e-mails in a row.

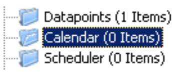
7.11 Local Schedule and Calendar

7.11.1 Create Calendar Patterns

When working with global definitions for calendar-based exception days such as “Holidays” a local calendar is used and needs to be configured with calendar patterns. A calendar pattern represents a class of days such as “Holidays”. The calendar patterns can then be used in a schedule to define daily schedules for exception days. The available calendar patterns should be created when the system configuration is engineered. The actually dates in the calendar patterns can be modified later at run-time.

To Create a Calendar Pattern

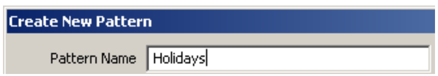
1. Locate the calendar object. When using a generic scheduler the corresponding generic calendar already exists in the **Scheduler** folder under the device folder. For a technology calendar, select the **Calendar** sub-folder of the respective port.



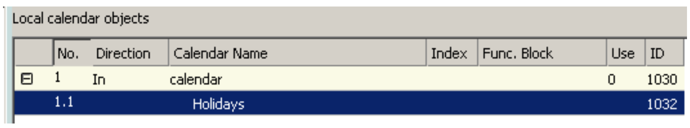
2. Select the calendar data point.

3. Right-click and select **Create Calendar Pattern…**

4. Enter a Pattern Name in the **Create Calendar Pattern** dialog



5. Click **Create Pattern**. The dialog closes and the calendar pattern appears beneath the calendar data point.

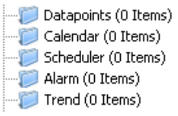


7.11.2 Create a Local Scheduler

For scheduling data points, a scheduler object must be created. Under each port folder, multiple local scheduler objects can be created. These local schedulers can then be configured to schedule data points.

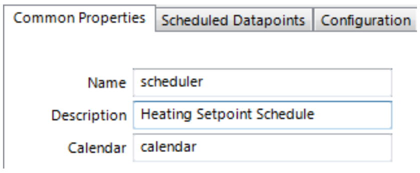
To Create a Local Scheduler

1. Under the device folder, select the **Scheduler** sub-folder to create a generic scheduler. For a technology scheduler, select the **Scheduler** sub-folder of the respective port.



2. Right-click in the data point list view and select **New Local Scheduler …**.

3. Enter a name for the schedule and a description. Note, that the schedule automatically detects a calendar, if it has previously been created.



4. Click **Create Schedule**. The new schedule appears in the data point list of the Scheduler sub-folder.

7.11.3 Configure Scheduled Data Points

When a local scheduler has been created, it needs to be configured, which data points it shall schedule. This is done by attaching data points to the scheduler. Note, that there may be limits, how many and which data points may be attached (see Section 6.5.1).

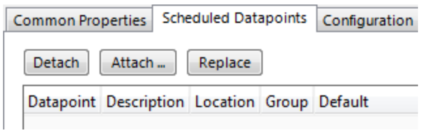
This configuration must be done as an initial setup. The daily schedules can be changed later in the Web UI or over the network. Which data points are scheduled can be changed in the Web UI.

To Attach Data Points to a Scheduler

1. Select the scheduler data point in the Scheduler sub-folder.

2. Right-click and select **Configure Schedule** from the context menu. The same dialog which appears when a new scheduler is created is shown and allows configuring the scheduler. Of course, this step can also be done directly when the data point is created.

3. Select the tab **Scheduled Datapoints**.

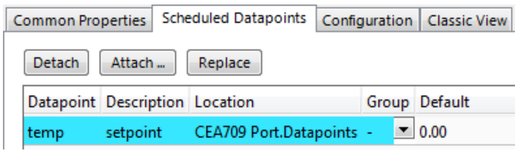


4. Click the button **Attach …** . This opens another data point selector window.

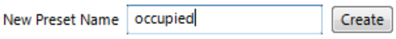
5. Select the data points to attach and click **OK**. For each of the attached data points, one or more lines appear in the list below the attach button. If the attached point is a structure, there will be one line for each element of the structure.

Tip! *Data points can also be attached to a scheduler by selecting a data point in the data point manager, drag it onto a scheduler data point and drop it on the scheduler data point.*

6. Enter a short text in the **Description** field in the second column of each line. This text will serve as a label, which will be shown on the device's UI to identify the data point.



7. Add new value presets by entering a name and pressing the **Create** button next to the input field.



Tip! *To generate presets automatically for multi-state data points, click the* ***Auto-Create*** *button. This button is available, if no other presets have been defined yet.*

8. For each new preset, a new column will appear in the list. In this column, enter the desired value for each of the attached points, which will be set when this value template is scheduled. The user may later edit the values for each preset on the device but cannot add new value presets unless there is only one line (one value) in the list.

9. If there are multiple output values which belong together, they can be grouped in order to save space on the device. For each group, the entered value is stored only once, even if there are more data points in the same group.

10. When done with the point and value setup, switch back to the **Configuration** tab or click **Save Changes** to leave the dialog.

Tip! *A shortcut to creating a scheduler object and attaching a data point is to select a data point in the data point manager, right-click on it and choose* ***Schedule Datapoint*** *from the context menu. This generates a scheduler and links that data point to it.*

7.11.4 Configure Scheduled Events

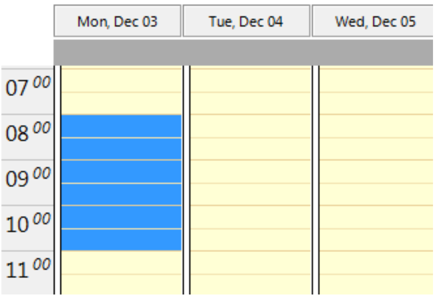
Once a scheduler is configured with attached data points and value presets, the schedule events can be defined. This can be done on the device or over the network at run-time, or also in the configuration software. A schedule consists of a number of scheduled events, each event starts at a defined time, has a scheduled value and ends at a defined within the same 24-hour period (starting at 00:00 and ending at 23:59 hours). The event can be configured to occur at a given date, for each weekday, or for a number of recurring dates.

In addition, scheduled events can be configured to occur on exception days from a calendar, such as "Holidays". An exception day always overrides a normal weekday. If more than one exception day is used, a priority must be assigned. This is necessary so that the system knows which schedule to follow on a day which matches more than one calendar pattern. Considering the priorities the calendar preview shows the effective schedule on a given day.

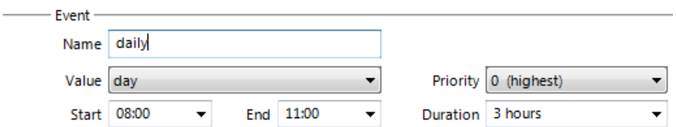
To Configure a Scheduled Event

1. Open the **Configure Schedule** dialog and click on the **Configuration** tab (see Section 7.11.3).

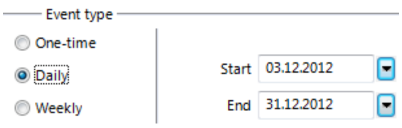
2. In the calendar view select the day for which to configure the scheduled event. Then select the event duration by clicking into the daily planner and dragging the mouse to the desired duration.



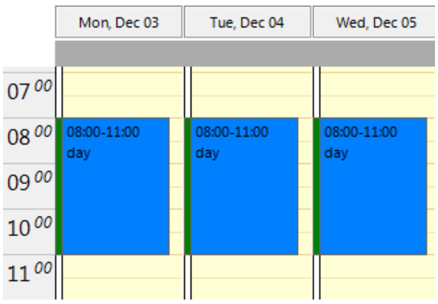
3. Then click on the button **Create new scheduled event** and enter a **Name** for the scheduled event (note, in BACnet there is no name to be specified). Choose the scheduled **value** or enter a scheduled value. Modify the **Start** and **End** time to your needs. Optionally you should set a priority, if scheduled events overlap in the preview.



4. Choose an **Event type**, which defines how the event is recurring.



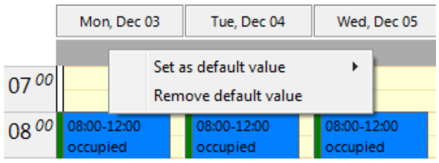
5. The click **OK**. The new event appears in the daily planner.



6. For viewing more details on overlapping events you can change to the **day view**. In all views events can be extended or shrunk using the upper or lower boundary handles or moved by clicking and dragging.

7. Right-click on an event in order to edit it. You can choose to modify it, change its color, or delete it.

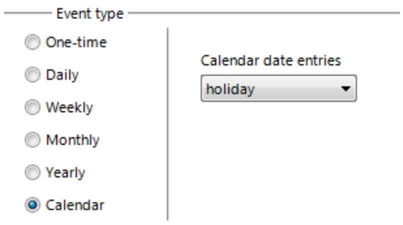
8. To set a default value, right-click onto the grey area right above the day planner. In the context menu choose a default value.



To Use Exception Days from a Calendar

1. For letting a scheduled event occur on exception days from a calendar, create a new scheduled event.

2. Select the event type **Calendar**. and choose one of the defined **calendar date entries**.



3. Note, that if the scheduled event may overlap with other events. In this case edit the priorities of the scheduled events. For example, if a given calendar day falls in both categories, “Holidays” and “Maintenance”, the scheduled event with the higher priority becomes effective. The highest available priority is marked **highest**. Note that the actual priority values depend on the technology (see Section 6.5.1).

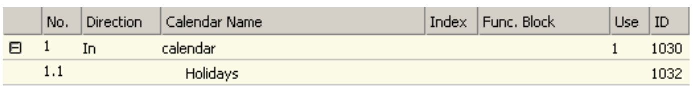
Important!	***Choose different priorities for different scheduled event. If two scheduled events overlap and their priorities are equal, it is not determined, which value is in effect.***

7.11.5 Configure Exception Days

When a local calendar is used, its calendar patterns need to be configured with exception days (pattern entries). The calendar patterns can be configured in the Configurator software, modified at run-time over the Web UI or over the network. When configuring in the software, the current exception days should be uploaded from the device, to work on the current configuration (see Section 7.4.3).

To Configure Exception Days in a Calendar Pattern

1. Select the **Calendar** sub-folder and select the calendar pattern, which shall be configured



	No.	Direction	Calendar Name	Index	Func. Block	Use	ID
⊟	1	In	calendar			1	1030
	1.1		Holidays				1032

2. Right-click and select **Configure Pattern …** in the context menu.

3. The **Configure Pattern** dialog appears as shown in Figure 191. Add dates to the calendar pattern by entering a Date Configuration. Then click **Add Entry**. The date appears in the Pattern Entries list on the right-hand side.

4. Edit an exception by selecting the pattern entry in the **Pattern Entries** list. Then modify the date configuration in the **Date Configuration** group box.

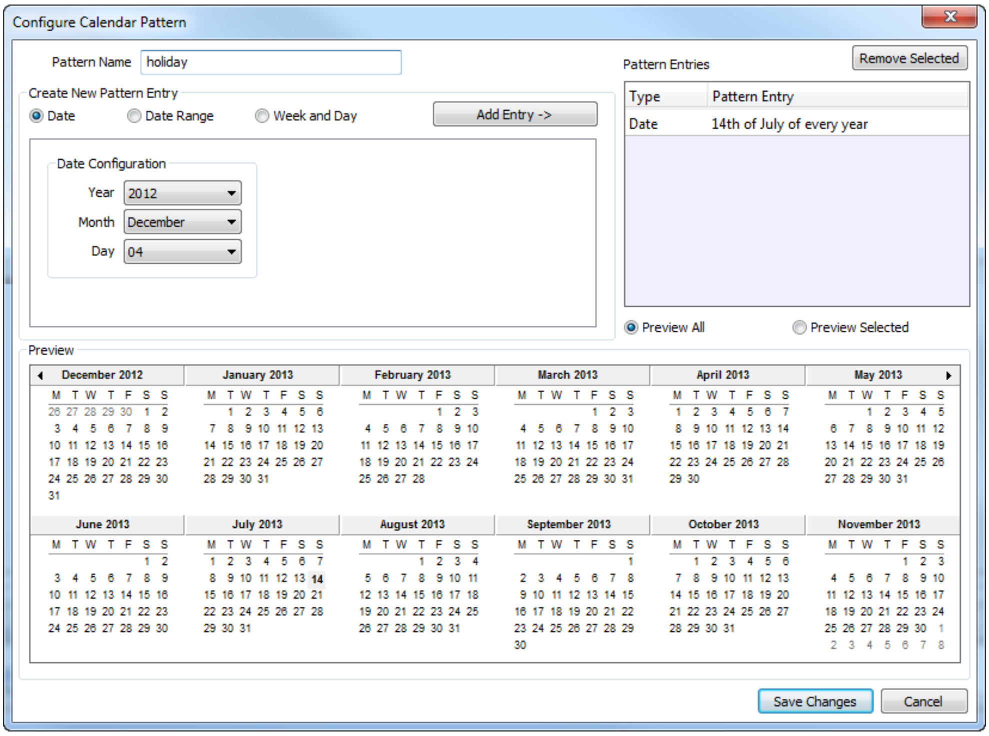


Figure 191: Configure Calendar Pattern Dialog.

5. Click **Save Changes** when all exception days have been entered.

Tip!	*If not sure, how a date configuration affects the calendar days, click on a pattern in the **Pattern Entries** list and the affected days will be highlighted in the **Preview**.*

7.11.6 Configure Control Data Points

A scheduler object can be configured to use special control data points. These data points can control the scheduler and expose additional state information of the scheduler on the network. The following control data points are available:

- **Scheduler Enable/Disable Datapoint**: This data point can be configured, which enables or disables the scheduler depending on its Boolean value.

- **Enable/Disable Feedback Datapoint**: This data point is updated with the current enabled state of the scheduler. This also reflects and an enable from the network.

- **Scheduled Preset Name**: This data point can be attached to be updated with the name of the currently active preset. Only string data points can be attached.

To Configure Control Data Points

1. Open the **Configure Schedule** dialog to configure daily schedules as described in Section 7.11.3.

2. Go to the **Scheduled Datapoints** tab.

3. In the Control Datapoints group box, click the ... button to add the desired control data point. A data point selection dialog opens.

4. Select a matching data point and click **OK**. For the preset name a string data point must be selected.

5. To remove an undesired control data point, click on the **Remove** button.

7.11.7 Using the Local Scheduler

Once the setup of the local scheduler is done, it is basically operational. It will immediately start to work based on the configuration data downloaded through the configuration software. You can verify the daily schedules and values of scheduled data points on the Web UI (see Section 5.4.3). The local schedule can be altered over the Web UI or over the network using the underlying networking protocol.

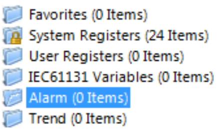
7.12 Local Alarming

7.12.1 Create an Alarm Server

To generate local alarms, either an existing (pre-allocated) alarm server can be used or a new alarm server needs to be created at first. The local alarm sources will report alarms to that alarm server. The alarm server is the interface to access local alarms. This can be done over the network or the Web UI.

To Create an Alarm Server

1. Under the device folder, select the **Alarm** sub-folder to create a generic alarm server. For a technology alarm server, select the **Alarm** sub-folder of the respective port.



2. Right-click in the data point list view and select **New Alarm Server …**.

3. In the **Create New Alarm Server** dialog box (as shown in Figure 192) enter **Name** and **Description** of the alarm server.

4. For alarm transitions you may define, which require acknowledgement and at which priority they are reported.

5. You may attach data points for storing alarm counters. These will be linked using the respective property relations (see Section 6.1.12).

6. When you create a generic alarm server, you may add technology alarm servers, that shall be reported to. Click on and choose one alarm server of each technology. If they don't exist, you need to create them under the technology port folders.

7. Click **Create**. The alarm server appears now in the data point list view.

8. For a BACnet technology alarm server, edit the instance number of the Notification Class object to your needs.

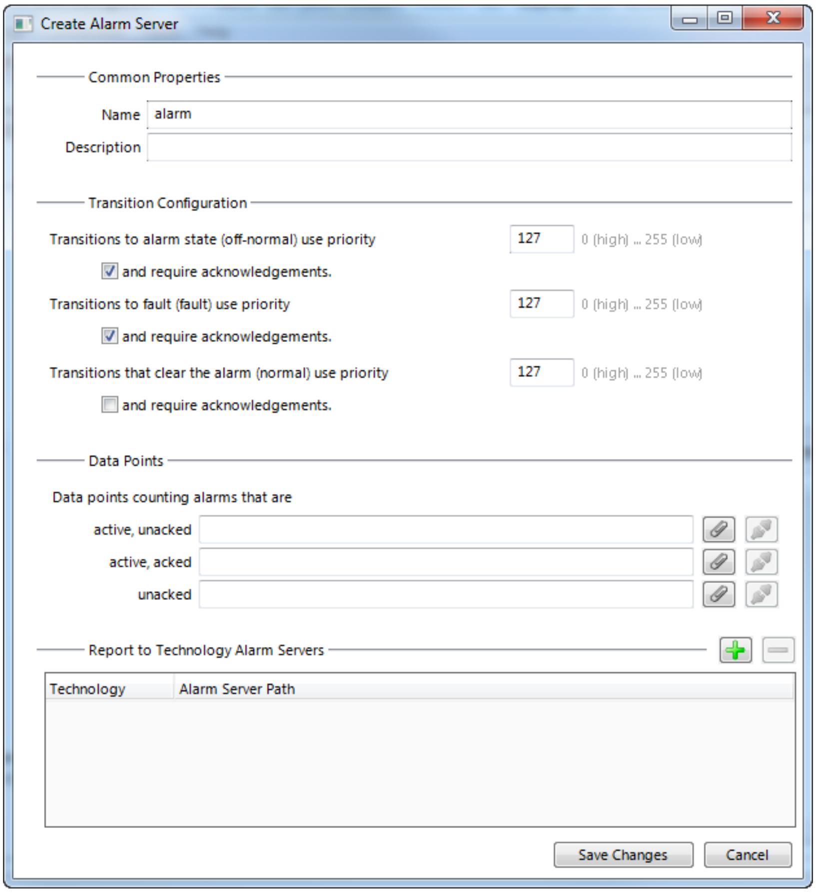


Figure 192: Create New Alarm Server dialog box.

7.12.2 Create an Alarm Condition

To generate alarms from data points, intrinsic reporting is used. For each data point an alarm condition must be defined. This condition employs an intrinsic algorithm to generate alarms based on the data point's value or by evaluating a feedback value. Depending on the data point type (analog, binary, multi-state), different conditions are defined. The alarm is reported to the attached alarm server.

To Create an Intrinsic Alarm Condition

1. Select a data point.

2. Right-click and select **Create Alarm Condition…** from the context menu.

3. For the alarm condition edit the following definitions, which apply to all condition types as shown in Figure 193. Select the **Alarm Server** which the alarm shall be reported to. Typically, you will choose a generic alarm server.

4. Enter an **Alarm Message**. This is shown when the alarm becomes active. You may add variable placeholders to this message by selecting one from the drop-down box **add var** on the right-hand side. Enter a **Clear Message**. This is shown when the alarm clears.

5. Check the option **Enable Fault Alarms**, if fault conditions (offline, unreliable) shall generate fault alarms. If enabled, enter a **Fault Message**, which is displayed along with the fault alarm when it occurs.

6. Optionally, enter a **Time Delay**, for which the condition must persist before the alarm becomes active or is cleared again. The delay is entered in seconds.

7. By clicking you may attach a data point, which is evaluated for enabling the alarm. This can also be done by editing the property relation 'enableAlarm' (see Section 7.7.8). Detach the data point by clicking .

8. Choose the option **value is different from** to define a feedback alarm. In this case the setpoint value of the alarmed data point is compared against the feedback value. A feedback data point can be attached for this purpose. This can also be done by editing the property relation 'feedbackRelation' (see Section 7.7.8).

9. Choose the option **value has condition** to define a value alarm. In this case the data point value is compared against the condition. Edit the condition in the box below this option.

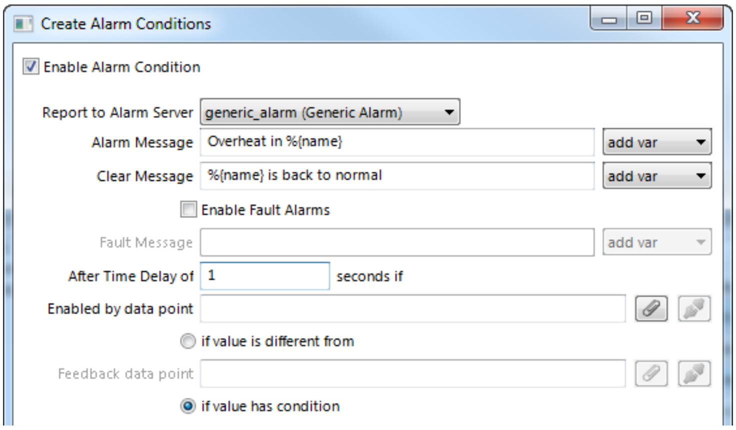


Figure 193: Common settings for an alarm condition.

10. For an analog feedback condition fill in the alarm condition as shown in Figure 194. A feedback alarm is generated, if the setpoint **differs by –** and **differs by +** value from the feedback value. Enter a **Deadband** to account for hysteresis. Attach or detach data points for those limits. This can also be done by editing the property relations 'lowLimit', 'highLimit', and 'deadband', respectively (see Section 7.7.8).

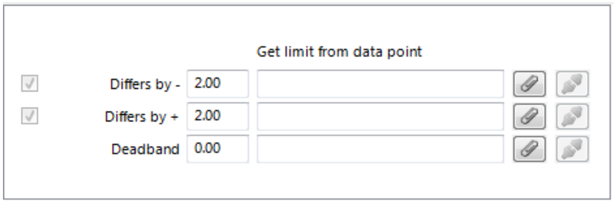


Figure 194: Condition for an Analog Feedback Alarm.

11. For an analog value condition fill in the alarm condition as shown in Figure 195. Select **Low Limit** and **High Limit** and put check marks, if they shall be employed. Enter a **Deadband** to account for hysteresis. Attach or detach data points for those limits. This can also be done by editing the property relations 'lowLimit', 'highLimit', and 'deadband', respectively (see Section 7.7.8).

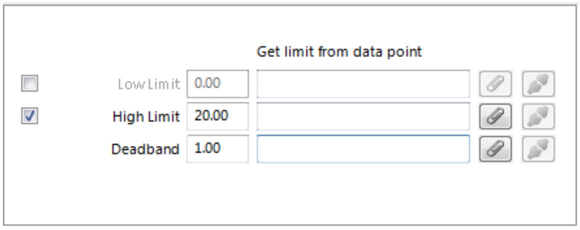


Figure 195: Alarm Condition for an Analog Value Condition.

12. For a binary data point define an alarm value in the alarm condition as shown in Figure 196. Select the **Alarm Value** which triggers the alarm.

Figure 196: Alarm Condition for a Binary Data Point.

13. For a multi-state data point define the alarm values in the alarm condition as shown in Figure 197. Select the states in the list **Not Alarmed** and move them to **Alarm on States** by clicking the arrow buttons.

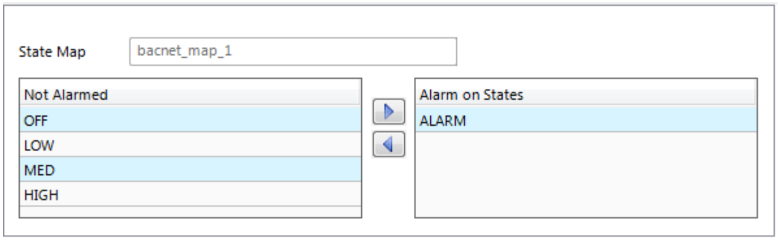


Figure 197: Alarm Condition for a Multi-State Data Point.

14. Click on **Create**. In the alarm column, the alarm sign will be added for those data points that have an alarm condition. If a sub-data point has been alarmed, the top-level data point will indicate this with the sign .

7.12.3 Deliver Alarms via E-mail

Updates in the alarm summary of an alarm object can be used as a trigger to send e-mail. For setting up e-mails, the account information has to be configured on the device, e.g., on the Web UI (see Section 5.2.10). Then an e-mail template can be created and the alarm point attached as a trigger. The e-mail template can be configured to contain certain alarm information in the text or subject field.

To Create an E-mail Template for Alarms

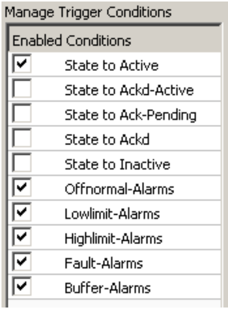
1. Create or configure an e-mail template as described in Section 7.10.1.

2. Change to the **Mail Triggers** tab.

3. Click the **Add…** button and select an alarm data point.

4. In the Mail Triggers list select the added trigger data point.

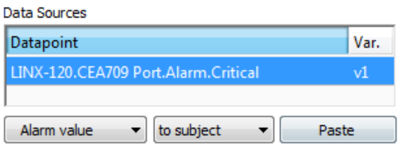
5. In the **Manage Trigger Conditions** list put a check mark on alarm conditions that shall invoke the transmission of the e-mail.



6. Change to the **Common E-Mail Properties** tab.

7. Add the alarm data point as a data source as described in Section 7.10.1.

8. Choose the desired alarm information from the fields in the drop-down **Selected Data Source Value** and paste a place holder into the e-mail text or subject field.



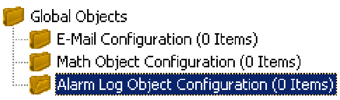
7.12.4 Create an Alarm Log

The alarm objects on the device contain an alarm summary (live list) of currently active and acknowledge-pending alarms. As soon as an alarm becomes inactive and has been acknowledged, it disappears from the alarm summary. To store a historical log of alarm transitions an *alarm log* needs to be created.

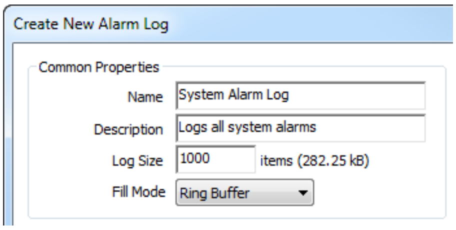
An alarm log can log transitions of one or more alarm objects. Its size is configurable. The alarm log can be operated as ring or as linear buffer. When operated as ring buffer, the oldest alarm log records are overwritten by new alarm transitions, as soon as its size limit is reached. When operated as linear buffer no more alarms are logged once the log is full.

To Create an Alarm Log

1. Under the **Global Objects** folder, select the **Alarm Log Object Configuration** sub-folder.



2. In the data point list right-click and select **New Alarm Log …** from the context menu.

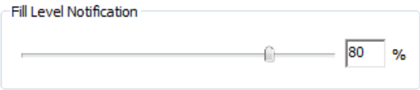


3. In the **New Alarm Log** dialog enter a **Name** for the alarm log. Optionally enter a **Description**.

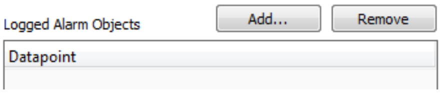
4. Enter a **Log Size**, which defines how many transitions are resident in the alarm log.

5. Select the desired **Fill Mode**. The default is **Ring Buffer**, which lets the log overwrite old records once it reaches its capacity. Select **Linear Buffer**, if recording shall be stopped in this case. The user has to purge the log before it continues recording.

6. Define a percentage for **Fill Level Notification**, which can be used to trigger the transmission of E-Mails.



7. Click on the button **Add…** on top of the **Logged Alarm Objects** list.



8. A data point selector dialog opens. Select one or more alarm objects that shall be logged and click **OK**. The alarm objects appear in the list.

9. Click **Create** to create the alarm log object.

7.12.5 Multi-Edit Alarm Conditions

For editing a large number of alarm conditions, some multi-edit features are available to assist in this task. On a multi-selection of data points, the user can execute two options from the context menu:

- **Configure Alarm Conditions**: Use this option from the context menu on a multi-selection of data points. The alarm condition of all selected data points can be edited. If all selected data points are of the same class (e.g., all analog data points) the alarm condition can be fully specified. Note that these settings will be applied to all data points. In alarm messages use variable place holders. For assigning different limit and enable data points use the manage relations tab (see Section 7.7.8).

- **Configure Alarm Messages**: Use this option from the context menu for editing alarm messages (alarm, clear, fault) for all data points in the selection. A dialog with a spreadsheet view appears as shown in Figure 198. Edit the messages directly in the spreadsheet. Alternatively, click on the export button to export the spreadsheet as a CSV file for editing in Excel. Import a CSV file with alarm messages in the menu **Tools → Import alarm messages**.

Note: *Use the feature to include data points from sub-folders and filter expressions to expand the ability to perform a multi-select.*

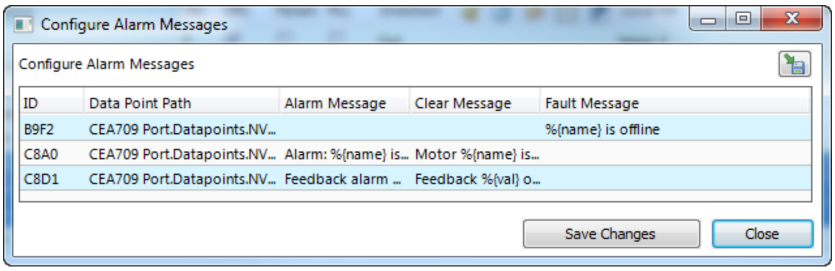


Figure 198: Spreadsheet for multi-edit of alarm messages.

7.13 Local Trending

7.13.1 Create a Local Trend

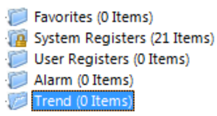
The value of a data point can be logged over time. This is referred to as trend data. To generate trend data a trend object has to be created. The trend data is stored in a data logger file. This file can be downloaded via FTP in binary or CSV format (see Section 8.1.2).

Trend objects can generate trend logs for single and multiple data points and can be operated in one of the following modes:

- **Interval Mode**: In this mode a snapshot of all trended data points is logged into the data logger file. Aligned log intervals can be configured.
- **COV Mode**: In this mode, each of the trended data points is logged separately, if and only if its value changes. For analog data points, a specific COV increment can be configured in the data point configuration properties of the trended data point.
- **Trigger Mode**: In this mode a snapshot of all trended data points is logged each time a trigger condition fires. The trigger condition is applied to a trigger data point.

To Create a Trend Object

1. Select the **Trend** folder of the device.



2. In the data point list right-click and select **New Trend …** from the context menu.

3. In the **Create New Trend Object** dialog (shown in Figure 199) enter a name and optionally a description for the trend log object.

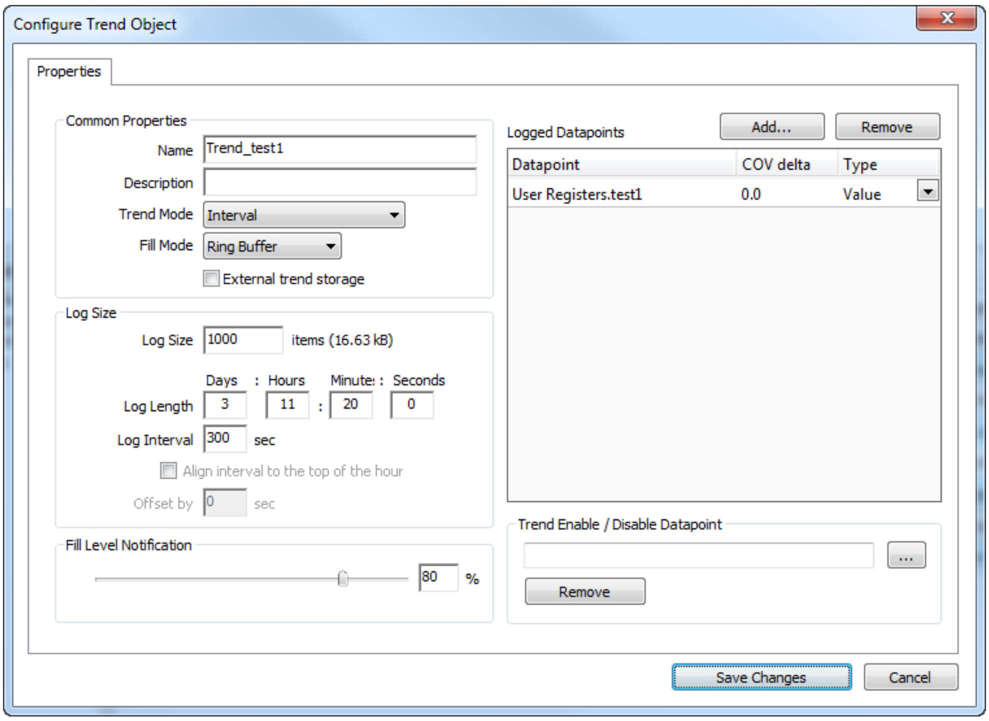


Figure 199: Basic Trend Object Configuration.

4. Select the desired **Trend Mode**.

5. On devices with SD cards, select **External trend storage**, if this trend log shall be backed up to an SD card. If doing so, also set the **Fill Level Notification**, which triggers when a backup is written to the external storage.

6. Select the **Log Size**. The display in the dialog will adapt the estimations for needed data logger file size in KB and duration of the trend log. Alternatively, for interval trends, the estimated log duration and log interval can be edited.

7. In the interval trend mode the **Align interval** option can be activated. Depending on the selected interval, the beginning of the interval is aligned to the wall time (e.g. every 15 minutes aligned to the top of the hour). An additional offset in seconds to that alignment can be specified (e.g. 5 seconds after those 15 minutes).

8. Select a **Fill Level Notification** percentage. This will decide at which fill-level trigger will fire. A fill-level trigger can be used to trigger the transmission of an e-mail (see Section 7.13.5) or a backup of log data to the SD card.

9. Click **Save changes** to store the basic configuration of the trend object. The new trend log object appears in the data point list of the Trend folder.

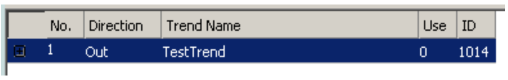
7.13.2 Configure Trended Data Points

When a local trend object has been created, it needs to be configured, which data points it shall log. This is done by attaching data points to the trend object. Only simple data points can be attached for trending, i.e., of class analog, binary, or multi-state. For trend log objects in the BACnet technology, single data points can be attached only.

The trending can be enabled/disabled on behalf of an *enable* data point. This data point should be of type *binary*. If the value of that enable data point is TRUE, the trend object logs data as defined by the trend mode. If the value of the enable is FALSE, trending is disabled. If no enable data point is configured, the trend log is always enabled.

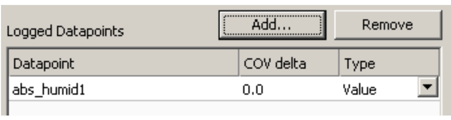
To Attach Data Points for Trending

1. Select the trend object in the **Trend** sub-folder.



2. Right-click and select **Configure Trend** from the context menu. The same dialog which appears when a new trend object is created is shown and allows configuring the trend object. Of course, this step can also be done directly when the object is created.

3. Add data points to be trended. Click on **Add …** which opens a data point selector window.



4. Select the data points and click **OK**. For each of the attached data points, a line appears in the list below the add button. The trended data points will also appear with the trend icon in the data point manager.

Tip! *Data points can also be attached to a trend by selecting a data point in the data point manager, drag it onto a trend object and drop it on the trend object.*

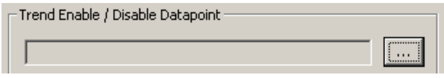
5. Data points can be removed from the trend by clicking **Remove**.

6. If COV mode was selected, the COV increment is displayed in the **COV delta** column. This value can be increased to produce less trend data. Note, that it cannot be lowered under the trended data point's own COV increment. Go to the data point configuration to change the COV increment in this case.

7. If the trended value of the data point shall be aggregated over the log interval, select the desired aggregation in the **Type** column. Available options are **Min**, **Max**, **Avg**.

Tip! *For creating multiple curves with min, average, and maximum values, add the same data point three times and select the different aggregation types.*

8. In addition, a special **Trend Enable** data point can be selected. If configured, the trend log will only log data, if the value of this data point evaluates **true**, i.e., is not zero. Click the ... button to select a data point.



9. To remove the enable data point, click the **Remove** button.

10. When done with the data point setup, click **Save Changes** to leave the dialog.

Tip! *A shortcut to creating a trend log object and attaching a data point is to select a data point in the data point manager, right-click on it and choose **Trend Datapoint** from the context menu. This generates a trend log and links that data point to it.*

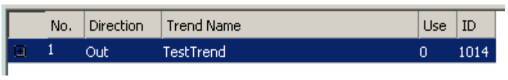
7.13.3 Trend Triggers

Local trend objects in CEA-709 can be operated in *trigger mode*. In this mode, one or more trigger data points cause the generation of a snapshot containing the values of the trended data points at the time instant the trigger is activated. For a trend object, one or more trigger conditions can be defined. Depending on the trigger data point type, the trigger conditions can be refined.

Note, that the behavior of the trigger data point is influenced by the COV properties of the data point. If the **Only notify on COV** property is checked, the data point triggers only if its value changes to the value of the trigger condition. If that property is not checked, the data point triggers on every write with a value that matches the trigger condition.

To Configure Trigger Data Points for Trending

1. Select the trend object in the **Trend** sub-folder.



2. Right-click and select **Configure Trend** from the context menu.

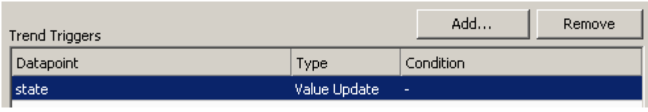
3. Change to the **Triggers** tab.

Note: *Of course, you can also change directly to the **Triggers** tab when creating a trend object.*

4. Click the **Add…** button. A data point selection dialog opens.

5. Select one or more data points and click **OK**.

6. The triggers appear now in the **Trend Triggers** list.



7. In the **Manage Trigger Conditions** you can refine the trigger condition depending on the trigger data point class.

8. When done with the data point setup, click **Save Changes** to leave the dialog.

7.13.4 Download Trend Data in CSV Format

Trend logs can be downloaded from the device via FTP in CSV format (see Section 8.1.2). The CSV contents are generated on-the-fly from the internal binary storage when accessing the file. Each trend log point has one CSV file. The files are located in

`/data/trend/`*TrendLogName_UID*`.csv`

Where *TrendLogName* is the data point name of the trend (Trend Name). The *UID* is the unique ID of the trend log object. The UID can be obtained from the ID column in the data point list of trend log data points as shown in Figure 200. This would result in the trend CSV file '`/data/trend/out_temp_107C.csv`'.

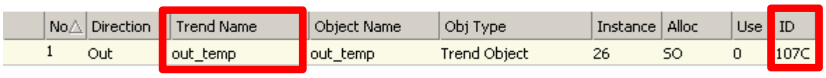


Figure 200: UID of data points.

Because the contents are generated on-the-fly, the file size in the FTP client will appear as 0 Bytes. The decimal point and CSV column separator can be configured in the system configuration of the Web UI (see Section 5.2.1). Note, that for a comma "," as the separator, the decimal point is a point. This is useful for English/U.S. applications. For countries that use the comma as the decimal point, select the semicolon as the CSV separator.

7.13.5 Deliver Trend Data via E-mail

Trend logs can be downloaded from the device via FTP. This requires an active action by the user. Alternatively, trend data can be sent as an e-mail attachment. For doing that, an e-mail template has to be setup for the trend log to be transmitted. The fill-level condition in the trend object can be used as a trigger to send an e-mail with the trend's data logger CSV file as an attachment.

For setting up e-mails, the account information has to be configured on the device, e.g., on the Web UI (see Section 5.2.10). Then an e-mail template can be created and the trend object attached as a trigger.

To Create an E-mail Template for Trends

1. Create or configure an e-mail template as described in Section 7.10.1.

2. Change to the **Mail Triggers** tab.

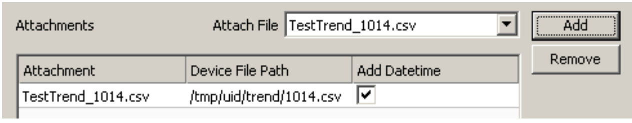
3. Click the **Add…** button and select a trend object.

4. In the **Mail Triggers** list, the added trigger data point appears with the **Fill Level** condition.

5. Change to the **Attachments** tab.

6. Select the trend log CSV file of the trend object in the **Attach File** drop-down box and click **Add**.

Note: *ZIP versions of the CSV files are also available. Select those to save transmission bandwidth and mailbox space.*



7. Click **OK** to complete the e-mail template configuration.

7.13.6 Technology Trends

In the BACnet technology, trend logs can be exposed on the BACnet port via special BACnet Trend Log objects. To create a technology trend select the port folder (e.g., **BACnet Port**) and then the **Trend** folder underneath. Follow the same instructions as described in Section 7.13.1. Please note, that certain restrictions apply to BACnet trends (see Section 6.6.4).

7.14 Math Objects

7.14.1 Create a Math Object

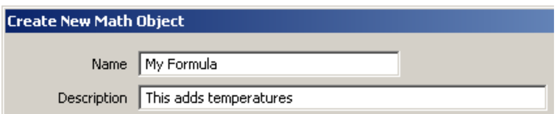
Math objects are advanced application objects that can execute mathematical operations on data points. A math object takes a number of input data points (variables v_1, v_2, …, v_n) and calculates a result value according to a specified formula. When configuring a math object, the input data points, output data points and the formula must be configured by the user. Input data points can be configured with a change-of-value condition, to trigger the math calculation only if the value changes more than a certain delta.

To Create a Math Object

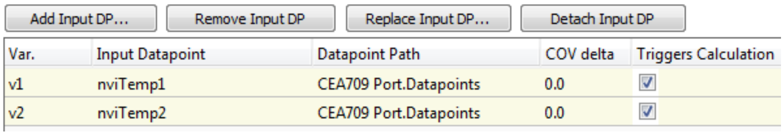
1. Under the **Global Objects** folder, select the **Math Object** sub-folder.

2. Right-click and select **New Math Object …** from the context menu.

3. In the **Create New Math Object** dialog, enter a name and optionally a description for the math object.



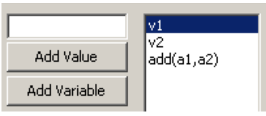
4. Attach input data points by clicking the **Add Input DP** button.



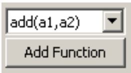
5. In the data point selector dialog, select the input data points and click **OK**. The data points appear as v1, v2, etc.

6. If the data point shall trigger the math calculation only after a certain change-of-value, enter a value into the **COV delta** column. If the data point shall never trigger the math calculation, de-select the **Triggers Calculation** check box.

7. Select the input data point and click **Add Variable** to push the variable on the evaluation stack.



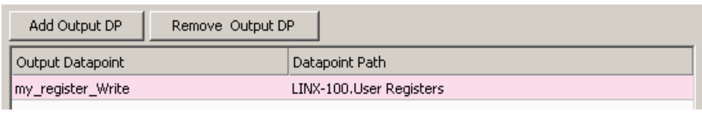
8. Select a function to be applied on the variables and click the **Add Function** button.



9. The resulting formula is displayed at the bottom of the dialog. Alternatively, the formula can be entered there.

Note: *When the formula entered at the bottom is still incomplete and does not yield a meaningful command sequence, the list showing the RPN equivalent will be empty. This allows the user to immediately see if the current input is valid or not.*

10. Add output data points by clicking the **Add Output DP button**.



11. In the data point selector dialog select the output data points and click **OK**.

Note: *On the L-DALI only user registers can be used as output data points of a math object.*

12. To create the math object click **Create**.

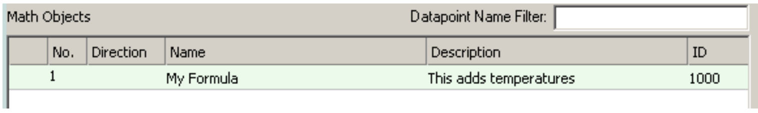
7.14.2 Editing a Math Object

Math objects can be edited once created. The formula can be changed, new variables added, or additional output data points added.

To Edit a Math Object

1. Under the **Global Objects** folder, select the **Math Object** sub-folder.

2. Select the math object in the data point list.



3. Right-click and select **Configure Math Object …** from the context menu.

4. Edit the math object as described in Section 7.14.1.

5. To replace an input data point by another input data point without re-writing the entire formula, click the **Replace Input DP …** button. This opens a data point selector dialog. Select the replacement data point there.

6. To detach an input data point click the **Detach Input DP** button. This leaves the respective variable slot empty.

7. To finalize the edit click on **Save Changes**.

7.15 Historic Filters

7.15.1 Create Historic Filters

Historic filters are used to work with historic values of a base data point. These historic values are derived by defining historic filter functions. An historic filter template is a collection of such filter functions and can be assigned to any analog, binary, or multi-state base data point. For more general information on historic filters refer to Section 6.4.6.

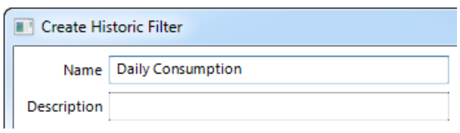
To Create Historic Filters

1. Select one or more data points that shall serve as the base data points.

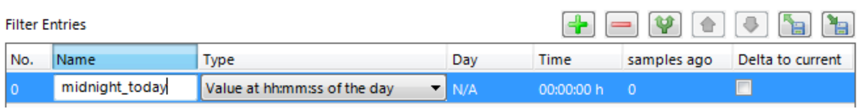
2. Right-click and select the **Configure Historic Filters …** item in the context menu. As an alternative, click on of the **Historic Filter** data point property.

3. The dialog **Select Historic Filter** opens. Click on to create a new one.

4. In the **Create Historic Filter** dialog enter a name and optionally a description.



5. To add a new filter function to the list click on the add button . Enter a filter entry **Name** and choose the desired period **Type**. The name will appear with the historic filter relation and helps identifying it. Depending on the type define the arguments **Day** and **Time**. Select how many **samples ago** the filter goes back. The most current sample is ‘0’, the previous one is ‘1’.



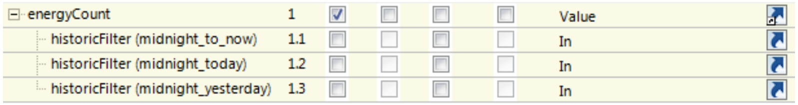
6. To duplicate an entry click on the duplicate button . On the duplicate modify the settings accordingly.

No.	Name	Type	Day	Time	samples ago	Delta to current
0	midnight_today	Value at hh:mm:ss of the day	N/A	00:00:00 h	0	☐
1	midnight_yesterday	Value at hh:mm:ss of the day	N/A	00:00:00 h	1	☐

7. For getting the difference of an historic value to the current value check the box **Delta to current**.

No.	Name	Type	Day	Time	samples ago	Delta to current
0	midnight_to_now	Value at hh:mm:ss of the day	N/A	00:00:00 h	0	☑

8. Click on **Save Changes** and select the created filter template. For each filter entry defined, a historicFilter property relation is created under the base data point(s).



7.15.2 Managing Historic Filter Resources

Historic filters are stored as template types in the project resources. The Configurator provides a resource manager, that can be used to view, edit, import and export historic filter resources. When modifying an historic filter template, all instances that use it are updated.

Select the menu **Tools → Manage Historic Filters …** to open the resource manager dialog. Select a filter type and click the edit button for modifying it. Use the plus button to add new filters and the minus button to delete selected filters. Click the duplicate button to create a duplicate of the selected filter. Click the import button to load historic filters from disk. When importing, filter definitions of the same name are overwritten. Click the export button to store current filters to disk.

7.16 Project Settings

The project settings allow defining certain default behavior and default settings used throughout the project. To access the project settings go to the menu **Settings → Project Settings…**. This opens the project settings dialog, which provides several tabs as described in the following sections.

7.16.1 General

The general tab of the project settings as shown in Figure 201 contains settings independent of the technology port. The settings are:

- **Project Name**: This setting allows entering a descriptive name for the project.
- **Device Configuration Download Default**: This group of settings defines, how the download of device configuration parameters shall proceed. If **Download only data point configuration** is selected, the configuration download does not download anything else than the data point configuration. If **Ask** is selected, the download will pop up a dialog in which the user can choose what to download. If **Download specific** is selected, the project settings of this dialog determine what is downloaded onto the device.

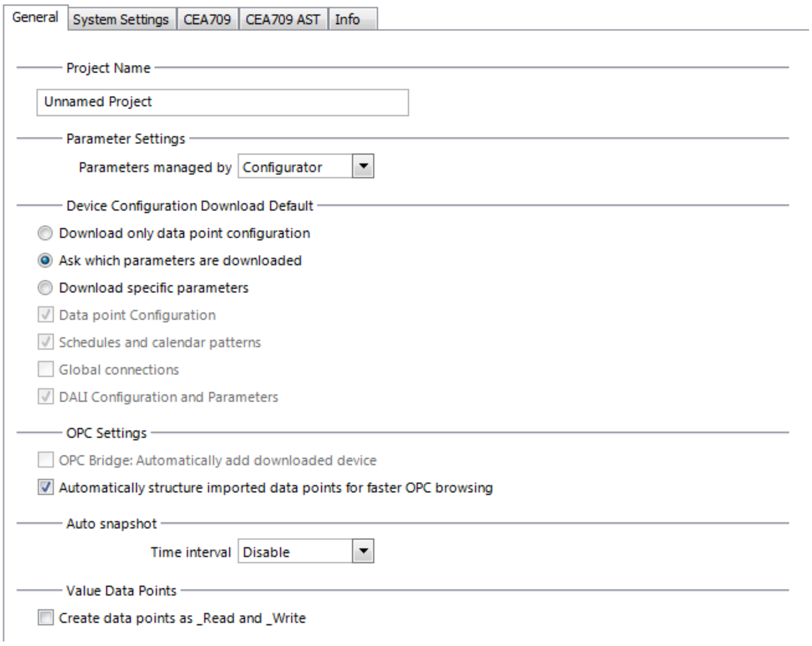


Figure 201: General Project Settings.

- **Automatically structure imported data points for faster OPC browsing**: This option enables the automatic generation of sub-folders when using data points on the device. A sub-folder is created for each scanned or imported device. This allows OPC clients to browse the OPC tags in a hierarchical way.
- **Value Data Points**: Select this option to create data points in the old style with "_Read" (input) and "_Write" (output) data points. Old configurations will have this option set to continue creating data points using the same style. Newly created configurations should not use this setting.

7.16.2 System Settings

This tab is shown in Figure 202. It can be used to configure the device through the Configurator software. In the configuration tree on the left-hand side the user can select certain groups of settings, e.g. Web server settings. The dialog displays the settings of the selected group in the dialog area. The structure is similar to the menu structure on the Web UI.

Enable the legacy mode if this is required by your network management tool (see Section 7.3.4).

Under the port configuration tree, the user can enable or disable communication protocols on the device's ports. Enabled protocols are marked with a checkmark. Click on the checkmark and toggle it. Note, that depending on the device model communication protocols on other ports may be disabled.

The IP address settings cannot be changed in this dialog. The FTP server cannot be disabled in this dialog, either. This ensures that the Configurator can maintain connection to the device.

The **Upload** button can be used to get the current device settings from the device and display them in this dialog. The **Download** button can be used to explicitly transfer the settings from this dialog onto the device.

Important! ***After downloading the device settings from this dialog the changes will be visible immediately on the Web UI but the device needs to be rebooted to let the changes take effect.***

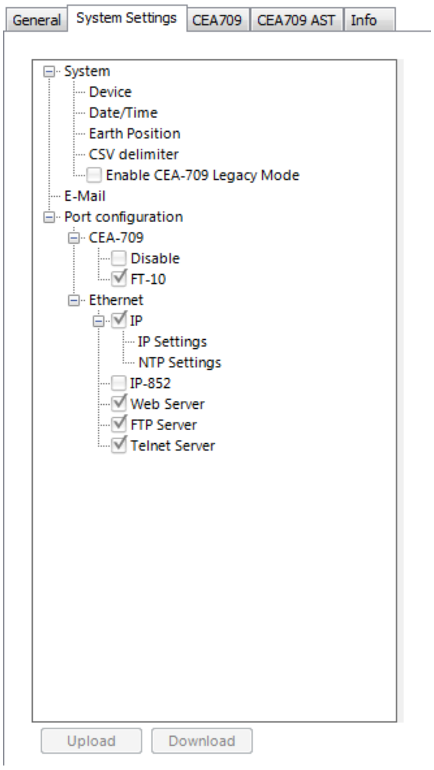


Figure 202: Device Configuration Settings

7.16.3 CEA-709 Settings (LDALI-10X only)

The CEA-709 configuration tab as shown in Figure 203 allows configuring properties of the device's CEA-709 port. The options are:

- **Interface:** This section allows changing the static LONMARK interface of the CEA-709 nodes. On the one hand parts of the interface, which are not required for a specific project (e.g. Constant Light Controller, Sunblind Controller, etc.) can be disabled. On the other hand additional functionality can be enabled (e.g. for emergency light applications or DALI buttons support). When enabling Button objects a template for the LONMARK Button objects created can be selected in the drop down box below. For more information on the LONMARK interface see Section 8.2.

Important! ***Downloading the data point configuration after changing the interface, will result in a new static interface! Thus, it is highly recommended to change this option only if connected via LNS to allow the Configurator to update the LNS database accordingly!***

If using a non-standard interface a XIF-File must be created to be able to add the device in a network management tool (e.g. LonMaker) in case of an offline workflow (see Section 7.7.12).

- **Constant Light Controller automatic bindings**: This section allows enabling and disabling the automatic bindings feature for each channel. See Section 8.2.8.1 for details on the automatic internal binding algorithm.
- **Configuration Download**: This group box contains self-configuration settings for the CEA-709 ports. This is necessary if the device shall be used without being commissioned by a network management tool. For each CEA-709 node the subnet/node and domain information can be configured. Set the check mark and enter the CEA-709 domain and subnet/node information. Note, that the domain must match the nodes' domain on the network and the subnet/node address must not be used by another device.

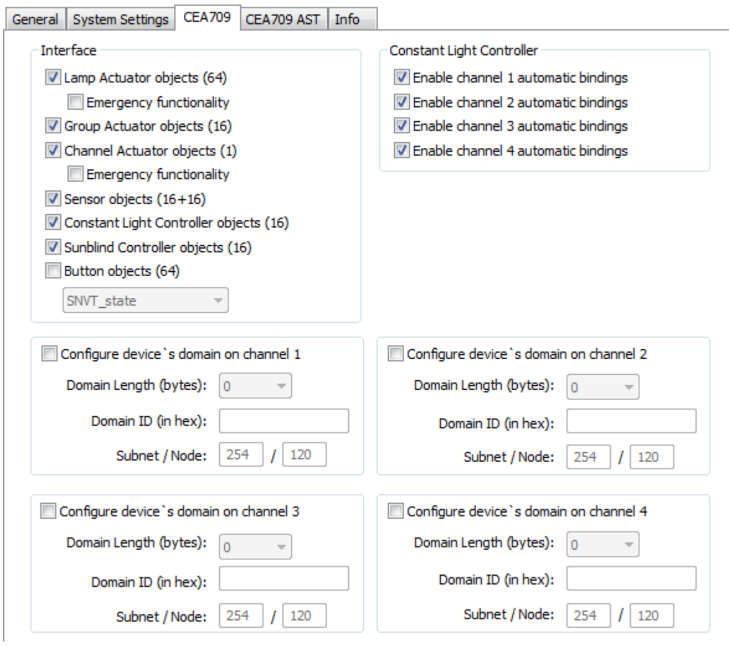


Figure 203: CEA-709 Project Settings.

7.16.4 CEA-709 AST Settings (LDALI-10X only)

For CEA-709 devices, the use of alarming, scheduling, and trending (AST) features requires additional resources (functional objects, CPs and NVs). As changing those resources will in turn change the static interface the L-DALI comes with a preconfigured AST resource configuration, which cannot be changed.

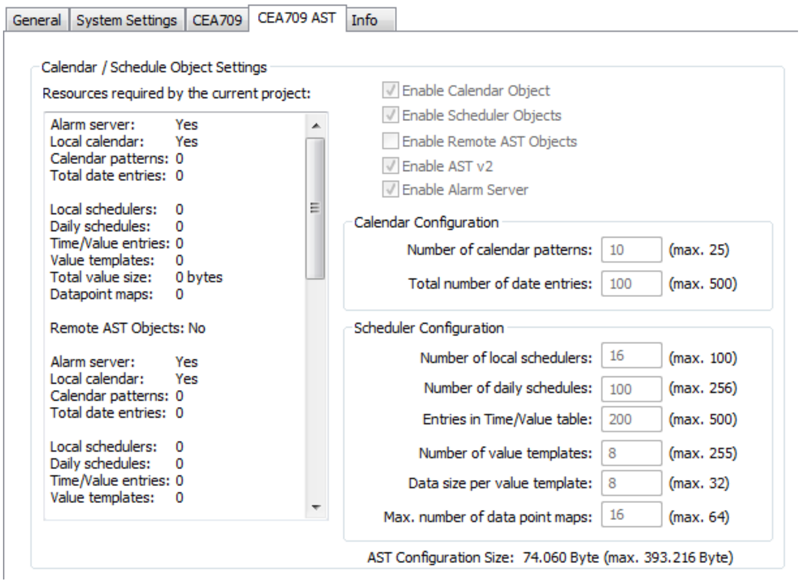


Figure 204: CEA-709 AST Project Settings.

The section **Resources required by the current project** on the left side of the dialog lists the AST resources already used by the current configuration. On the right side the available resources as defined by the static interface of the L-DALI are listed. The lists include the following values:

- **Enable Calendar Object:** This checkbox enables a LONMARK compliant calendar object on the device. It is always enabled.

- **Enable Scheduler Objects:** This checkbox enables local LONMARK compliant scheduler objects on the device. It is always enabled
- **Number of calendar patterns:** Specifies the maximum number of different exception schedules (day classes like holiday, maintenance day) supported by this calendar object.
- **Total number of date entries:** Specifies the maximum number of date definitions which may be stored by the calendar. This is the sum of all date definitions from all calendar entries. A date definition is for example a single date, a date range, or a week and day pattern.
- **Number of local schedulers:** This is the number of local scheduler objects which should be available on the device. Each local scheduler data point created in the data point manager will connect to one of these scheduler objects. There may be more scheduler objects available on the device than are actually used at a certain time. It is a good idea to have some spare scheduler objects ready, in case another scheduler is needed.
- **Number of daily schedules:** This is the maximum number of schedules supported by each scheduler object. This number must at least be 7, since a scheduler always needs to provide one schedule for each day of the week (default weekly schedule). For each special day defined by the calendar or embedded exception day, an additional daily schedule is required to support it.
- **Entries in Time/Value table:** This is the total number of entries in each scheduler defining a value template that should apply on a specific day starting at a specific time (the time table).
- **Number of value templates:** This is the maximum number of value templates supported by each scheduler.
- **Data size per value template:** This specifies the buffer size reserved to hold the data for each value template. More data points or bigger data structures require a bigger value buffer.
- **Max. number of data point maps:** Specifies the maximum number of individual data points that this scheduler is able to control.
- **AST Configuration Size:** This number in Bytes is calculated from the scheduler settings above and represents the total size of the LONMARK configuration properties file stored on the device. While certain settings can be freely edited within the given limits, the resulting configuration size is also limited.

This dialog can be used to determine the resources used and which of the resources – if any – exceeds the corresponding limit.

7.16.5 BACnet Settings (LDALI-20X only)

The BACnet configuration tab as shown in Figure 205 allows configuring properties of the device's BACnet port. The options in the **BACnet Settings** section are:

- **Enable Unsolicited COV**: Put a check mark on this option to enable COV-U on the BACnet port. When active, the device sends unsolicited COV broadcast on all BACnet objects, when their value changes in accordance to the respective COV rules.
- **Always create value objects on auto-create**: If activated, the auto-create BACnet points function of the configuration software creates commandable value objects (AV, BV, MV) instead of output objects (AO, BO, MO) and non-commandable value objects (AV, BV, MV) instead of input objects (AI, BI, MI). This feature can be activated if the regular input/output model is not desired.
- **Use 255.255.255.255 for global broadcast**: This setting overrides the standard behavior of BACnet to send broadcasts as global IP broadcasts. This can solve scanning problems with some BACnet devices.

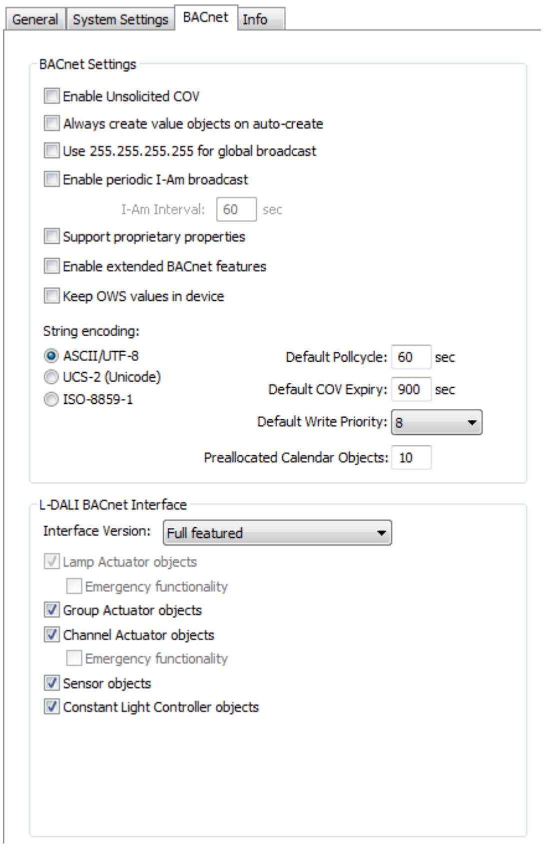


Figure 205: BACnet Project Settings.

- **Enable periodic I-Am broadcast**: This setting enables the periodic transmission of I-Am broadcasts. Specify the interval in seconds. If disabled, the device sends an I-Am only when starting up. This is the default behavior of BACnet devices.
- **Support proprietary properties**: Check this option if a scan on a remote device shall find proprietary properties in addition to the standard properties of supported objects.
- **Enable extended BACnet features**: Check this option to enable additional properties in BACnet server objects. This affects Elapsed_Active_Time, etc. properties in binary objects, custom properties in scheduler objects (value labels).
- **Keep OWS values in device**: Check this option, if BACnet properties changed by the OWS shall be maintained in the device even after a new configuration download. Without this option, a new configuration will overwrite any changes made by the OWS with the values defines in the configuration (e.g., high and low limits of alarm conditions). The default is to overwrite with configuration values.
- **String encoding**: This setting defines, how strings in BACnet objects are encoded. By default it is ASCII/UTF-8, which is compatible with most BACnet software. To support characters of Western European languages, choose ISO-8859-1. To support Unicode character sets (e.g., Japanese) select UCS-2.
- **Default Poll cycle, Default COV Expiry, Default Write Priority**: This setting defines the default values that are used when creating new client mappings. Changing this option does not affect already existing client mappings. The default write priority is also used when writing to commandable server objects.
- **Preallocated Calendar Objects**: This setting defines how many BACnet calendar objects shall be created as a default. These are filled up with calendar patterns as they are defined.

The option in the **L-DALI BACnet Interface** section are:

- **Interface Version:** This setting determines the scheme used for the BACnet object IDs and the BACnet objects available (see Section 8.3.1).
- **Interface:** This section allows configuring with function is available via BACnet objects. One the one hand parts of the interface, which are not required for a specific project (e.g. Constant Light Controller, etc.) can be disabled. On the other hand additional functionality can be enabled (e.g. for emergency light applications or DALI buttons support). See Section 8.3.

7.16.6 Project Information

This tab is shown in Figure 206. It provides fields to enter additional information about the project such as author name and a reference field. The comments text area allows entering free text describing the project. For instance it can be used to document a revision history. The fields **Last Saved** and **Configurator Version** are filled in when saving the project. When creating a new project without having it saved for the first time, those fields are empty.

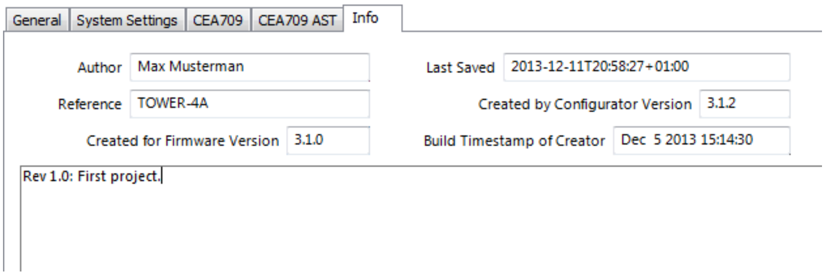


Figure 206: Info tab in project setting.

8 Operating Interfaces

8.1 Common Interface

8.1.1 Schedule and Calendar XML Files

The daily schedule and calendar pattern configuration can be changes at run-time over the Web UI or the network. An alternate way to change that configuration is to download a schedule and calendar XML file via FTP onto the device. After the file has been downloaded, the new configuration becomes effective immediately. The device does not need to be rebooted. The files are located in

```
/tmp/uid/sched/UID.xml
/tmp/uid/cal/UID.xml
```

The *UID* is the unique ID of the data point. The UID can be obtained from the ID column in the data point list as shown in Figure 200. A schedule data point with UID 107C would result in the schedule XML file '`/tmp/uid/sched/107C.xml`'. The UID remains constant for the life time of the data point even when the name or description is changed.

The content of the XML file must be compliant to the scheduleCfg schema. This schema can be found at the LOYTEC Web site. The XML documents can refer to the target namespace 'http://www.loytec.com/xsd/scheduleCfg/1.0/'.

8.1.2 Trend Log CSV File

The CSV file format for a trend log and the location of those files are defined in this section. The trend log CSV files are accessible either via their UID only, or in combination with contents of the trend log object name. The files are located in

```
/tmp/uid/trend/UID.csv
/data/trend/Datapointname_UID.csv
```

The *UID* is the unique ID of the data point. The UID can be obtained from the ID column in the data point list as shown in Figure 200. For a more user-friendly listing of the files, the *Datapointname* contains the trend log's object name. It is truncated after 23 ASCII characters to fit the requirements of the file system. A trend CSV file for the trend object 'trend0' and the UID '107C' would result in the CSV file '`/data/trend/trend0_107C.csv`'. The UID remains constant for the life time of the object even when the name is changed.

The CSV file format for a trend log is defined in this section. The CSV file starts with a header, containing at least the first line, which specifies the CSV format (log_csv_ver). The current version is 2. The next line contains the field log_device. It has trailing fields that specify the vendor, product code, firmware version and device ID string. The Device ID String can be one of the following: (IP) 192.168.24.100, (BACnet Device) 224100, (CEA-709 NID) NID.

The log_info line specifies the fields UID and name of the trend log object. The line log_create has two fields specifying the date and time when this CSV log was generated. The line log_capacity has two fields: the current number of log entries in the file and the log capacity.

Following are one or more lines of log_item. Each line specifies a trended data point. The first field is the index, the second the ID of the logged data point, the third the data point name. The data point name can be augmented by engineering units in square brackets. Log entries in the CSV refer to the item index to identify the data point, for which the entry was logged.

```
#log_csv_ver;2
#log_device;LOYTEC;Product Code;Firmware Version;Device ID String; Serial No
#log_info;Log-ID;Log Name
#log_create;YYY-MM-DD;HH:MM:SS
#log_capacity;filled;capacity
#log_item;index;UID;data point name [units]
```

After those lines any number of comment lines starting with a hash character ‘#’ are allowed. One line contains the column headings. Lines that are not comments specify one log record per line, using the column information as described below. The columns are separated by commas ‘,’ or semi-colons ‘;’. If commas are used as a separator, the decimal point must be a point ‘.’. If semi-colons are used, the decimal point must be a comma ‘,’.

There are as many value columns as value sources specified in the header. If at a given date/time more values are logged, all of them appear in the same line. If at that given time some sources did not log values, those columns are left empty.

Column	Field	Example	Description
A	Sequence Number	50	The log record sequence number. This is the monotonously increasing sequence number, which is unique for each log record.
B	Source	0	Data point source identifier. Indexes into logger_entry header. For value lines in a multi-column CSV, this field indexes the first column, which has a value. For the ERROR record type, the field indexes the data source that caused the error. For LOGSTATE, TIMECHANGE records this field is not applicable and can be left at zero.
C	Record Type	2	The record type: LOGSTATE (0), BOOL (1), REAL (2), ENUM (3), UNSIGNED (4), SIGNED (5), NULL (7), ERROR (8), TIMECHANGE (9)
D	Error/Time Change/Log Status	1	This field is valid for records of type ERROR, TIMECHANGE, and LOGSTATUS.
E	Date/Time	2007-11-02 15:34:22	The date/time of the log record. This is in the format YYYY-MM-DD HH:MM:SS.
F	Value 0	24,5	Logged value from source 0 or empty
G	Value 1	200	Logged value from source 1 or empty
…	…		
…	Value *n* – 1	5000	Logged value from source *n* – 1 or empty

Table 21: Columns of the Trend Log CSV File

There are as many value columns as value sources specified in the header. If at a given date/time more values are logged, all of them appear in the same line. If at that given time some sources did not log values, those columns are left empty. The “Source” column in a multi-value CSV refers to the first data source that supplied a value in a given line.

8.1.3 Alarm Log CSV File

The historical alarm logs are also accessible as CSV-formatted files. The alarm log CSV files are accessible either via their UID only, or in combination with contents of the alarm log object name. The files are located in

```
/tmp/uid/allog/UID.csv
/data/allog/Alarmlogname_UID.csv
```

The *UID* is the unique ID of the alarm log object. The UID can be obtained from the ID column in the data point list of the alarm log folder, similar to obtaining the UID of trend log objects. For a more user-friendly listing of the files, the *Alarmlogname* contains the alarm log's object name. It is truncated after 23 ASCII characters to fit the requirements of the file system. A trend CSV file for the alarm log object 'alarmlog0' and the UID '100C' would result in the CSV file '`/data/allog/alarmlog0_100C.csv`'. The UID remains constant for the life time of the object even when the name is changed.

The CSV format of the alarm log CSV file is identical to the trend log CSV format as described in Section 8.1.2.

8.1.4 Emergency Light Test Log CSV File

When an emergency light test is performed the results are logged. These logs are available as CSV-formatted files. There is a log file for each group and one for each channel, which contains entries from emergency lights not assigned to a group. The files are located in

```
/tmp/app/grp<chnl><grp>_emerg_tst.csv
/tmp/app/chl<chnl>_emerg_txt.csv
```

Example: */tmp/app/grp104_emerg_tst.csv* contains the log information generated by all emergency lights assigned to group 4 on channel 1, */tmp/app/chl1_emerg_tst.csv* contains the log information generated by all emergency lights on channel 1 assigned to no group.

The logs contain information on all tests, no matter whether they were triggered using the *nviEmergTest* (see Section 8.2.5.6), the Web Interface (see Section 5.3.4.4), or it was automatically started by the ballast itself due the configured auto-test calendar (see Section 5.3.4.7).

Logs can be downloaded using FTP or attached to e-mail templates (see Section 7.10.3).

8.2 CEA-709 Interface (LDALI-10X only)

Depending on the L-DALI device type 1 (LDALI-3E101/LDALI-E101-U/LDALI-3101-U), 2 (LDALI-3E102), or 4 (LDALI-3E104) DALI channels are supported. Each DALI channel is implemented as a separated CEA-709 node. Every one of these up to four CEA-709 nodes is a full grown CEA-709 node, including its own unique node id ("Neuron ID"), network address, network variable tables etc. However, all these nodes share one common physical network connection (see Section 4.9.2).

All configuration properties listed in this section are implemented as Configuration Parameters using configuration files (CPT, CPV, and CPC files). File access is provided via LONMARK FTP.

The L-DALI can be used with different static network interface configurations. Depending on the application the different object types can be enabled or disabled to add functionality (e.g. button support or additional network variables for applications with emergency lights) or to reduce the number of objects, network variables and configuration properties respectively.

The interface configuration can be chosen via the Project Settings in the Configurator Software (see Section 7.16.2).

8.2.1 Node Object #0000

The L-DALI provides a node object conforming to the LONMARK guidelines for each DALI channel.

- The Node Object accepts the following commands via *nviRequest*: RQ_NORMAL, RQ_UPDATE_STATUS, RQ_REPORT_MASK, RQ_ENABLE, RQ_DISABLE, RQ_UPDATE_ALARM, RQ_CLEAR_ALARM, RQ_RESET, RQ_CLEAR_RESET
- LONMARK alarming is supported via *nvoAlarm* (SNVT_alarm) and *nvoAlarm_2* (SNVT_alarm_2). This allows devices supporting the LONMARK alarm notifier profile to receive alarms generated by the L-DALI and react with a defined action (e.g., send an email). By supporting both alarm SNVTs, SNVT_alarm and SNVT_alarm_2, legacy and state-of-the-art alarm handling is supported.
- nviDateEvent (*SNVT_date_event*), nvoDateResync (*SNVT_switch*): If not bound, the local calendar is used. If a global calendar shall be used, both of these NVs must be bound to the respective NVs of the global calendar object.
- nviTimeSet (*SNVT_time_stamp*): When writing to this NV, the system is set, if the configure time-source is “LONMARK” or “Auto” (see Section 5.2.11). The time value is interpreted as local time.
- nvoSystemTemp (*SNVT_temp*): This NV can be used to poll the system temperature of the L-DALI. It does not send updates and must be polled.
- nvoSupplyVolt (*SNVT_volt*): This NV can be used to poll the supply voltage of the L-DALI. It does not send updates and must be polled.
- nvoIpAddress (*SNVT_str_asc*): This NV can be used to poll the IP address of the L-DALI. It does not send updates.
- nciEarthPos (*SNVT_earth_pos*): This configuration property can be used to set the earth position of the L-DALI. It has been implemented as an NV to make other devices send that configuration to the L-DALI over the network (e.g., from a GPS device).

This object type cannot be disabled.

8.2.2 Real-Time Keeper Object #3300

The L-DALI includes one standard LONMARK real-time keeper object for each DALI channel. The Real-Time Keeper Object is used to synchronize the system time of multiple LONMARK compliant devices.

The object has the following network variables:

- nvoTimeDate (*SNVT_time_stamp*): Propagates the devices current system time and date (local time). It is typically bound to the nviTimeSet input network variable of the node objects of the LONMARK compliant devices, which are synchronized with the system time of the L-DALI. The update rate of the nvoTimeDate can be configured using the configuration property SCPTupdateRate (default every 60 seconds).

This object type cannot be disabled.

8.2.3 Calendar Object #0006

The L-DALI includes one standard LONMARK calendar object for each DALI channel.

This object type cannot be disabled.

8.2.4 Scheduler Object #0007

The L-DALI includes 16 standard LONMARK scheduler objects for each DALI channel.

This object type cannot be disabled.

8.2.5 Lamp actuator Object #3040

There are three different types of the lamp actuator objects:

- Lamp actuator object: related to DALI lamp (64 per channel)
- Group actuator object: related to DALI group (16 per channel).
- Channel actuator object: related to the DALI channel (1 per channel).

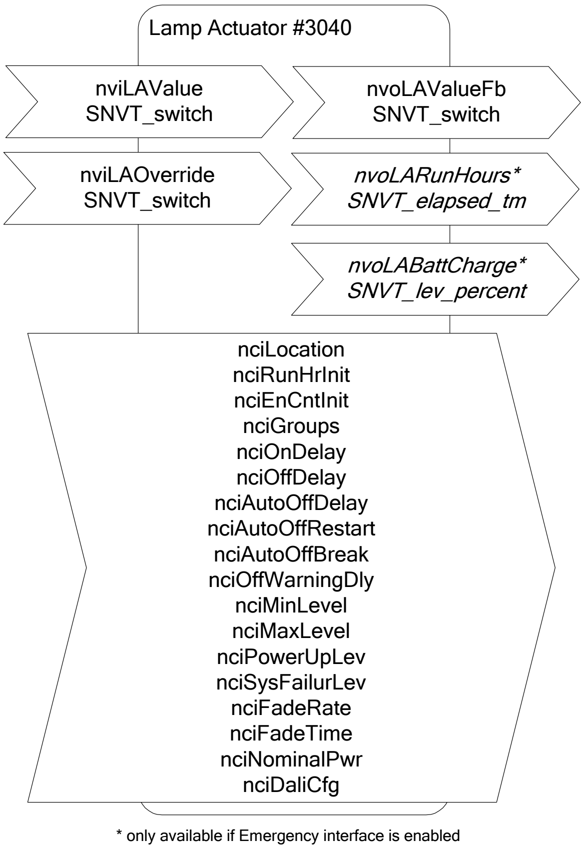


Figure 207: Lamp Actuator Object

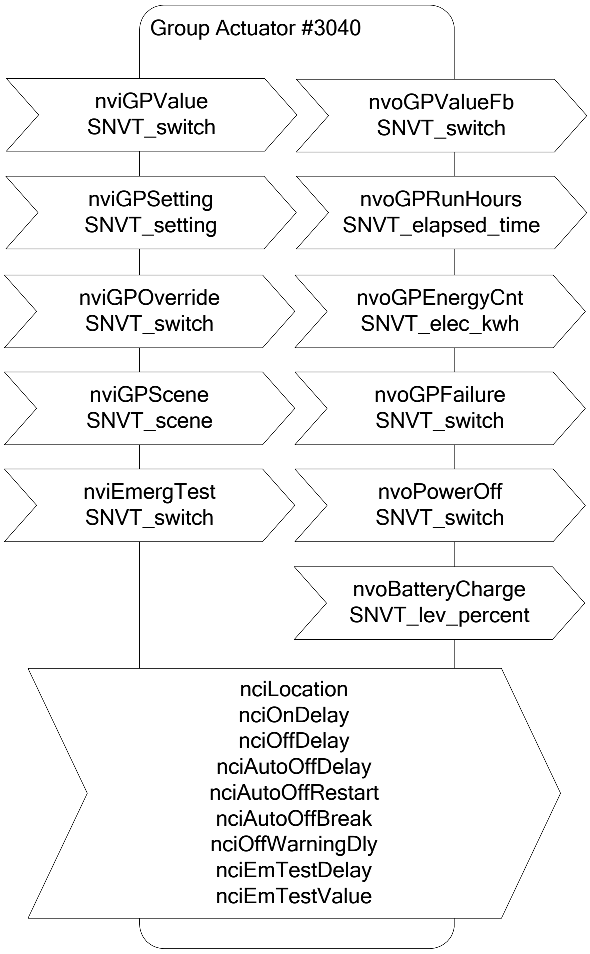


Figure 208: Group Actuator Object

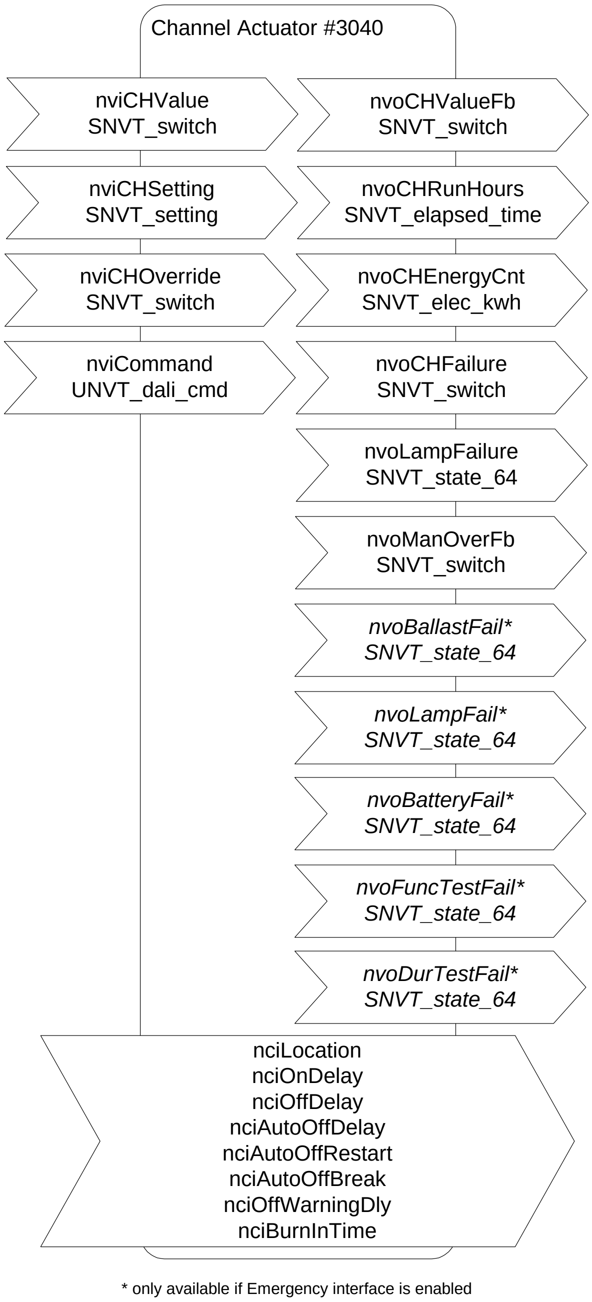


Figure 209: Object for channel

These object types are enabled in the default interface. They can be disabled.

Note:	*If the Lamp Actuator objects are disabled, the functional blocks are still present, but only contain the configuration properties corresponding to the DALI parameters of the corresponding ballast.*

8.2.5.1 Control

The L-DALI offers different methods to control lamps. All network variables described below have the same priority. This means that the last network variable update determines the state of the lamp.

Individual Control

Each lamp can be controlled individually via the switch input network variable *nviLAValue* of the lamp actuator object. Any new value received is transmitting to the corresponding DALI lamp in fading mode (configuration property *nciFadeTime* is used).

Group Control

Lamps belonging to a group can be controlled together by the following network variables of the group actuator object.

- ***nviGPValue*:** This network variable has the same functionality as *nviLAValue* but controls a whole group. Any new value received is transmitting to the DALI lamps in the group using fading mode (configuration property *nciFadeTime* is used).

- ***nviGPSetting*:** Using this input network variable of type SNVT_setting dimming devices can change the brightness relatively. By *function* = SET_UP or *function* = SET_DOWN the light is dimmed relatively by the amount defined in the corresponding *setting* field. By *function* = SET_STOP ongoing fades are stopped. When the light is switched off by *function* = SET_OFF, the last light level is saved temporarily. When the light is switched on again (function = SET_ON) the remembered light level is restored. An absolute light level can be set by *function* = SET_STATE. The corresponding *setting* field defines the target light level. The light value is changed in ramping mode (configuration property *nciFadeRate* is used).

- ***nviGPScene*:** Using this input network variable of type SNVT_scene up to 16 DALI scenes can be saved and recalled. The command SC_LEARN stores the current light levels of all lamps of the group in the specified scene number. By a SC_RECALL command previously stored scenes can be recalled. Scenes can be deleted by a SC_RESET command. When a new scene is selected the lamp values are adjusted in fading mode (configuration property *nciFadeTime* is used).

Channel Control

All lamps on the same DALI channel can be controlled together by the following network variables of the channel actuator object.

- ***nviCHValue*:** This input network variable works in the same way as *nviLAvalue* and *nviGPvalue* but affects all lamps on a channel.

- ***nviCHSetting*:** This input network variable works in the same way as *nviGPSetting* but affects all lamps on a channel.

8.2.5.2 Maximum and Minimum Light Level

For a dimmable lamp the maximum and minimum light level of a lamp can be configured via the configuration properties *nciMinLevel* and *nciMaxLevel* of type SNVT_switch.

For a non-dimmable lamp set *nciMinLevel* = *nciMaxLevel* = 100%.

8.2.5.3 Timing Parameters

Figure 210 shows the behavior of the lamp actuator if the light is switched on/off via one of the switch (*nviLAValue, nviGAValue, nviCHValue*) or via one of the setting input network variables (*nviGASetting, nviCHSetting*). When the ON command is received, the lamp is switched to the specified value after the time *nciOnDelay* has expired. When the lamp is already on and a new ON command is received the lamp is switched to the new value immediately. If the lamp is on and an OFF command is received the lamp is switched off after the time *nciOffDelay*.

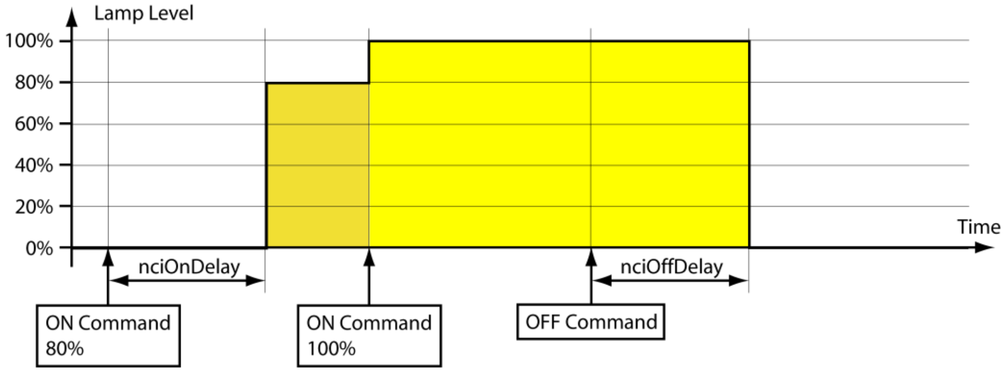


Figure 210: Timing Parameters

For staircase lighting an automatic cutoff can be configured in *nciAutoOffDelay* as shown in Figure 211. When this time expires, the lamp switches off automatically. To warn the user that the light is going to turn off, an off-warning delay (*nciOffWarningDly*) can be configured. During this time the light is dimmed to 50% of the previous level. For non-dimmable lights (*nciMinLevel*=*nciMaxLevel*) the light is blinking.

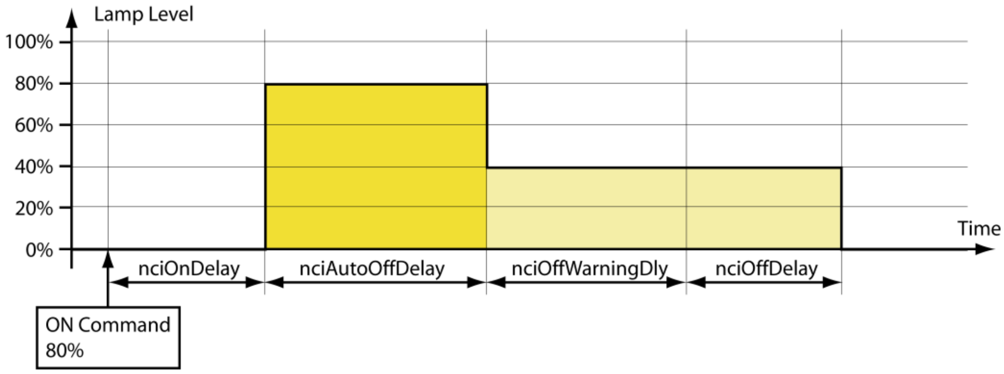


Figure 211: Timing Parameters with *nciAutoOffDelay* and *nciOffWarningDly*

The behavior of the auto-off timer can be modified with the two configuration properties *nciAutoOffRestart* and *nciAutoOffBreak*. If *nciAutoOffRestart* is set to ST_ON, the auto-off timer can be restarted by a new ON command (see Figure 212). If *nciAutoOffBreak* is set to ST_ON the auto-off timer can be stopped before the time has expired by switching the light off via one of the applicable input network variables (see Figure 213).

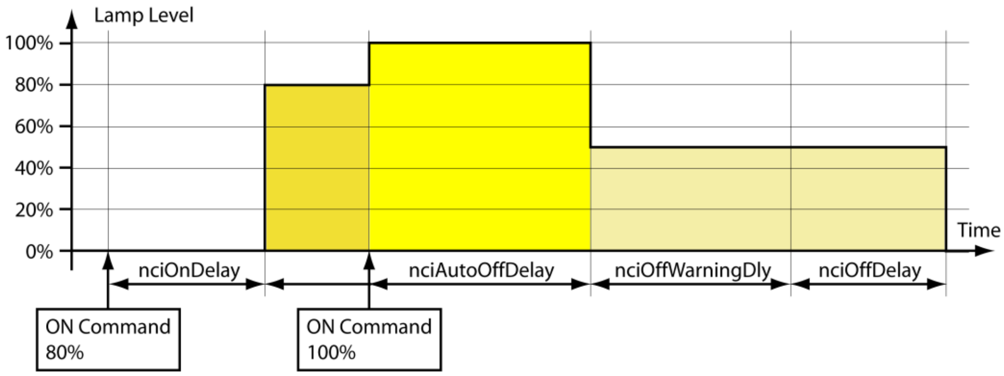


Figure 212: Timing Parameters (*nciAutoOffRestart*=ST_ON)

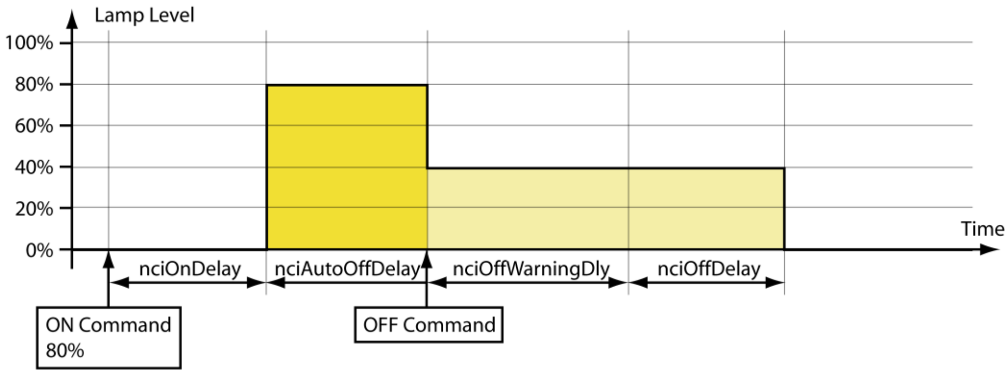


Figure 213: Timing Parameters (*nciAutoOffBreak*=ST_ON)

8.2.5.4 Prioritized Control

Lamp values can be overridden by the following network variables:

- ***nviLAOverride****:* This input network variable of type SNVT_switch overrides the value of *nviLAValue*.
- ***nviGPOverride****:* This input network variable of type SNVT_switch overrides the value of *nviGPValue*, *nviGPSetting*, and *nviGPScene*.
- ***nviCHOverride****:* This input network variable of type SNVT_switch overrides the value of *nviCHValue* and *nviCHSetting*.

If *nviXXOverride* is valid (.state=0/1) it controls the lamp/group/channel output. Updates of on the non-priorized input network variables (see Section 8.2.5.1) are ignored. If *nviXXOverride* is set to invalid (.state=-1) control of the lamp/group/channel output is returned to the non-priorized input network variable.

The light value is changed in fading mode (configuration property *nciFadeTime* is used).

Prioritized control commands are not affected by the timing parameters described in Section 8.2.5.3.

8.2.5.5 Feedback

The lamp object, group object and channel object each report the current state via a feedback variable of type SNVT_switch:

- *nvoLAValueFb*: lamp object feedback
- *nvoGPValueFb*: group object feedback
- *nvoCHValueFb*: channel object feedback

The channel object reports if it is in manual override mode via the network variable *nvoManOverFb*.

8.2.5.6 Emergency Lights

The L-DALI supports testing of emergency lights using its CEA-709 interface. The test is started and stopped by using the network variable input *nviEmergTest*, which is available in each Group Actuator object. To start the test the *state* part of the *SNVT_switch* must be set to SW_ON (1). The *value* part selects one of the following tests:

- **Function test** (*value* = 1%): Starts the function test of all DALI emergency lights in this group, which support this function. For details on the DALI emergency lighting function test see IEC 62386-202. If the Emergency interface is enabled the *nvoFuncTestFail* will show whether the last function test performed has failed (see below).
- **Duration test** (*value* = 2 %): Starts the duration test of all DALI emergency lights in this group, which support this function. For details on the DALI emergency lighting duration test see IEC 62386-202. If the Emergency interface is enabled the *nvoDurTestFail* will show whether the last duration test performed has failed (see below).
- **Mains test** (*value* = 100%): Typically used in a central battery system, to verify lights are dimmed to the emergency level when mains fail. When the test is started via *nviEmergTest*, the power of the emergency lamps is cut by setting the switch network variable output *nvoPowerOff* to ON. After the delays *nciEmTestDelay*, the value of all lamps in the group is compared to the reference value specified in *nciEmTestValue*. If a lamp does not reach the reference value (dim value above or below reference value) or if a lamp is offline, an alarm is generated as shown in Figure 214.

Test results will be stored in the appropriate emergency light test log (see Section 8.1.4). To abort any active test the *SNVT_switch* must be set to {SW_OFF, 0%}. To periodically execute tests a scheduler controlling *nviEmergTest* can be configured. See Section 7.11.3 on how to configure a scheduler.

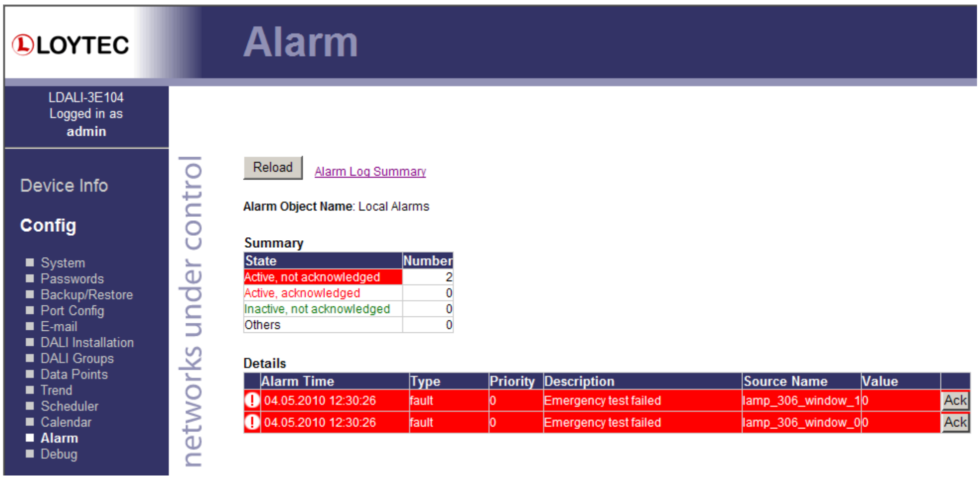


Figure 214: Emergency Light Test Alarm

If the interface configuration **Emergency** is selected the following additional, emergency light related network variables are available:

- *nvoLARunHour* (Lamp Actuator): Report run-hours per ballast.
- *nvoLABattCharge* (Lamp Actuator): Report battery charge per emergency light ballast (if ballast supports it).
- *nvoBatteryFail* (Channel Actuator): Report battery failure for each emergency light ballast.
- *nvoFuncTestFail* (Channel Actuator): Report whether last function test failed for each emergency light ballast.
- *nvoDurTestFail* (Channel Actuator): Report whether last function test failed for each emergency light ballast.

8.2.5.7 Status Monitoring

The L-DALI monitors the DALI lamps and reports any failure in the output network variables *nvoGPFailure* (group object) and *nvoCHFailure* (channel object).

A lamp failure is reported if either

- bus power for the related DALI channel fails,
- the DALI ballast is not reachable via the DALI channel, or
- the DALI ballast reports a failure (e.g. lamp failure) in its DALI status register.

The value of the network variable *nvoGPFailure* of the group object reflects the percentage of failed ballasts in this group.

The value of the network variable *nvoCHFailure* of the channel object reflects the percentage of failed ballasts in this channel.

The network variable *nvoLampFailure* of type SNVT_state_64 reports the status of the all lamps in a channel. Each bit corresponds to the lamp with the same index. In case of a failure the bit is set.

8.2.5.8 Statistics

The L-DALI reports the run hours and energy consumptions of the DALI lamps.

Run Hours

The L-DALI determines how long each DALI lamp was switched on. If a lamp is replaced the run hours can be initialized via the configuration property *nciRunHrInit* of the lamp object. To limit the number of network variables the run hours are reported not separately for each lamp but for a whole group or channel.

The group object reports the maximum run hours of all lamps in the groups via the output network variable *nvoGPRunHours*.

The channel object reports the maximum run hours of all lamps on the channel via the output network variable *nvoCHRunHours*.

Energy Consumption

The L-DALI device can calculate the energy consumption of the DALI lamps. For that purpose it needs to know the nominal power of all connected DALI lamps. If the configuration property *nciNominalPwr* of the lamp object is set to 0 (AUTO), the L-DALI tries to obtain the energy consumption from the DALI device. However, not all DALI ballasts support this proprietary extension. In this case the *nciNominalPwr* has to be configured with the nominal power of the ballast. The energy consumption can be initialized via the configuration property *nciEnrgyCntInit* of the lamp object.

To reduce the number of network variables the energy consumption is reported not separately for each lamp but for a whole group or channel. The group object reports the sum of energy consumed by all lamps in the groups via the output network variable *nvoGPEnergyCnt*. Equally the network variable *nvoCHEnergyCnt* reports the sum of energy consumed by all lamps on the channel.

8.2.5.9 Burn-In Function

Some lamps require a burn-in time during which they must not be dimmed. The burn-in mode can be activated by setting the *nviCommand* network variable of the channel object to START_BURN_IN. The burn-in time is defined by the configuration property *nciBurnInTime*. During this time the lamps will only be switched to on (100%) or off (0%) but not dimmed. To abort the burn-in mode prematurely *nviCommand* can be set to STOP_BURN_IN.

8.2.5.10 Fail Safe Functions

The light level a lamp adopts after power-up is defined in the configuration property *nciPowerUpLev*.

The light level a lamp adopts in case of a DALI system failure is defined in the configuration property *nciSysFailureLev*.

8.2.5.11 Colour Control

Controlling the colour of devices of type “colour control” (device type 8) is possible via the *nviCommand* available on the Channel actuator object.

8.2.5.12 Input Network Variables

nviXXValue	
Type	SNVT_switch
Valid Range	.value: 0..100% .state: 0, 1 On: .state = 1 and .value > 0 Off: .state = 0 and .value=any or .state=1 and .value = 0
Default Value	-
Description	Used to control the lamp/group/channel. This input is effective only if *nviXXOverride.state* is invalid (-1). If an update is received on this network variable the corresponding DALI lamp/group/channel is dimmed in fading mode (constant time, configuration property *nciFadeTime* is used). Lamp actuator only: If an invalid value is received, the lamp adopts the level defined by the *nciSysFailurLev*.

<table>
<tr><th colspan="2">nviXXSetting</th></tr>
<tr><td>Type</td><td>SNVT_setting</td></tr>
<tr><td>Valid Range</td><td>.function: SET_OFF, SET_ON, SET_UP, SET_DOWN, SET_STATE
.setting: 0..100%</td></tr>
<tr><td>Default Value</td><td>-</td></tr>
<tr><td>Description</td><td>Used to control the group/channel. This input is effective only if nviXXOverride.state is invalid (-1).
<table>
<tr><th>Function</th><th>Description</th></tr>
<tr><td>SET_UP,
SET_DOWN</td><td>The light is dimmed relatively by the amount defined in the .setting field.</td></tr>
<tr><td>SET_STOP</td><td>Stop an ongoing fade.</td></tr>
<tr><td>SET_OFF</td><td>The light is switched off and the last light level is saved temporarily.</td></tr>
<tr><td>SET_ON</td><td>The light is switched on to the light level saved at the time of the last SET_OFF command.</td></tr>
<tr><td>SET_STATE</td><td>Set the light to the level defined by the .setting field.</td></tr>
</table>
The light is dimmed in ramping mode (constant rate, configuration property nciFadeRate is used).</td></tr>
</table>

nviXXOverride	
Type	SNVT_switch
Valid Range	.value: 0..100% .state: 0, 1, -1
Default Value	-
Description	If *nviXXOverride* is valid (.state=0/1), the current lamp/group/channel output is saved temporarily and the lamp/group/channel output is set to the specified value. If *nviXXOverride* is set to invalid (.state=-1), the lamp/group/channel output is set to the previously saved value. The light is dimmed in fading mode (constant time, configuration property *nciFadeTime* is used).

nviGPScene	
Type	SNVT_scene
Valid Range	.function: SC_LEARN, SC_RECALL, SC_RESET .scene_number: 1..16
Default Value	-
Description	Using this input network variable up to 16 DALI scenes can be saved and recalled. **Function** / **Description** SC_LEARN: Stores the current light levels of all lamps of the group in the specified scene number. SC_RECALL: Recall a previously stored scenes. SC_RESET: Delete a scene. When a new scene is selected the lamp values are adjusted in fading mode (configuration property *nciFadeTime* is used).

nviEmergTest	
Type	SNVT_switch
Valid Range	.value: 0..100% .state: 0, 1
Default Value	-
Description	Used to start the test of emergency lamps in a group. To start the test the *state* part of the *SNVT_switch* must be set to SW_ON (1). The *value* part selects one of the following tests: • **Function test** (value = 1%): Starts the function test of all DALI emergency lights in this group, which support this function. For details on the DALI emergency lighting function test see IEC 62386-202. If the Emergency interface is enabled the *nvoFuncTestFail* will show whether the last function test performed has failed. • **Duration test** (value = 2 %): Starts the duration test of all DALI emergency lights in this group, which support this function. For details on the DALI emergency lighting duration test see IEC 62386-202. If the Emergency interface is enabled the *nvoDurTestFail* will show whether the last duration test performed has failed. • **Mains test** (value = 100%): Typically used in a central battery system, to verify lights are dimmed to the emergency level when mains fail. When the test is started via *nviEmergTest*, the power of the emergency lamps is cut by setting the switch network variable output *nvoPowerOff* to ON. After the delays *nciEmTestDelay*, the value of all lamps in the group is compared to the reference value specified in *nciEmTestValue*. If a lamp does not reach the reference value (dim value above or below reference value) or if a lamp is offline, an alarm is generated as shown in Figure 214. Test results will be stored in the appropriate emergency light test log (see Section 8.1.4). To abort any active test the *SNVT_switch* must be set to {SW_OFF, 0%}.

<table>
<tr><th colspan="2">nviCommand</th></tr>
<tr><td>Type</td><td>UNVT_dali_cmd</td></tr>
<tr><td>Valid Range</td><td>.cmd: START_BURN_IN=0, STOP_BURN_IN=1, DALI_CMD_SET_COLOUR=2
.data.type: BALLAST=0, GROUP=1, CHANNEL=2
.data.index: 0..63
.data.arg: Union containing arguments depending on selected command.</td></tr>
<tr><td>Default Value</td><td>-</td></tr>
<tr><td>Description</td><td>Used to start and stop the burn-in mode.
<table>
<tr><th>Function</th><th>Description</th></tr>
<tr><td>START_BURN_IN</td><td>Start the burn-in mode for a ballast, group, or channel. The burn-in time is defined by the configuration property nciBurnInTime.</td></tr>
<tr><td>STOP_BURN_IN</td><td>Abort the burn-in mode for a ballast, group, or channel.</td></tr>
<tr><td>DALI_CMD_SET_COLOUR</td><td>Change light color for ballast, group, or channel. The argument field is defined as follows:
<pre>
struct set_colour {
  dali_colour_type_t type;
  union value {
     struct xy_coordinate {
        uint16 x;
        uint16 y;
     };
     uint16 colour_temperature_tc;
     uint16 primary_n_dim_level[6];
     struct rgbwaf {
        uint8 control;
        uint8 red;
        uint8 green;
        uint8 blue;
        uint8 white;
        uint8 amber;
        uint8 freecolour;
     };
  };
};
</pre>
The field type defines the way the colour is specified:

DALI_COLOUR_TYPE_XY_COORDINATE (0): The light colour is defined by the x any y coordinates in the CIE1931 colour space chromaticity diagram. The x and y values are given in the structure value.xy_coordinate.
DALI_COLOUR_TYPE_TC (1): The light colour is defined by the colour temperature Tc. Its value is given in the field value.colour_temperature_tc.
DALI_COLOUR_TYPE_PRIMARY_N (2): The light colour is defined by the dim level of up to 6 channels. The corresponding values are given in the array value.primary_n_dim_level.
DALI_COLOUR_TYPE_RGBWAF (3): The light colour is defined by colour channels red, green, blue, white, amber and freecolour. See IEC 62386-209 on details. The corresponding values ar given in the structure value.rgbwaf.

Note: Not all colour control types are supported by all DALI colour control ballasts. Check the documentation of the DALI ballast on which types are supported.</td></tr>
</table>
If the command requires an address the field data.type defines whether the command addresses a single ballast, a group or the all devices on the channel (broadcast). The field data.index gives the index of the associated lamp actuator (0-63) or group actuator (0-15) object, respectively. In case of type CHANNEL the index is ignored.</td></tr>
</table>

8.2.5.13 Output Network Variables

nvoXXValueFb	
Type	SNVT_switch
Valid Range	.value: 0..100% .state: 0, 1 On: .state = 1 and .value > 0 Off: .state = 0 and .value=any or .state=1 and .value = 0
Default Value	-
Description	Feedback value of the lamp/group/channel object. For a lamp actuator object this value is the current state and percentage of level intensity of the corresponding DALI ballast. The *.state* part is –1 (invalid) if the ballast is not present. For a group actuator or channel actuator object this value changes, whenever all ballasts of the group/channel adopt the same dim vale.

nvoXXRunHours	
Type	SNVT_elapsed_tm
Valid Range	.day: 0..65534 .hour: 0..23 .minute: 0..59 .second: 0..59
Default Value	-
Description	Reports the run (ON) hours. L-DALI calculates the run hours of each DALI ballast. The network variable *nvoGPRunHours* reports the maximum run hours of the lamps in a group. The network variable *nvoCHRunHours* reports the maximum run hours of the lamps in a channel. The network variable *nvoLARunHours* reports the run hours of each lamp. This network variable is only present if the Emergency interface is enabled (see Section 8.2). To reduce the number of network variables, the value is not available as a network variable on the Standard interface. If a lamp is replaced the run hours of a lamp can be initialized using the configuration property *nciRunHrInit*. If the ballast is capable of maintaining the run hours by itself (e.g. some DALI emergency lights), resetting the run hours via *nciRunHrInit* will also reset the run hours counter in the ballast.

nvoXXEnergyCnt	
Type	SNVT_elec_kwh
Valid Range	0..65534 Resolution: 1kWh
Default Value	-
Description	This network variable reports the energy consumption of a group or channel. L-DALI calculates the energy consumption of each DALI ballast. To reduce the number of network variables, the calculated value is not available as a network variable on the lamp actuator object but is used to calculate the total energy consumption for the group and channel actuator objects. The network variable *nvoGPEnergyCnt* reports the total energy consumption of all lamps in a group. The network variable *nvoCHEnergyCnt* reports the total energy consumption of all lamps in a channel. The nominal power of a lamp can be configured using the configuration property *nciNominalPwr*. The energy consumption of a lamp can be initialized with the configuration property *nciEnCntInit*.

nvoXXFailure	
Type	SNVT_switch
Valid Range	.value: 0..100% .state: 0, 1, -1
Default Value	-
Description	This network variable reports the status of the lamps in a group or channel. The network variable *nvoGPFailure* reports the percentage of failed ballasts in a group. The network variable *nvoCHFailure* reports the percentage of failed ballasts in a channel. A ballast failure is reported if either • Bus power for the related DALI channel fails, • The DALI ballast is not reachable via the DALI channel, or • The DALI ballast reports a failure (e.g. lamp failure) in its DALI status register.

nvoPowerOff	
Type	SNVT_switch
Valid Range	.value: 0, 100 .state: 0, 1
Default Value	-
Description	This network variable is used during the emergency light test to cut the power. The power is cut with an ON value (100.0, 1) and reapplied with an OFF value (0.0, 0).

nvoLampFailure	
Type	SNVT_state_64
Valid Range	.bit0: 0, 1 … .bit63: 0, 1
Default Value	-
Description	This network variable reports the status of the ballasts in the channel. Each bit corresponds to the lamp with the same index. In case of a failure the bit is set. A failure occurs if the corresponding ballast is offline or if it indicates a lamp failure or a ballast failure.

nvoBatteryCharge, nvoLABattCharge	
Type	SNVT_lev_percent
Valid Range	-163.840..163.830 Resolution: 0.005
Default Value	-
Description	Reports the battery charge of a self-contained emergency light or a group of self-contained emergency lights (if supported by the emergency light). The network variable *nvoBatteryCharge* reports the minimum battery charge of the emergency lights in a group. The network variable *nvoLABattCharge* reports the battery charge of each self-contained emergency light. This network variable is only present if the Emergency interface is enabled (see Section 8.2). To reduce the number of network variables, the value is not available as a network variable on the Standard interface.

nvoManOverFb	
Type	SNVT_switch
Valid Range	.value: 0, 100 .state: 0, 1, -1 On: .state=1 and .value=100 Off: .state=0 and .value=0 Auto: .state=-1 and .value=0
Default Value	-
Description	This network variable reports the current state of manual override for a channel.

nvoBallastFail	
Type	SNVT_state_64
Valid Range	.bit0: 0, 1 … .bit63: 0, 1
Default Value	-
Description	This network variable reports the status of the ballasts in the channel. Each bit corresponds to the ballast with the same index. In case the ballast indicates a ballast failure the bit is set.

nvoLampFail	
Type	SNVT_state_64
Valid Range	.bit0: 0, 1 … .bit63: 0, 1
Default Value	-
Description	This network variable reports the status of the lamps in the channel. Each bit corresponds to the ballast with the same index. In case the ballast indicates a lamp failure the bit is set.

nvoBatteryFail	
Type	SNVT_state_64
Valid Range	.bit0: 0, 1 … .bit63: 0, 1
Default Value	-
Description	This network variable reports the battery status of the self-contained emergency lights in the channel. Each bit corresponds to the ballast with the same index. In case the ballast indicates a battery failure the bit is set. Note, that this function is not supported by all types of self-contained emergency lights.

nvoFuncTestFail	
Type	SNVT_state_64
Valid Range	.bit0: 0, 1 … .bit63: 0, 1
Default Value	-
Description	This network variable reports the result of the last function test performed by each self-contained emergency light in the channel. Each bit corresponds to the ballast with the same index. In case the ballast indicates a failed function test the bit is set. Note, that this function is not supported by all types of self-contained emergency lights.

nvoDurTestFail	
Type	SNVT_state_64
Valid Range	.bit0: 0, 1 … .bit63: 0, 1
Default Value	-
Description	This network variable reports the result of the last duration test performed by each self-contained emergency light in the channel. Each bit corresponds to the ballast with the same index. In case the ballast indicates a failed duration test the bit is set. Note, that this function is not supported by all types of self-contained emergency lights.

8.2.5.14 Configuration Properties

All configuration properties listed in this section are implemented as Configuration Parameters using configuration files (CPT, CPV, and CPC files). File access is provided via LONMARK FTP.

nciLocation	
Type	SCPTLocation (SNVT_str_asc)
Valid Range	31 ASCII characters
Default Value	Lamp X/Group X/Channel X
Description	This configuration property can be used to specify the name of the lamp/group/channel.

nciRunHrInit	
Type	SCPTrunHrInit (SNVT_elapsed_tm)
Valid Range	.day: 0..65534 .hour: 0..23 .minute: 0..59 .second: 0..59
Default Value	0
Description	This configuration property is used to store or reset the initial value of the run hours counter *nvoRunHours* (only for lamp actuator objects). To reset the value set it to its invalid value (all 0xFF).

nciEnCntInit	
Type	UCPTenrgyCntInit (SNVT_elec_whr_f)
Valid Range	0..3.40282E38 Wh
Default Value	0
Description	This configuration property is used to store or reset the initial value of the energy counter *nvoEnergyCnt (*only for a lamp actuator objects). To reset the value set it to its invalid value (NaN).

nciGroups	
Type	UCPTgroups (SNVT_state)
Valid Range	.bit0: 0, 1 … .bit15: 0, 1
Default Value	0
Description	This configuration property defines the group membership of a lamp. Each bit corresponds to the group of the same index. If the bit is set, the lamp is member of the group.

nciOnDelay	
Type	UCPTonDelay (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	0
Description	This configuration property determines the delay after which the lamp value output is switched on.

nciOffDelay	
Type	UCPTOffDelay (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	0
Description	This configuration property determines the delay after which the lamp value output is switched off.

nciAutoOffDelay	
Type	UCPTautoOffDelay (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec 0s means disabled (no delay)
Default Value	0
Description	This configuration property determines the delay after which the lamp output is switched off automatically. The timer is started after receiving a lamp ON command via the *nviXXValue* or *nviXXSetting* network variable inputs. The configuration properties *nciAutoOffRestart* and *nciAutoOffBreak* modify the ehavior of the auto-off timer.

<table>
<tr><th colspan="2">nciAutoOffRestart</th></tr>
<tr><td>Type</td><td>UCPTautoOffRestart (SNVT_lev_disc)</td></tr>
<tr><td>Valid Range</td><td>ST_ON, ST_OFF</td></tr>
<tr><td>Default Value</td><td>ST_OFF</td></tr>
<tr><td>Description</td><td>This configuration property determines if the auto off timer can be restarted by an ON command via the nviXXValue or nviXXSetting network variable inputs.
<table>
<tr><th>Function</th><th>Description</th></tr>
<tr><td>ST_ON</td><td>While the auto-off timer is running, the timer can be restarted by an ON command via the nviXXValue or nviXXSetting network variable inputs.</td></tr>
<tr><td>ST_OFF</td><td>While the auto-off timer is running, all ON command via the nviXXValue or nviXXSetting network variable inputs are ignored.</td></tr>
</table></td></tr>
</table>

<table>
<tr><th colspan="2">nciAutoOffBreak</th></tr>
<tr><td>Type</td><td>UCPTautoOffBreak (SNVT_lev_disc)</td></tr>
<tr><td>Valid Range</td><td>ST_ON, ST_OFF</td></tr>
<tr><td>Default Value</td><td>ST_OFF</td></tr>
<tr><td>Description</td><td>This configuration property determines if the auto off timer can be aborted prematurely by an OFF command via the nviXXValue or nviXXSetting network variable inputs.
<table>
<tr><th>Function</th><th>Description</th></tr>
<tr><td>ST_ON</td><td>While the auto-off timer is running, the timer can be aborted prematurely by an OFF command via the nviXXValue or nviXXSetting network variable inputs.</td></tr>
<tr><td>ST_OFF</td><td>While the auto-off timer is running, all OFF command via the nviXXValue or nviXXSetting network variable inputs are ignored.</td></tr>
</table></td></tr>
</table>

nciOffWarningDly	
Type	UCPToffWarningDly (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec 0s means disabled (no delay)
Default Value	0
Description	This configuration property defines the time during which the user will be notified that the light will be switched off shortly. During this time, the light is dimmed to 50% of the current level. Lamps which cannot be dimmed (*nciMinLevel* = *nciMaxLevel*) blink during this time.

nciMinLevel	
Type	UCPTminLevel (SNVT_lev_cont)
Valid Range	0..100% Resolution: 0.5
Default Value	0.1%
Description	This configuration property specifies the minimum dim level of a lamp. Corresponds to the DALI-Register MIN LEVEL.

nciMaxLevel	
Type	UCPTmaxLevel (SNVT_lev_cont)
Valid Range	0..100% Resolution: 0.5
Default Value	100%
Description	This configuration property specifies the maximum dim level of a lamp. Corresponds to the DALI-Register MAX LEVEL.

nciPowerUpLev	
Type	UCPTpowerUpLevel (SNVT_lev_cont)
Valid Range	0..100% (for some ballasts 0% is not allowed) Resolution: 0.5
Default Value	100%
Description	Initial dim level of DALI lamp after power has been applied to lamp. Corresponds to the DALI-Register POWER ON LEVEL.

nciSysFailurLev	
Type	UCPTsysFailureLevel (SNVT_lev_cont)
Valid Range	0..100% Resolution: 0.5 255/invalid value means keep current dim level
Default Value	100%
Description	This configuration property determines the dim value of a DALI lamp in case of a DALI system failure. Corresponds to the DALI-Register SYSTEM FAILURE LEVEL.

nciFadeRate	
Type	UCPTfadeRate
Valid Range	2.8..360 steps/s
Default Value	45 steps/s
Description	This configuration property determines the fade rate of a lamp in percent per second in ramping mode. Corresponds to the DALI-Register FADE RATE.

nciFadeTime	
Type	UCPTfadeTime (SNVT_time_sec)
Valid Range	0..100 sec
Default Value	0 sec
Description	This configuration property determines the duration of a fade time in fading mode. Corresponds to the DALI-Register FADE TIME.

nciNominalPwr	
Type	UCPTnominalPwr (SNVT_power)
Valid Range	0..6553.5 W Resolution: 0.1 W 0 means AUTO
Default Value	AUTO
Description	This configuration property specifies the nominal power of the lamp. It is used to calculate the energy consumption. Some DALI ballasts can report their nominal power. In this case the configuration property should be set to 0 (AUTO).

nciEmTestDelay	
Type	UCPTtestDelay (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	0
Description	This configuration property is used for the emergency light test which is started by the network variable input *nviEmergTest*. It defines the time after which the dim level of the emergency lamps is compared with the reference value *nciEmTestValue*.

nciEmTestValue	
Type	UCPTtestValue (SNVT_lev_cont)
Valid Range	0..100% Resolution: 0.5
Default Value	0
Description	This configuration property is used for the emergency light test which is started by the network variable input *nviEmergTest*. It defines the reference value to which the dim level of the emergency lamps will be compared.

nciBurnInTime	
Type	UCPTburnInTime (SNVT_time_hour)
Valid Range	0..65,535 h
Default Value	100 h
Description	This configuration defines the burn-in time for new lamp. The burn-in mode is started via the network variable input *nviCommand*.

nciDaliCfg	
Type	UCPTdaliCfg
Valid Range	-
Default Value	-
Description	This configuration property contains the device type specific DALI-Registers. It is only available if the device type of a device is known, either because it was assigned or because the device type was set in the DALI Installation tab.

8.2.6 Light Sensor Object #1010

The L-DALI provides 16 light sensor objects for each DALI channel. A light sensor object makes the measured lux level of a DALI light sensor available on the CEA-709 network.

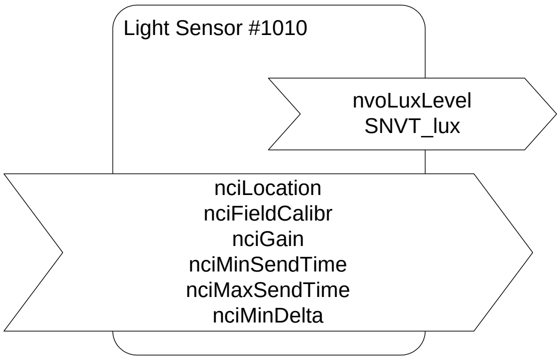


Figure 215: Light Sensor Object

This object type is enabled in the default interface. It can be disabled (together with the Occupancy Sensor objects).

8.2.6.1 Sensor Calibration

To provide a reliable measurement the light sensor needs to be calibrated. The calibration is best performed with the LINX Configurator software (refer to Section 7.6.2) or using the Web Interface (refer to Section 5.3.4.6).

8.2.6.2 Output Network Variables

nvoLuxLevel	
Type	SNVT_lux
Valid Range	0..65,335 lux
Default Value	0
Description	This network variable provides the lux level measured by the light sensor. If no corresponding light sensor is present on the DALI network the value is 0.

8.2.6.3 Configuration Properties

All configuration properties listed in this section are implemented as Configuration Parameters using configuration files (CPT, CPV, and CPC files). File access is provided via LONMARK FTP.

nciLocation	
Type	SCPTLocation (SNVT_str_asc)
Valid Range	31 ASCII characters
Default Value	Sensor X
Description	This configuration property can be used to specify the name of the light sensor.

nciFieldCalibr	
Type	UCPTfieldCalibArray
Valid Range	.index: 0..6 .value: 0..65,535 lux
Default Value	0/0
Description	This configuration property is used to calibrate the light sensor. To counter any non-linearity the sensor can be calibrated under up to seven different illumination levels (table with index 0-6). The *.index* field specifies the index into the gain table (*nciGain*) for different illumination levels. The *.value* field is used to specify the lux value measured with an external luxmeter. It is used together with the lux value reported by the sensor to compute the corresponding entry in the gain table. When reading the value from the device (CP upload) the value 0/0 will be returned. Writing this value will be ignored. This protects the calibration values from accidently being overwritten during (re-)commissioning.

nciGain	
Type	UCPTgainArray
Valid Range	.gain_mul[7]: 0..65,535 lux .gain_div[7]: 0..65,535 lux
Default Value	All 0/0.
Description	This configuration property is used to store the calibration data. .gain_mul[i] contains the lux value measured by a luxmeter. .gain_div[i] contains the lux value measured by the light sensor. Default values 0/0 are ignored.

nciMinSendTime	
Type	SCPTminSendTime (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	1.0 sec
Description	Minimum time between updates of the *nvoLuxLevel* output.

nciMaxSendTime	
Type	SCPTmaxSendTime (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	60.0 sec
Description	Maximum time between updates of the *nvoLuxLevel* output (heartbeat functionality).

nciMinDelta	
Type	SCPTminDeltaLevel (SNVT_lev_cont)
Valid Range	0..100% Resolution: 0.5
Default Value	2.5%
Description	This configuration property specifies the amount by which the lux level must change before *nvoLuxLevel* is updated.

8.2.7 Occupancy Sensor Object #1060

The L-DALI provides 16 occupancy sensor objects for each DALI channel. The occupancy sensor object makes the state of a DALI occupancy sensor available on the CEA-709 network.

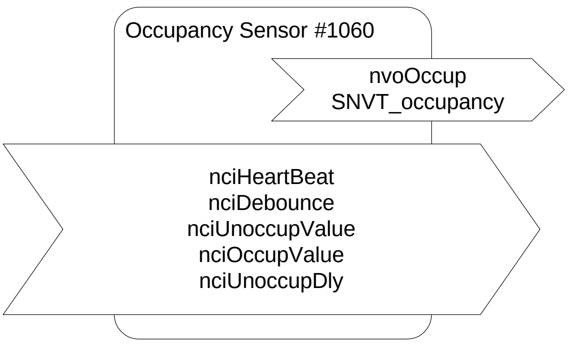


Figure 216: Occupancy Sensor Object.

This object type is enabled in the default interface. It can be disabled (together with the Light Sensor objects).

8.2.7.1 Timing Description

The timing diagram shown in Figure 217 defines the different timers used by the occupancy sensor object.

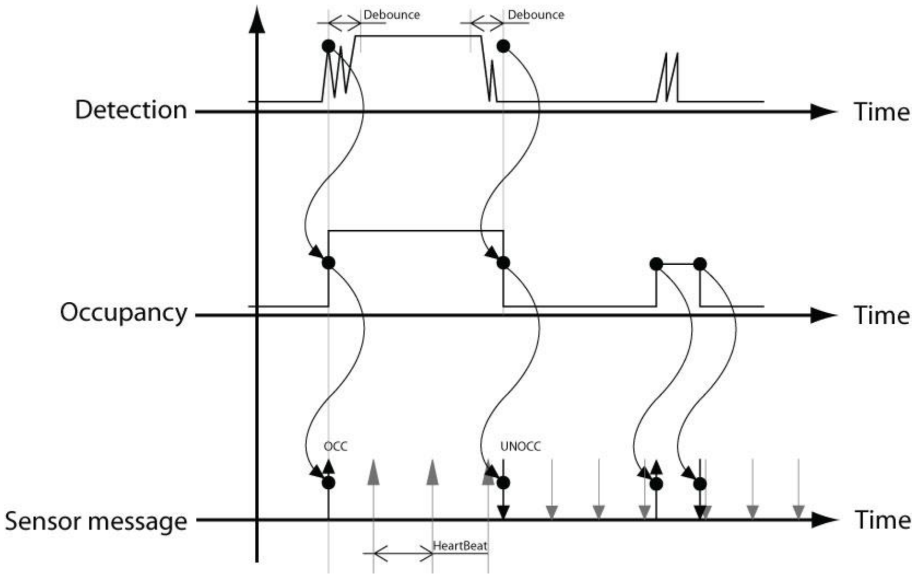


Figure 217: Timing of occupancy detection.

8.2.7.2 Output Network Variables

nvoOccup	
Type	SNVT_occupancy
Valid Range	OC_NUL (-1): Invalid Value OC_OCCUPIED (0): Area is occupied OC_UNOCCUPIED (1): Area is unoccupied OC_BYPASS (2): Area is temporarily occupied for the bypass period OC_STANDBY (3): Area is temporarily unoccupied
Default Value	OC_NUL
Description	This network variable provides the occupancy state measured by the occupancy sensor. If no corresponding occupancy sensor is present on the DALI network the value is OC_NUL.

8.2.7.3 Configuration Properties

All configuration properties listed in this section are implemented as Configuration Parameters using configuration files (CPT, CPV, and CPC files). File access is provided via LONMARK FTP.

nciHeartBeat	
Type	SCPTheartbeat (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	120.0 sec
Description	Maximum period of time that expires before the object automatically transmits the present value of the *nvoOccup* output NV.

nciDebounce	
Type	SCPTdebounce (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	0 sec
Description	Debouncing time to generate the detection envelope for occupancy detection and the OCCUPIED and NON OCCUPIED message.

nciUnoccupValue	
Type	UCPTunoccupiedValue (SNVT_occupancy)
Valid Range	OC_NUL (-1): Invalid Value OC_OCCUPIED (0): Area is occupied OC_UNOCCUPIED (1): Area is unoccupied OC_BYPASS (2): Area is temporarily occupied for the bypass period OC_STANDBY (3): Area is temporarily unoccupied
Default Value	OC_UNOCCUPIED
Description	This configuration property defines what value is transmitted via *nvoOccup* when the sensor detects that the room is unoccupied.

nciOccupValue	
Type	UCPToccupiedValue (SNVT_occupancy)
Valid Range	OC_NUL (-1): Invalid Value OC_OCCUPIED (0): Area is occupied OC_UNOCCUPIED (1): Area is unoccupied OC_BYPASS (2): Area is temporarily occupied for the bypass period OC_STANDBY (3): Area is temporarily unoccupied
Default Value	OC_ OCCUPIED
Description	This configuration property defines what value is transmitted via *nvoOccup* when the sensor detects that the room is occupied.

nciUnoccupDly	
Type	UCPTunoccupDly (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec 0s means disabled (no delay)
Default Value	0 sec
Description	This configuration property specifies the delay after which *nvoOccup* adopts the unoccupied value. The timer is started when the occupancy sensor detects that the room is unoccupied

8.2.8 Constant Light Controller Object #3050

The L-DALI provides 16 constant light controller objects with built in occupancy controller functionality for each DALI channel.

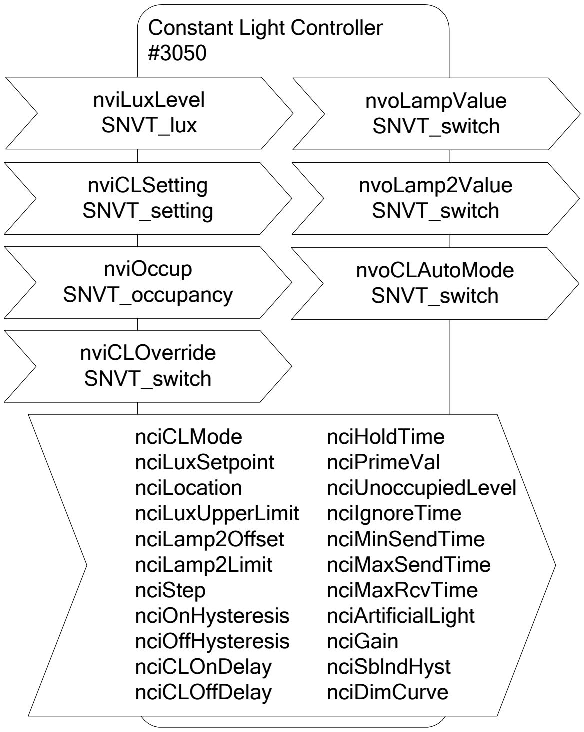


Figure 218: Constant Light Controller Object

This object type is enabled in the default interface. It can be disabled.

8.2.8.1 Interaction with other LONMARK Objects

To minimize the number of bindings that have to be created the constant light controller uses default connections to the local light sensor, occupancy sensor, and group objects in case no "real" bindings are done. This function is referred to as **Automatic Internal Bindings**. Figure 219 shows these default connections of the constant light controller object. Per default the constant light controller with index X controls the group object with index X. The current lux level is received from the light sensor object with index X and the occupancy state is received from the occupancy sensor X. The default connections can be changed by creating bindings with the network management tool. A detailed description under which conditions the default connections apply is given in Table 22.

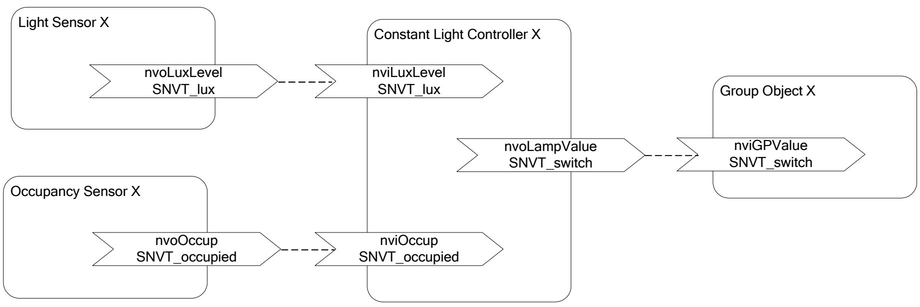


Figure 219: Default Connections to other LONMARK Objects

Per default automatic bindings are enabled. They can be disabled using the LINX Configurator software (see Section 7.16.2) or the Web-Interface (see Section 5.3.2).

Default Connection	Condition
Lux Level	The default connection applies if the following network variables are not bound: • *nviLuxLevel* (Constant Light Controller X) • *nviOccup* (Constant Light Controller X)
Occupancy State	The default connection applies if the following network variables are not bound: • *nviLuxLevel* (Constant Light Controller X) • *nviOccup* (Constant Light Controller X)
Lamp Value	The default connection applies if the following network variables are not bound: • *nvoLampValue* (Constant Light Controller X) • *nviGPValue* (Group Object X) • *nviGPSetting* (Group Object X) • *nviGPOverride* (Group Object X) • *nviGPScene* (Group Object X)

Table 22: Default Connections for Automatic Internal Bindings.

When using local DALI sensors and/or DALI groups internal manual CLC bindings can be configured (see Section 5.3.2 and 7.6). The Web-Interface for CLC Bindings can also be used to verify the current status of all bindings (internal and NVs).

8.2.8.2 Setpoint

The desired setpoint for the illumination level is configured via the configuration property *nciLuxSetpoint*. The input network variable *nviCLSetting* of type SNVT_setting switches the constant light controller on or off. Furthermore this network variable can be used to temporarily adjust the setpoint.

8.2.8.3 Operating Modes

The operating mode of the constant light controller is set via the configuration property *nciCLMode*. Table 23 shows the different operating modes and under which conditions they are used:

Operating Mode	Description
REGULATOR	A light sensor that measures the indoor illumination is installed. An occupancy sensor is installed.
REGULATOR_NO_OCC	A light sensor that measures the indoor illumination is installed. No occupancy sensor is installed.
REGULATOR_AUTO	Similar to REGULATOR, but a manual override is automatically relinquished when the room becomes unoccupied.
REGULATOR_BEDROOM	Similar to REGULATOR_AUTO, but a manual override is not automatically relinquished when the lights are overridden to off.
CONTROL	A light sensor that measures the outdoor illumination is installed. An occupancy sensor is installed.
CONTROL_NO_OCC	A light sensor that measures the outdoor illumination is installed. No occupancy sensor is installed.
CONTROL_AUTO	Similar to CONTROL, but a manual override is automatically relinquished when the room becomes unoccupied.
CONTROL_BEDROOM	Similar to CONTROL_AUTO, but a manual override is not automatically relinquished when the lights are overridden to off.
PRESENCE	Based on occupancy the lights shall be switched on and off.
PRESENCE_AUTO	Similar to PRESENCE, but a manual override is automatically relinquished when the room becomes unoccupied.
MANUAL_ON_AUTO_OFF	Lights are switched on manually, but shall be switched off when the room becomes unoccupied.
AMBIENT	Effect or ambient lighting is switched on based on the measured illumination level.
UPDATER	A DALI sensor with constant light controller functionality shall be mapped to the CEA-709 network. Only the setpoint shall be adjustable via the CEA-709 interface.

Table 23: Constant Light Controller Operation Modes

Regulator Mode

The REGULATOR mode has to be selected if the installed light sensor measures the indoor brightness. The constant light controller receives the current indoor lux level via the input network variable *nviLuxLevel* and adjusts the lamp level (from 0% to 100%) by means of a fuzzy control algorithm.

When the light is switched on the control algorithm selects a lamp value which results in a lux level close to the desired setpoint and then adjusts the lamp value in increments once every second until the configured setpoint is reached. The size of increments is adjusted dynamically but will never exceed the percentage value defined in the configuration property *nciStep*.

The regulator mode is available in different flavors:

- **With our without occupancy sensor**: If an occupancy sensor is installed and the room becomes unoccupied, the current light level is saved and the light is dimmed to unoccupied level (see Section 8.2.8.8). If the room becomes occupied before the light has reached the unoccupied level, the saved light level is restored.

- **With automatic relinquish**: Any manual override will be automatically relinquished when the room becomes unoccupied. In a special bedroom mode the automatic relinquish will only take place if the lights are on. If they were switched off manually, the controller does not return to automatic mode when the area turns unoccupied.

Control Mode

The CONTROL mode has to be selected if the installed light sensor measures the outdoor illumination. The constant light controller receives the current outdoor lux level via the input network variable *nviLuxLevel* and sets the lamp level (from 0% to 100%) accordingly (see Figure 220).

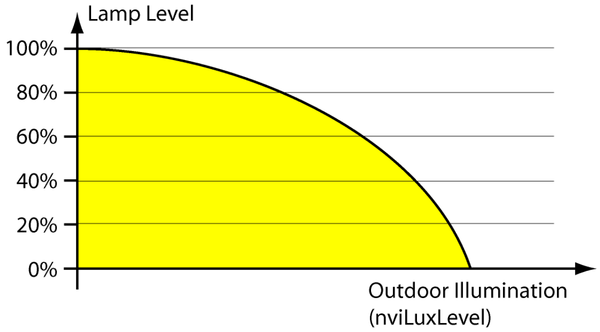


Figure 220: Control Mode

The configuration property *nciStep* ensures that the light level does not change abruptly. Per second the light level is not changed more than the percentage value defined by this configuration property. When the light is switched on or off, *nciStep* is not taken into account.

To ensure that the setpoint is reached under all weather conditions, the control curve shown Figure 220 has to be adjusted by calibrating the constant light controller. The calibration of the constant light controller is best performed with the LINX Configurator software (refer to Section 7.6.3).

The control mode is available in different flavors:

- **With our without occupancy sensor**: If an occupancy sensor is installed and the room becomes unoccupied, the current light level is saved and the light is dimmed to unoccupied level (see Section 8.2.8.8). If the room becomes occupied before the light has reached the unoccupied level, the saved light level is restored.

- **With automatic relinquish**: Any manual override will be automatically relinquished when the room becomes unoccupied. In a special bedroom mode the automatic relinquish will only take place if the lights are on. If they were switched off manually, the controller does not return to automatic mode when the area turns unoccupied.

Presence Mode

The PRESENCE mode has to be if the constant light controller shall operate as occupancy controller. In this mode the light is switched to the value *nciPrimeVal* if the room is occupied and to the unoccupied level (see Section 8.2.8.8) once the room becomes unoccupied.

If the network variable input *nviLuxLevel* is bound, the light is switched to the value *nciPrimeVal* only if the lux level is below *nciLuxSetpoint* and the room is occupied.

The presence mode is available in different flavors:

- **With or without automatic relinquish**: If automatic relinquish is active any manual override will be automatically relinquished when the room becomes unoccupied.

Manual-On/Auto-Off Mode

The MANUAL_ON_AUTO_OFF mode is a variation of the PRESENCE mode. However, in this mode the constant light controller does not switch on the lights, when the room becomes occupied. Rather, it switches the lights to the unoccupied level (see Section 8.2.8.8) when the room becomes unoccupied, using *nciHoldTime* similar to PRESENCE mode.

Lights can be switched on (or off) using *nviCLOverride* or *nviCLSetting* or via some other DALI master (e.g. a DALI push-button).

Ambient Mode

The AMBIENT mode is used if effect lighting (e.g. lights in a shop window, facade lighting, floor lights etc.) has to be switched on or off depending on a measured light level. Lights are switched to the value *nciPrimeVal* when the *nviLuxLevel* is below *nciLuxSetpoint* and to the unoccupied level (see Section 8.2.8.8) when *nviLuxLevel* is higher than *nciLuxSetpoint.*

Updater Mode

The UPDATER mode has to be selected when a DALI light/occupancy sensor with constant light controller functionality is installed on the DALI network. In this mode no light level computation is performed. The controller simply forwards the parameters to the DALI constant light controller and retrieves lamp setting from the DALI light sensor.

In general, LOYTEC does not recommend using UPDATER mode. Wherever possible use REGULATOR or CONTROL mode instead and use DALI sensors only as lux level and occupancy information source via the corresponding sensor objects. These modes allow improved influence of the constant light controller’s parameters via the fieldbus side.

8.2.8.4 Prioritized Control

The network variable input *nviCLOverride* can be used to override the constant light controller algorithm. Valid (.*state*=0/1) values received on this inputs are directly passed to the *nvoLampValue* output.

Similar an override is possible via the network variable input *nviCLSetting* when setting the .*function* field to SET_OFF or to SET_STATE. When using SET_STATE the value of the .*setting* field is directly passed to the *nvoLampValue* output.

When internal CLC bindings are used to control DALI groups an override can also be performed using a DALI push-button device controlling the same groups as the constant light controller (see Section 8.2.8.9).

To return to automatic mode the following options exist:

- When the network variable input *nviCLOverride* was used to override the constant light controller, set the .*state* field to -1 (invalid).
- Set the .*function* field of *nviCLSetting* to SET_ON.
- When using one of the modes with automatic override relinquish (xxx_AUTO or xxx_BEDROOM) the constant light controller will return to automatic mode when it changes its state to unoccupied.

Whether the constant light controller is in automatic mode or in manual/override mode is reflected by the network variable output *nvoCLAutoMode*.

8.2.8.5 Controlling Multiple Light Bands

The L-DALI constant light controller allows controlling two light bands (groups). The primary light band is near the inside of the building, the secondary light band is near the window front. Depending on the outdoor light intensity the primary light band has to be brighter than the secondary light band to illuminate the room evenly.

The primary light band is controlled by the output network variable *nvoLampValue*, the secondary light band by the output network variable *nvoLamp2Value*. Further for both light bands internal bindings to local DALI groups and their corresponding Group Actuator objects can be done (see Section 5.3.2 and 7.6).

The maximum difference between the two light bands can be configured via the configuration property *nciLamp2Offset* as shown in Figure 221. The configuration property *nciLamp2Limit* defines the light level above which the values of *nvoLampValue* and *nvoLamp2Value* are identical.

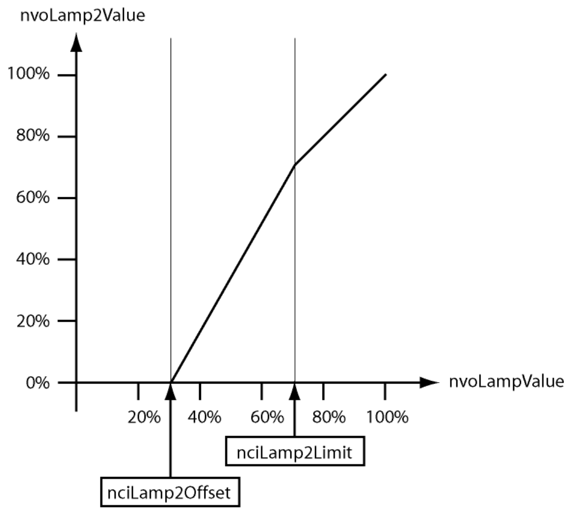


Figure 221: Secondary Light Band

8.2.8.6 Occupancy Detection

The constant light controller receives the occupancy information via the network variable input *nviOccup*.

If a hold time is configured (*nciHoldTime* > 0) occupancy evaluation is event based. That is, the constant light controller changes to the occupied state when the value OC_OCCUPIED is received. When the OC_UNOCCUPIED value is received the constant light controller remains in the occupied state until the hold time configured in *nciHoldTime* has expired and then changes to the unoccupied state. Further, if no OC_OCCUPIED is received for the time configured in *nciHoldTime* the constant light controller changes to the unoccupied state, too.

This behavior typically requires the occupancy sensor to send OC_OCCUPIED with a regular interval (heartbeat). It allows using multiple occupancy sensors with the same constant light controller (fan-in).

If no hold time is configured (*nciHoldTime* = 0) occupancy evaluation is state based. That is, the constant light controller changes to the occupied state when the value

OC_OCCUPIED is received and to unoccupied state when the value OC_UNOCCUPIED is received.

This behavior is typically required if the occupancy information is only propagated if the state changes (no heartbeat functionality). It does not allow using multiple occupancy (fan-in) sensors without additional logic (e.g. an Occupancy Controller).

After the light has been switched off any updates on the *nviOccup* input are ignored for the time specified in *nciIgnoreTime*.

8.2.8.7 Timing Parameters

To avoid that the constant light controller switches the light on and off repeatedly if the measured lux value is near the setpoint, a hysteresis can be configured. The configuration parameter *nciOffHysteresis* defines the hysteresis for switching off the lamp and the configuration property *nciOnHysteresis* defines the hysteresis for switching on the lamp. In addition a delay can be configured before the lamp is switched on (*nciClOnDelay*) or off (*nciClOffDelay*). Figure 222 illustrates these configuration parameters.

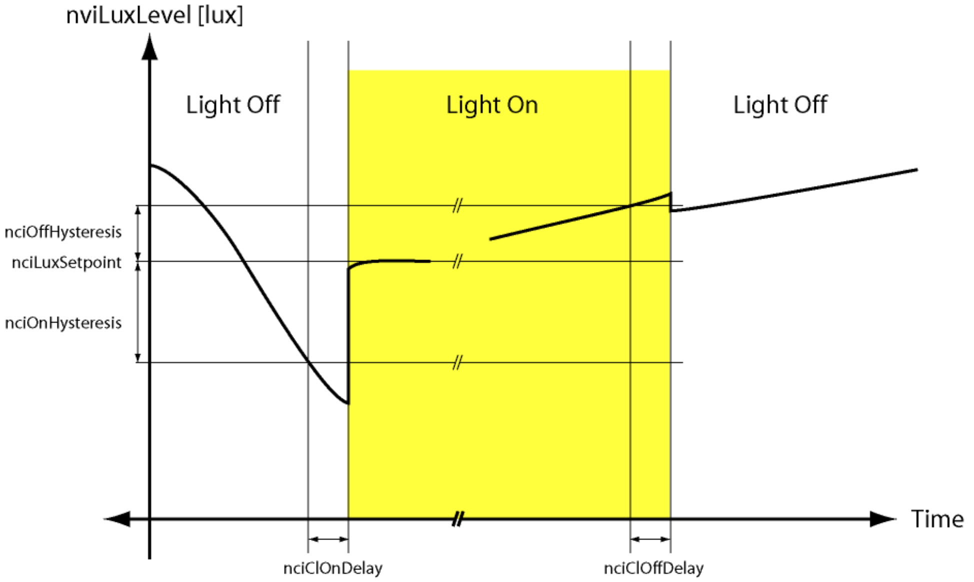


Figure 222: Constant Light Controller Timing

8.2.8.8 Unoccupied Level

If the constant light controller enters unoccupied state (see Section 8.2.8.6) *nciUnoccupiedLevel* and *nciCLOffDelay* determine the behavior as shown in Table 24.

nciUnoccupiedLevel	***nciCLOffDelay***	**Behavior**
0	0	Dim down and switch off lights
> 0	0	Dim down lights to value defined in *nciUnoccupiedLevel*.
0	> 0	Dim down to 0.5% and switch off lights after *nciCLOffDelay*.
> 0	> 0	Dim down lights to value defined in *nciUnoccupiedLevel* and switch off lights after *nciCLOffDelay*.

Table 24: Behavior of constant light controller when entering unoccupied state.

8.2.8.9 Interaction with DALI Buttons

When internal CLC bindings are used to control DALI groups an override can also be performed using a DALI push-button device like the LDALI-BM1 (or some other DALI master) controlling the same groups as the constant light controller.

Whenever the light level of one of the ballasts controlled by the constant light controller is changed by operating a DALI button the constant light controller will detect this and stop controlling the group and enter override mode. Furthermore, the *nviCLOverride* and the *nvoLampValue* and *nvoLamp2Value* will be updated with the new feedback value of the group controlled by the DALI button. See Section 8.2.8.4 on how to relinquish the override and return to automatic operation.

See Section 5.3.4.7 and 7.6 on how to set up DALI button functions.

8.2.8.10 Interacting with Sunblind Controller

One or more local sunblind controller objects can be linked to a constant light controller object. Typically sunblind controllers are linked to the constant light controller of the same room. If they change the sunblind's position or rotation the sunblind controllers inform the linked constant light controller. The constant light controller uses this information to "ignore" temporary changes of the rooms illumination levels due to the moving sunblinds and, thus, can avoid to switch on lights unnecessarily.

8.2.8.11 Museum Mode

The museum mode is activated by setting an upper limit for the lux level in the room using the configuration property *nciLuxUpperLimit*. In museum mode the constant light controller cooperates with the linked sunblind controllers to ensure that the measured lux level stays between *nciLuxSetpoint* and *nciLuxUpperLimit*. Therefore at least one sunblind controller has to be linked to the constant light controller.

If the light intensity in the room needs to be increased to reach the lower setpoint (*nciLuxSetpoint*) there are two possibilities: Increase the lamp level or open the sunblinds. The configuration property *nciSblndHyst* defines how much the lamp level is increased or decreased before opening or closing the sunblinds respectively.

8.2.8.12 Set-Up Checklist

To get a working constant light controller object at least the following configuration steps have to be performed:

1. Bindings: If the constant light controller is not operated with its default auto-bindings (see Section 8.2.8.1) at least the following NVs have to be bound:

    - nviOccup
    - nviLuxLevel
    - nvoLampValue

    Alternatively internal manual bindings can be used (see Section 5.3.2 and 7.6). The Web-Interface for CLC Bindings can be used to verify the current status of all bindings (internal and NVs).

2. Mode: The operating mode of the constant light controller has to be configured using *nciCLMode* (see Section 8.2.8.3).

3. Disable override: Ensure no override is active via the NV *nviCLOverride* (see Section 8.2.8.4).

8.2.8.13 Input Network Variables

nviLuxLevel	
Type	SNVT_lux
Valid Range	0..65,335 lux
Default Value	-
Description	This network variable input provides the illumination level measured by the light sensor. If the variable is not bound, the *nvoLuxLevel* output of the light sensor object with the same instance number as the constant light controller is used.

nviCLSetting	
Type	SNVT_setting
Valid Range	.function: SET_OFF (0): Override to off. SET_ON (1) Enable/Relinquish override. SET_DOWN (2) Decrease setpoint by specified value. SET_UP (3) Increase setpoint by specified value. SET_STATE (5) Override to specified value. .setting: 0..100, resolution 0.5 .rotation: not used
Default Value	
Description	This network variable input is used to enable (SET_ON) or disable (SET_OFF) the controller, to adjust the set point of the constant light controller (SET_UP and SET_DOWN) or to manually update the *nvoLampValue* and the *nvoLamp2Value* (SET_STATE).

Function	**Description**
SET_OFF	Deactivates the constant light controller and switches off the lamp value outputs *nvoLampValue* and *nvoLamp2Value*.
SET_ON	Activates the constant light controller. Relinquishes any override.
SET_UP, SET_DOWN	The set point of the controller can be increased (SET_UP) or decreased (SET_DOWN) temporarily. The *.setting* value is used to define the relative size of the increment/decrement. The changes made to the set point are not stored permanently into the memory. The next SET_ON command restored the original set point value (*nciLuxSetpoint*).
SET_STATE	Deactivates the constant light controller and sets *nvoLampValue* and *nvoLamp2Value* to the value specified in the *.setting* field (see Section 8.2.8.4). *nciLamp2Offset* and *nciLamp2Limit* are not considered.

nviOccup	
Type	SNVT_occupancy
Valid Range	OC_NUL (-1): Invalid Value OC_OCCUPIED (0): Area is occupied OC_UNOCCUPIED (1): Area is unoccupied OC_BYPASS (2): Ignored OC_STANDBY (3): Ignored
Default Value	OC_NUL
Description	This network variable input provides the occupancy state measured by the occupancy sensor. If the variable is not bound, the *nvoOccup* output of the occupancy sensor object with the same instance number as the constant light controller is used.

nviCLOverride	
Type	SNVT_switch
Valid Range	.value: 0..100% .state: 0, 1, -1 On: .state=1 and .value>0 Off: .state=0 and .value=0 Invalid: .state=-1
Default Value	
Description	This network variable input can be used to manually override the constant light controller (see Section 8.2.8.4). If a valid value is received (*.state* = 0/1), the constant light controller is disabled and *nvoLampValue* and *nvoLamp2Value* are set to the specified value. *nciLamp2Offset* and *nciLamp2Limit* are not considered. If an invalid value (*.state* –1) is received, the constant light controller returned to automatic mode. When performing an override via a DALI button the network variable will reflect the resulting feedback value of the group controlled by the DALI button (see Section 8.2.8.9).

8.2.8.14 Output Network Variables

nvoLampValue	
Type	SNVT_switch
Valid Range	.value: 0..100% .state: 0, 1, -1 On: .state=1 and .value>0 Off: .state=0 and .value=0 Invalid: .state=-1
Default Value	
Description	This network variable outputs provides the state (ON or OFF), and the percentage level of intensity for a lamp actuator. If the variable is not bound, the value is transmitted to the *nviGPValue* input of the DALI group object with the same instance number as the constant light controller.

nvoLamp2Value	
Type	SNVT_switch
Valid Range	.value: 0..100% .state: 0, 1, -1 On: .state=1 and .value>0 Off: .state=0 and .value=0 Invalid: .state=-1
Default Value	
Description	This network variable provides the state (ON or OFF), and the percentage level of intensity for the secondary light band. The value for the secondary light band (*nvoLamp2Value*) is set according to the configuration properties *nciLamp2Offset* and *nciLamp2Limit*.

nvoCLAutoMode	
Type	SNVT_switch
Valid Range	On: .state=1 and .value=100 Off: .state=0 and .value=0
Default Value	
Description	This network variable output provides the current state of the constant light controller. Off: The constant light controller is disabled or overridden by *nviCLOverride* or the DALI Mode button/LCD UI. On: The constant light controller is enabled and not overridden by *nviCLOverride* or the DALI Mode button/LCD UI.

8.2.8.15 Configuration Properties

All configuration properties listed in this section are implemented as Configuration Parameters using configuration files (CPT, CPV, and CPC files). File access is provided via LONMARK FTP.

nciLocation	
Type	SCPTLocation (SNVT_str_asc)
Valid Range	31 ASCII characters
Default Value	Light Controller X
Description	This configuration property can be used to specify the name of the constant light controller

nciLuxSetpoint	
Type	SCPTluxSetpoint (SNVT_lux)
Valid Range	0..65,335 lux
Default Value	500 lux
Description	Defines the illumination level setpoint.

nciCLMode	
Type	UCPTclMode (byte)
Valid Range	CL_MODE_DISABLED CL_MODE_REGULATOR CL_MODE_CONTROL CL_MODE_UPDATER CL_MODE_PRESENCE CL_MODE_REGULATOR_NO_OCC CL_MODE_CONTROL_NO_OCC CL_MANUAL_ON_AUTO_OFF CL_MODE_REGULATOR_AUTO CL_MODE_CONTROL_AUTO CL_MODE_PRESENCE_AUTO CL_MODE_AMBIENT CL_MODE_REGULATOR_BEDROOM CL_MODE_PRESENCE_BEDROOM
Default Value	CL_MODE_DISABLED
Description	This configuration property defines the operating mode of the constant light controller.

Operating Mode	**Description**
REGULATOR	A light sensor that measures the indoor illumination is installed. An occupancy sensor is installed.
REGULATOR_NO_OCC	A light sensor that measures the indoor illumination is installed. No occupancy sensor is installed.
REGULATOR_AUTO	Similar to REGULATOR, but a manual override is automatically relinquished when the room becomes unoccupied.
REGULATOR_BEDROOM	Similar to REGULATOR_AUTO, but a manual override is not automatically relinquished when the lights are overridden to off.
CONTROL	A light sensor that measures the outdoor illumination is installed. An occupancy sensor is installed.
CONTROL_NO_OCC	A light sensor that measures the outdoor illumination is installed. No occupancy sensor is installed.
CONTROL_AUTO	Similar to CONTROL, but a manual override is automatically relinquished when the room becomes unoccupied.
CONTROL_BEDROOM	Similar to CONTROL_AUTO, but a manual override is not automatically relinquished when the lights are overridden to off.
PRESENCE	Based on occupancy the lights shall be switched on and off.
PRESENCE_AUTO	Similar to PRESENCE, but a manual override is automatically relinquished when the room becomes unoccupied.
MANUAL_ON_AUTO_OFF	Lights are switched on manually, but shall be switched off when the room becomes unoccupied.
AMBIENT	Effect or ambient lighting is switched on based on the measured illumination level (switching on ambient lights does not influence measured illumination level).
UPDATER	A DALI sensor with constant light controller functionality shall be mapped to the CEA-709 network. Only the setpoint shall be adjustable via the CEA-709 interface.

nciLamp2Offset	
Type	UCPTlampOffset (SNVT_lev_cont)
Valid Range	0..100% Resolution: 0.5
Default Value	30%
Description	This configuration property defines the maximum offset between the output for the primary light band (*nvoLampValue*) and the secondary light band (*nvoLamp2Value*).

nciLamp2Limit	
Type	UCPTlampLimit (SNVT_lev_cont)
Valid Range	0..100% Resolution: 0.5
Default Value	70%
Description	This configuration property specifies the dim level at which the output for the secondary light band (*nvoLamp2Value*) becomes identical to the primary output (*nvoLampValue*).

nciStep	
Type	SCPTstep (SNVT_lev_cont)
Valid Range	0..100% Resolution: 0.5
Default Value	3%
Description	This configuration property defines the maximum step size that the constant light controller will take to approach the target illumination. Per second the light level is not changed more than the percentage value defined by this configuration property.

nciOnHysteresis	
Type	UCPTonHysteresis (SNVT_lev_cont)
Valid Range	0..100% Resolution: 0.5
Default Value	5%
Description	This configuration property defines the hysteresis for the input lux level as percentage of the setpoint for switching ON the lamp.

nciOffHysteresis	
Type	UCPToffHysteresis (SNVT_lev_cont)
Valid Range	0..100% Resolution: 0.5
Default Value	5%
Description	This configuration property defines the hysteresis for the input lux level as percentage of the setpoint for switching OFF the lamp.

nciMinSendTime	
Type	SCPTminSendTime (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	0.0 sec (disabled)
Description	Minimum time between updates of the *nvoLampValue* and *nvoLamp2Value* outputs.

nciMaxSendTime	
Type	SCPTmaxSendTime (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	300.0 sec
Description	Maximum time between updates of the *nvoLampValue* and *nvoLamp2Value* outputs (heartbeat functionality).

nciMaxRcvTime	
Type	SCPTmaxRcvTime (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	0.0 sec (disabled)
Description	This configuration property specifies the expected update interval of the *nviLuxLevel* network variable input. If the time elapses without an update of the variable an alarm is generated. If no update is received on *nviLuxLevel* for 10 intervals the *nvoLampValue* and *nvoLamp2Value* network variable outputs are set to 50%.

nciCLOffDelay	
Type	UCPTclOffDelay (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	300.0 sec
Description	This configuration property specifies the delay after which the lamp value output (nvoLampValue/ nvoLamp2Value) is switched off.

nciCLOnDelay	
Type	UCPTclOnDelay (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	0.0 sec
Description	This configuration property specifies the delay after which the lamp value output (nvoLampValue/ nvoLamp2Value) is switched on.

nciArtificialLight	
Type	UCPTartificialLight (SNVT_muldiv)
Valid Range	.multipier: 0..65,335 lux .divisor: 0..65,335 lux
Default Value	-
Description	This configuration property is used if the light sensor has a different sensibility for artificial light then for natural light. To configure the artificial light factor do the following: • Darken the room and set the lamp to the maximum value. Measure the lux value on the reference area with a luxmeter and with the light sensor. Then switch off the lamps and measure the lux value again with a luxmeter and the light sensor. • Enter the difference of the lux measured by the luxmeter in *.multiplier* and the lux value difference measured by the light sensor in *.divisor*. The value can be reset by setting the value to its invalid value (all 0xFF).

nciGain	
Type	SCPTgain (SNVT_muldiv)
Valid Range	.multipier: 0..65,335 lux .divisor: 0..65,335 lux
Default Value	1/1
Description	This configuration property is used to calibrate the light sensor. The configuration property is only needed if the light sensor does not have its own way of calibration. To calibrate the light sensor enter the lux value measured by a luxmeter in *.multiplier* and the lux value of the light sensor in *.divisor*. The lux level should be near the setpoint for best results. The value can be reset by setting the value to its invalid value (all 0xFF).

nciDimCurve	
Type	UCPTdimmingCurve
Valid Range	.used: 0, 1 .level_0: 0..100%, Resolution: 0.5 .level_1: 0..100%, Resolution: 0.5 .level_2: 0..100%, Resolution: 0.5 .level_3: 0..100%, Resolution: 0.5 .level_4: 0..100%, Resolution: 0.5 .level_5: 0..100%, Resolution: 0.5 .level_6: 0..100%, Resolution: 0.5 .level_7: 0..100%, Resolution: 0.5 .level_8: 0..100%, Resolution: 0.5 .level_9: 0..100%, Resolution: 0.5 .level_10: 0..100%, Resolution: 0.5
Default Value	.used = 0 .level_0 = 1% .level_1 = 1.5% .level_2 = 2.5% .level_3 = 4% .level_4 = 6.5% .level_5 = 10% .level_6 = 16% .level_7 = 25% .level_8 = 40% .level_9 = 64% .level_10 = 100%
Description	This configuration property permits to adjust the characteristic of the lamp for the human vision. If DALI lights are used the default values can be kept.

nciHoldTime	
Type	SCPTholdTime (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	600.0 sec
Description	This configuration property defines the hold time for the occupied state. When the OC_OCCUPIED value is received via *nviOccup* the constant light controller remains in the occupied state until the hold time has expired and then changes to the unoccupied state. If set to 0 the unoccupied state is entered when OC_UNOCCUPIED is received. **Note: Setting the *nciHoldTime* to 0 does not work when using the Echelon LonMaker Browser. As a workaround the LINX Configurator can be used to set the parameters. If used in Plug-In mode parameter values will be synced to the LNS database when downloaded to the L-DALI.**

nciPrimeVal	
Type	SNVT_switch
Valid Range	.value: 0..100% .state: 0, 1, -1 On: .state=1 and .value>0 Off: .state=0 and .value=0
Default Value	100%
Description	If the constant light controller operates in PRESENCE or AMBIENT mode, this configuration property defines the value adopted by *nvoLampValue* when the room is occupied.

nciUnoccupiedLevel	
Type	UCPTunoccupiedLevel (SNVT_lev_cont)
Valid Range	0..100% Resolution: 0.5
Default Value	0.0%
Description	This configuration property defines the value adopted by *nvoLampValue* when the constant light controller enters unoccupied state (see Section 8.2.8.8).

nciIgnoreTime	
Type	UCPTignoreTime (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	0.0 sec
Description	This configuration property defines the time during which updates on *nviOccup* are ignored after that lamp was switched off.

nciLuxUpperLimit	
Type	UCPTluxUpperLimit (SNVT_lux)
Valid Range	0..65,335 lux
Default Value	0 lux
Description	The museum mode is activated by setting this configuration property to a value greater than 0. In museum mode the constant light controller cooperates with the sunblind controller to ensure that the measured lux level stays between *nciLuxSetpoint* and *nciLuxUpperLimit*. Therefore a sunblind controller has to be linked to the constant light controller by writing the constant light controller index to the configuration property *nciClNumber* of the sunblind controller.

nciSblndHyst	
Type	UCPTsunblindHysteresis (SNVT_lev_cont)
Valid Range	0..100% Resolution: 0.5
Default Value	20%
Description	This configuration property applies only for museum mode. It defines by which percentage the lamp dim value is increased before letting more sunlight in by opening the sunblinds.

8.2.9 Sunblind Controller Object #6111

The L-DALI provides 16 sunblind controller objects for each DALI channel.

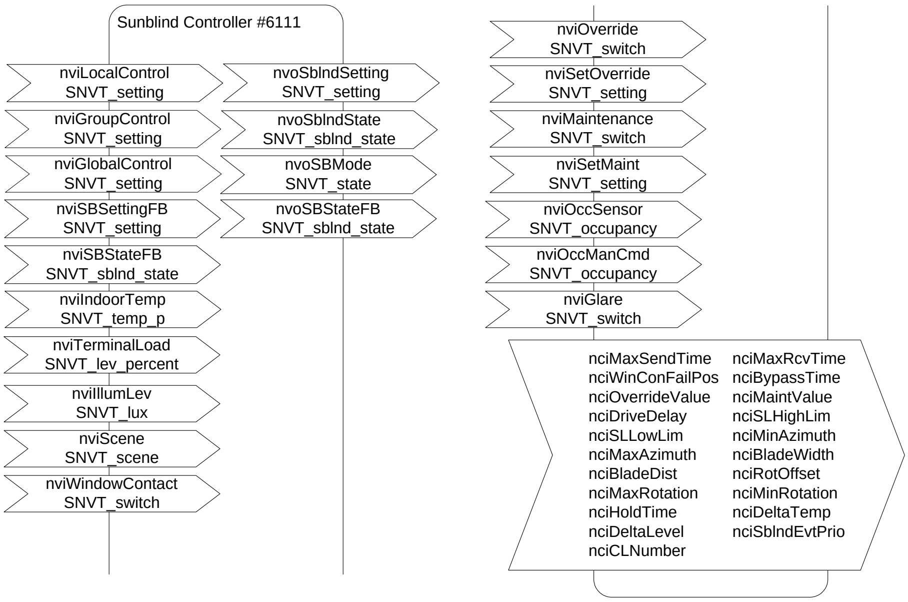


Figure 223: Sunblind Controller Object

This object type is enabled in the default interface. It can be disabled.

8.2.9.1 Glare Protection

The sunblind controller can automatically control the connected sunblind actuator to provide glare protection. To do so the sunblind controller needs to know the geometry of the sunblind blades.

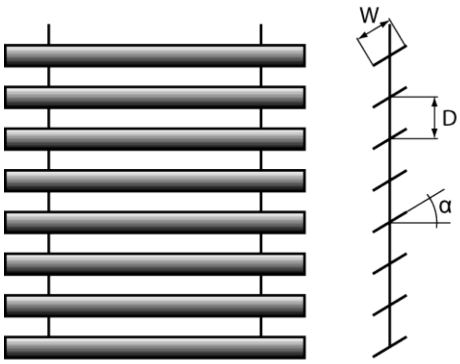


Figure 224: Sunblind Geometry

The blade distance (D) has to be configured in the configuration property *nciBladeDist* and the blade width in *nciBladeWidth*. The minimum and maximum rotation angle (α) is configured in *nciMinRot* and *nciMaxRot*. Based on this geometry data and the current sun elevation (*nviSunElevation*) the glare protection algorithm calculates the rotation angle for the blades. The rotation angle can be optimized for a specific sunblind type by specifying an offset in the configuration property *nciRotOffset*.

The following parameters determine if glare protection is necessary:

- Sun Azimuth (*nviSunAzimuth*): Depending on the orientation of the window, the sun can only shine into the room from a certain azimuth in the morning (*nciMinAzimuth*) to a certain azimuth in the evening (*nciMaxAzimuzh*).

- Outdoor lux level (*nviSunLux*): The glare protection will be activated if the measured outdoor lux level is above *nciSlHiLimGlare*. If the outdoor lux level falls below *nciSlLoLimGlare* the glare protection will be deactivated. These two configuration properties implement a hysteresis.

- Surrounding buildings/objects: The network variable inputs *nviGlobalGlare* and *nviGlobalShadow* can be used to provide geometry information on surrounding buildings/objects. Glare protection is active when the corresponding bit in *nviGlobalGlare* is active and the one in *nviGlobalShadow* is inactive.

Alternatively the need for glare protection can be computed by a separate device. In this case the network variable input *nviGlare* can be used to override the automatic glare detection algorithm.

8.2.9.2 View Protection

To protect rooms against prying eyes the sunblind can be closed automatically when the indoor light is switched on and the outdoor light level (*nviSunLux*) falls below the indoor light level (*nviIllumLev*). To avoid that the sunblind opens and closes repeatedly a hysteresis can be configured using the configuration property *nciViewProtHyst*.

8.2.9.3 Heating/Cooling

If the room is not occupied, the L-DALI can close the sunblind to provide thermal isolation in order to save energy.

The network variable input *nviTerminalLoad* is used to inform the controller of the current heating/cooling demand of the system. Positive values indicate that cooling energy is required, while negative values indicate that heating energy is required.

If *nviTerminalLoad* is not bound, but *nviOutdoorTemp* is bound, heating demand is assumed if the moving average outdoor temperature falls below 12°C/54°F, cooling demand is assumed if it is above 19°C/66°F.

The algorithm takes the following parameters into account:

- **Outdoor lux level (*nviSunLux*):** The energy of the sun can be used to warm a room. The configuration property *nciSlHiLimTem* defines the lux level above which the sun energy will make a noticeable contribution to warm the room if the sunblind is open. The configuration property *nciSlLoLimTem* define the lux level below which the sun energy is too low to warm the room. To avoid that the sunblind opens and closes repeatedly a hysteresis can be specified using the configuration property *nciTermSLHyst*.

- **Temperature:** The outdoor temperature (*nviOutdoorTemp*) is compared with the indoor temperature (*nviIndoorTemp*). Depending on whether the system is in heating or in cooling mode and on the temperature difference the sunblinds are closed to increase thermal isolation of the room. To avoid that the sunblind opens and closes repeatedly a hysteresis can be specified using the configuration property *nciTermTempHyst*.

8.2.9.4 Manual Control

To control the sunblind manually the L-DALI sunblind controller offers the possibilities to directly select a setpoint and to select a scene.

Direct Control

The sunblind can be controlled manually by the following network variable inputs of type SNVT_setting:

- ***nviLocalControl***: Using this input network variable a local control devices can provide the setpoint for the sunblind.
- ***nviGroupControl***: Using this input network variable a control device which controls a group of sunblinds can provide the setpoint for the sunblind.
- ***nviGlobalControl***: Using this input network variable a control device which controls all sunblinds can provide the setpoint for the sunblind.

Per default *nviLocalControl* takes precedence over *nviGroupControl* which in turn takes precedence over *nviGlobalControl*. The priorities can be changes as described in Section 8.2.9.9. Local, group, and global control commands can be cancelled by setting the *.function* field of the corresponding network variable to SET_NUL.

Scenes

Using the input network variable *nviSBScene* of type SNVT_scene up to 16 sunblind scenes can be saved and recalled. The command SC_LEARN stores the current sunblind position and rotation in the specified scene number. By a SC_RECALL command previously stored scenes can be recalled. Scenes can be deleted by a SC_RESET command.

8.2.9.5 Prioritized Control

The L-DALI sunblind controller offers two methods for prioritized control: maintenance mode and override mode. Maintenance mode has the highest priority and cannot be changed. The priority of override mode can be changed as described in Section 8.2.9.9.

Maintenance Mode

There are two alternative ways to control the sunblind in maintenance mode:

- ***nviSetMaint:*** This input network variable of type SNVT_setting overrides the sunblind output. The maintenance mode override is canceled by setting the *.function* field to SET_NULL.
- ***nviMaintenance, nciMaintValue:*** The *nviMaintenance* network variable of type SNVT_switch can be used to switch on/off maintenance mode. If the maintenance mode is active, the sunblind output is set to the value of the configuration property *nciMaintValue* (type *SNVT_setting*).

Override Mode

There are two alternative ways to control the sunblind in override mode:

- ***nviSetOverride:*** This input network variable of type SNVT_setting overrides the sunblind output. The override is canceled by setting the .function field to SET_NULL.

- ***nviSBOverride, nciOverrideValue****:* The *nviSBOverride* network variable of type SNVT_switch can be used to switch on/off override mode. If the override mode is active, the sunblind output is set to the value of the configuration property *nciOverrideValue* (type *SNVT_setting*).

8.2.9.6 Museum Mode

For the museum mode of the constant light controller (see Section 8.2.8.11) a sunblind controller must be linked to the constant light controller.

8.2.9.7 Output and Feedback

The L-DALI sunblind controller supports actuators with a control input of type *SNVT_sblnd_state* (see Figure 225) as well as sunblind actuators with a control input of type *SNVT_setting* (see Figure 226).

The configuration property *nciDriveDelay* defines the delay which the sunblind controller waits until it updates its outputs. If different drive delays are configured for the different sunblind controllers, the sunblinds will not drive at the same time avoiding a power peak.

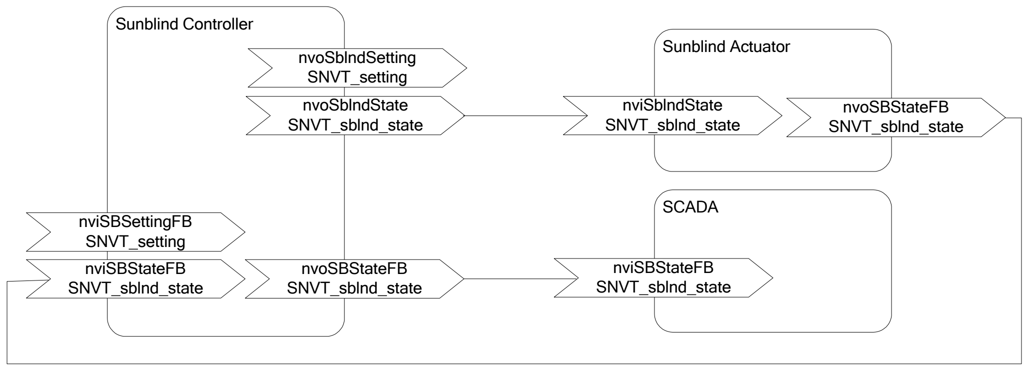


Figure 225: Controlling a Sunblind Actuator via SNVT_sblnd_state

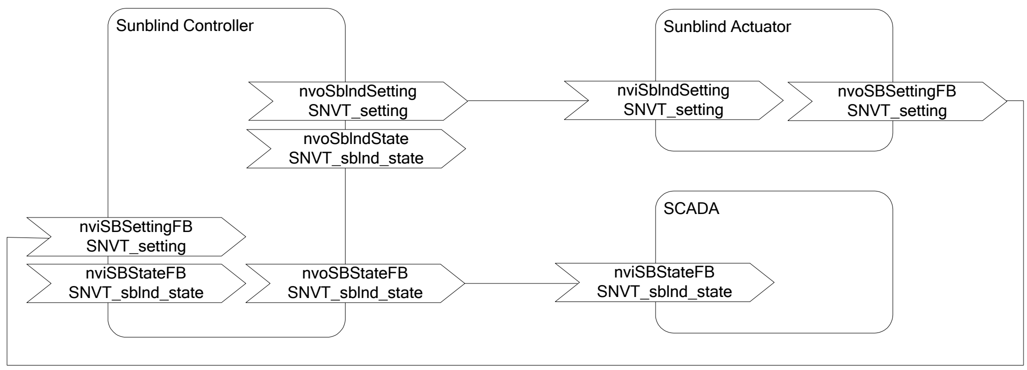


Figure 226: Controlling a Sunblind Actuator via SNVT_setting

The output network variable *nvoSBStateFB* can be used by e.g. a SCADA system to display the current state of the sunblind. It includes status information from the sunblind controller, which is merged with feedback information from the sunblind actuator, if available.

The event state (e.g. maintenance mode, rain condition, wind, etc) of the sunblind controller is reported in the output network variable *nvoSBMode*.

8.2.9.8 Occupancy Detection

The occupied/unoccupied status of the room can be determined by an occupancy sensor and/or a manual command.

Occupancy Sensor

To connect an occupancy sensor the network variable input *nviOccSensor* is used. Its behavior is identical to the *nviOccup* of the Constant Light Controller object (see Section 8.2.8.6).

Manual Command

The network variable input *nviOccManCmd* is used to override the occupancy sensor or if no occupancy sensor is available. This network variable of type *SNVT_occupancy* can have the following values:

- ***OC_OCCUPIED:*** The sunblind controller should operate in the occupied mode.
- ***OC_UNOCCUPIED:*** The sunblind controller should operate in the unoccupied mode.
- ***OC_BYPASS:*** The sunblind controller should operate in the occupied mode for a period of time defined by the configuration property *nciBypassTime*.

8.2.9.9 Sunblind Controller Events and Priority

The operation of the sunblind controller is influenced by a number of events. Table 25 describes the different events and shows their default priority. The priority of events can be changed. Events can be deactivated by setting the priority to -1. Usually this is done with the LINX Configurator as described in Section 7.6.4.

Priority	Event	Description
18 (highest)	Set maintenance request	The network variable input *nviSetMaint* is active (.*function* not SET_NUL). See Section 8.2.9.5 for detail.
17	Maintenance request	The network variable input *nviMaintenance* is switched on. The sunblind moves to the position specified in *nciMaintValue.* See Section 8.2.9.5 for detail.
16	Weather sensor alarm*	At least one of the weather sensors failed: • No update has been received on *nviWind* for longer than *nciWindRcvT.* • No update has been received on *nviRain* for longer than *nciRainRcvT.* • No update has been received on *nviFrost* for longer than *nciFrostRcvT.* The alarm condition for the wind, rain, and frost sensor is active only if the corresponding network variable input is bound. In case of a weather sensor alarm the sunblind drives to the position defined by *nciWinConFailPos.*
15	Window sensor alarm*	Window sensor failed: No update has been received on *nviWindowContact* for longer than *nciMaxRcvTime.* This alarm is only active if the *nviWindowContact* network variable is bound. In case of a window sensor alarm the sunblind drives to the position defined by *nciWinConFailPos*.

Priority	Event	Description
14	Frost condition*	The sunblind controller enters the frost state if the network variable input *nviFrost* is active. When *nviFrost* becomes inactive the sunblind controller remains in the frost state for the hold time configured in *nciRainFroOffDly*. The sunblind does not move as long as the sunblind controller is in the frost state.
13	Wind*	The sunblind controller enters the wind alarm state if the measured wind speed (*nviWind*) exceeds the threshold configured in *nciWindLimit*. When the wind speed falls below the threshold the sunblind controller remains in the wind alarm state for the hold time configured in *nciWindOffDly*. The sunblind remains open while the sunblind controller is in the wind alarm state.
12	Rain condition*	The sunblind controller enters the rain state if the network variable input *nviRain* is active. When *nviRain* becomes inactive the sunblind controller remains in the rain state for the hold time configured in *nciRainFroOffDly*. The sunblind remains open while the sunblind controller is in the rain state.
11	Set override request	The network variable input *nviSetOverride* is active (.function !=SET_NUL). See Section 8.2.9.5 for detail.
10	Override request	The network variable input *nviSBOverride* is switched on. The sunblind drives to the position specified in *nciOverrideValue*. See Section 8.2.9.5 for detail.
9	Window open*	The window is open: The network variable input *nviWindowContact* is active. The polarity of the window contact can be changed via the configuration property *nciInvWinCon*. The sunblind does not move as long as the window is open.
8	Local control	The network variable input *nviLocalControl* is active. See Section 8.2.9.4 for detail.
7	Group control	The network variable input *nviGroupControl* is active. See Section 8.2.9.4 for detail.
6	Global control	The network variable input *nviGlobalControl* is active. See Section 8.2.9.4 for detail.
5	Scene selection	Sunblind scene recalled via network variable input *nviScene*. See Section 8.2.9.4 for detail.
4	Anti-glare*	The sunblind is controlled by the anti-glare algorithm. Only active if room is occupied. See Section 8.2.9.1 for detail.
3	View protection*	The sunblind is controlled by the view protection algorithm. Only active if room is occupied. See Section 8.2.9.2 for detail.
2	Constant light request*	The sunblind is linked to the constant light controller and is controlled by the museum mode algorithm. Only active if room is occupied. See Section 8.2.8.11 for detail.
1	Up if occupied*	The room is occupied. See Section 8.2.9.8 for detail. If the room is occupied the sunblind moves up.
0 (lowest)	Heat/cool request*	The sunblind is controlled by the heating/cooling algorithm. Only active if room is not occupied. See Section 8.2.9.3 for detail.

Table 25: Sunblind Controller Events and Priority

For events marked with *) a minimum time can be configured via the configuration property *nciAutoEvMinTm*. These events will stay active for at least the configured time even if the event condition disappears before this time expires. In this way the sunblind controller ensures that the sunblind does not drive too frequently.

In case none of the events is active the sunblind remains in its current position.

8.2.9.10 Input Network Variables

nviLocalControl	
Type	SNVT_setting
Valid Range	.function: SET_NUL (-1): Invalid value. SET_DOWN (2) Decrease setting by specified value. SET_UP (3) Increase setting by specified value. SET_STOP (4) Stop action. SET_STATE (5) Setting on at specified value. .setting: 0..100, resolution 0.5 .rotation: -359.98 .. 360.00, resolution 0.02
Default Value	-
Description	This network variable input is used to manually control the sunblind. Usually a local control device gives this command. A local control command can be canceled by setting .function to SET_NUL.

nviGroupControl	
Type	SNVT_setting
Valid Range	.function: SET_NUL (-1): Invalid value SET_DOWN (2) Decrease setting by specified value SET_UP (3) Increase setting by specified value SET_STOP (4) Stop action SET_STATE (5) Setting on at specified value .setting: 0..100, resolution 0.5 .rotation: -359.98 .. 360.00, resolution 0.02
Default Value	-
Description	This network variable input is used to manually control the sunblind. Usually a device which is intended to control groups of controllers or actuators gives this command A group control command can be canceled by setting .function to SET_NUL.

nviGlobalControl	
Type	SNVT_setting
Valid Range	.function: SET_NUL (-1): Invalid value SET_DOWN (2) Decrease setting by specified value SET_UP (3) Increase setting by specified value SET_STOP (4) Stop action SET_STATE (5) Setting on at specified value .setting: 0..100, resolution 0.5 .rotation: -359.98 .. 360.00, resolution 0.02
Default Value	-
Description	This network variable input is used to manually control the sunblind. Usually a device which is intended to control all controllers or actuators gives this command A global control command can be canceled by setting .function to SET_NUL.

nviSBSettingFB	
Type	SNVT_setting
Valid Range	.function: SET_NUL (-1): Invalid value. SET_DOWN (2) Decrease setting by specified value. SET_UP (3) Increase setting by specified value. SET_STOP (4) Stop action. SET_STATE (5) Setting on at specified value. .setting: 0..100, resolution 0.5 .rotation: -359.98 .. 360.00, resolution 0.02
Default Value	-
Description	Feedback from sunblind actuator of type *SNVT_setting*.

nviSBStateFB	
Type	SNVT_sblnd_state
Valid Range	.pos.function: SET_NUL (-1): Invalid value. SET_DOWN (2) Decrease setting by specified value. SET_UP (3) Increase setting by specified value. SET_STOP (4) Stop action. SET_STATE (5) Setting on at specified value. .pos.setting: 0..100, resolution 0.5 .pos.rotation: -359.98 .. 360.00, resolution 0.02 .cmd_source: SBCS_NUL, SBCS_LOCAL, SBCS_GROUP, … .error_code: SBE_NUL, SBE_NO_ERROR, SBE_IN_PROGRESS, …
Default Value	-
Description	Feedback from sunblind actuator of type *SNVT_sblnd_state*.

nviIndoorTemp	
Type	SNVT_temp_p
Valid Range	-273.17°C...327.66°C Resolution: 0.01°C
Default Value	-
Description	This network variable input provides the room temperature measured by a temperature sensor. This input is used if the sunblinds are used for heating/cooling (see Section 8.2.9.3).

nviTerminalLoad	
Type	SNVT_lev_percent
Valid Range	-163.840..163.830 Resolution: 0.005
Default Value	
Description	This network variable input is used if the sunblinds are used for heating/cooling (see Section 8.2.9.3). The input informs the controller of the current heating/cooling demand. Positive values indicate that cooling energy is required, while negative values indicate that heating energy is required. If the network variable in not bound, the heating/cooling feature is disabled.

nviIllumLev	
Type	SNVT_lux
Valid Range	0..65,335 lux
Default Value	-
Description	This network variable input provides the illumination level in the room. The input is used for view protection (see Section 8.2.9.2). If *nviIllumLev* is higher than the *nviSunLux* then the sunblind is closed to protect the room against prying eyes.

nviScene	
Type	SNVT_scene
Valid Range	0..65,335 lux
Default Value	.function: SC_LEARN, SC_RECALL, SC_NUL .scene_number: 1..16
Description	Using this input network variable up to 16 sunblind scenes can be saved and recalled.

Function	Description
SC_LEARN	Stores the current sunblind position and rotation in the specified scene number.
SC_RECALL	Recall a previously stored scene.
SC_RESET	Delete a scene.

nviWindowContact	
Type	SNVT_switch
Valid Range	.value: 0..100% .state: 0, 1 Open: .state=1 and .value=100 Closed: .state=0 and .value=0
Default Value	-
Description	This network variable input informs the controller if the window is open or close. If the window is open, the sunblind stops moving.

nviOverride	
Type	SNVT_switch
Valid Range	.value: 0..100% .state: 0, 1 On: .state=1 and .value>=0.5% Off: .state=0 and .value=0
Default Value	-
Description	This network variable input is used to switch on/off override mode. If override mode is activated (.state=1 and .value between 0.5% and 100%), the sunblind adopts the setting specified by *nciOverrideValue*.

nviSetOverride	
Type	SNVT_setting
Valid Range	.function: SET_NUL (-1): Invalid Value SET_DOWN (2) Decrease setting by specified value SET_UP (3) Increase setting by specified value SET_STOP (4) Stop action SET_STATE (5) Setting on at specified value .setting: 0..100, resolution 0.5 .rotation: -359.98 .. 360.00, resolution 0.02
Default Value	-
Description	This input network variable overrides the sunblind output. The override is canceled by setting the .function field to SET_NULL.

nviMaintenance	
Type	SNVT_switch
Valid Range	.value: 0..100% .state: 0, 1 On: .state=1 and .value>=0.5% Off: .state=0 and .value=0
Default Value	-
Description	This network variable input is used to switch on/off maintenance mode. If maintenance mode is activated (.state=1 and .value between 0.5% and 100%), the sunblind adopts the setting specified by *nciMaitnValue*.

nviSetMaint	
Type	SNVT_setting
Valid Range	.function: SET_NUL (-1): Invalid Value SET_DOWN (2) Decrease setting by specified value SET_UP (3) Increase setting by specified value SET_STOP (4) Stop action SET_STATE (5) Setting on at specified value .setting: 0..100, resolution 0.5 .rotation: -359.98 .. 360.00, resolution 0.02
Default Value	-
Description	This input network variable overrides the sunblind output. The override is canceled by setting the .function field to SET_NULL.

<table>
<tr><th colspan="2">nviOccupSensor</th></tr>
<tr><td>Type</td><td>SNVT_occupancy</td></tr>
<tr><td>Valid Range</td><td>OC_NUL (-1): Invalid Value
OC_OCCUPIED (0): Area is occupied
OC_UNOCCUPIED (1): Area is unoccupied
OC_BYPASS (2): Ignored
OC_STANDBY (3): Ignored</td></tr>
<tr><td>Default Value</td><td>-</td></tr>
<tr><td>Description</td><td>This network variable input provides the occupancy state measured by an occupancy sensor. A hold time can be configured using the configuration property nciHoldTime.</td></tr>
</table>

<table>
<tr><th colspan="2">nviOccManCmd</th></tr>
<tr><td>Type</td><td>SNVT_occupancy</td></tr>
<tr><td>Valid Range</td><td>OC_NUL (-1): Invalid Value
OC_OCCUPIED (0): Area is occupied
OC_UNOCCUPIED (1): Area is unoccupied
OC_BYPASS (2): Area is temporarily occupied for the bypass period
OC_STANDBY (3): Ignored</td></tr>
<tr><td>Default Value</td><td>-</td></tr>
<tr><td>Description</td><td>This network variable input can be used to manually control the occupancy state.
<table>
<tr><th>Value</th><th>Description</th></tr>
<tr><td>OC_NUL</td><td>This is the initial value after power-up and it remains until another value is received. It is used to indicate that this network variable input is invalid, unused or to cancel a previous command.</td></tr>
<tr><td>OC_OCCUPIED</td><td>The sunblind controller should operate in the occupied mode.</td></tr>
<tr><td>OC_UNOCCUPIED</td><td>The sunblind controller should operate in the unoccupied mode</td></tr>
<tr><td>OC_BYPASS</td><td>The sunblind controller should operate in the occupied mode for a period of time defined by nciBypassTime.</td></tr>
</table></td></tr>
</table>

<table>
<tr><th colspan="2">nviGlare</th></tr>
<tr><td>Type</td><td>SNVT_switch</td></tr>
<tr><td>Valid Range</td><td>.value: 0..100%
.state: 0, 1,-1
On: .state=1 and .value>=0.5%
Off: .state=0 and .value=0
Auto .state=-1</td></tr>
<tr><td>Default Value</td><td>-</td></tr>
<tr><td>Description</td><td>This network variable is used to override the automatic glare detection algorithm (see Section 8.2.9.1).
<table>
<tr><th>Value</th><th>Description</th></tr>
<tr><td>On</td><td>Glare condition.</td></tr>
<tr><td>Off</td><td>No glare condition.</td></tr>
<tr><td>Auto</td><td>Use automatic glare detection algorithm.</td></tr>
</table></td></tr>
</table>

8.2.9.11 Output Network Variables

nvoSblndSetting	
Type	SNVT_setting
Valid Range	.function: SET_NUL (-1): Invalid Value SET_DOWN (2) Decrease setting by specified value SET_UP (3) Increase setting by specified value SET_STOP (4) Stop action SET_STATE (5) Setting on at specified value .setting: 0..100, resolution 0.5 .rotation: -359.98 .. 360.00, resolution 0.02
Default Value	-
Description	This network variable output provides the setpoint for the sunblind actuator.

nvoSblndState	
Type	SNVT_sblnd_state
Valid Range	.pos.function: SET_NUL (-1): Invalid Value SET_DOWN (2) Decrease setting by specified value SET_UP (3) Increase setting by specified value SET_STOP (4) Stop action SET_STATE (5) Setting on at specified value .pos.setting: 0..100, resolution 0.5 .pos.rotation: -359.98 .. 360.00, resolution 0.02 .cmd_source: SBCS_NUL, SBCS_LOCAL, SBCS_GROUP, … .error_code: SBE_NUL, SBE_NO_ERROR, SBE_IN_PROGRESS, …
Default Value	-
Description	This network variable output provides the setpoint for the sunblind actuator and reports errors and the cause of the latest change (as determined by the sunblind controller).

nvoSBMode	
Type	SNVT_state_64
Valid Range	.bit0: Heat/cool request .bit1: Up if occupied .bit2: Constant light request .bit3: View protection .bit4: Anti-glare .bit5: Scene selection .bit6: Global control .bit7: Group control .bit8: Local control .bit9: Window open .bit10: Override request .bit11: Set override request .bit12: Rain condition .bit13: Wind .bit14: Frost condition .bit15: Window sensor alarm .bit16: Weather sensor alarm .bit17: Maintenance request .bit18: Set maintenance request .bit19: Not used … .bit64: Not used
Default Value	-
Description	This network variable output reports the active events of the sunblind controller (see Section 8.2.9.9).

nvoSBStateFB	
Type	SNVT_sblnd_state
Valid Range	.pos.function: SET_NUL (-1): Invalid Value SET_OFF (0): Setting off SET_ON (1) Setting on SET_DOWN (2) Decrease setting by specified value SET_UP (3) Increase setting by specified value SET_STOP (4) Stop action SET_STATE (5) Setting on at specified value .pos.setting: 0..100, resolution 0.5 .pos.rotation: -359.98 .. 360.00, resolution 0.02 .cmd_source: SBCS_NUL, SBCS_LOCAL, SBCS_GROUP, … .error_code: SBE_NUL, SBE_NO_ERROR, SBE_IN_PROGRESS, …
Default Value	-
Description	This network variable output provides a feedback to a SCADA system: • If the sunblind actuator provides a feedback variable of type *SNVT_sblnd_state* and it is bound to *nviSBStateFB* the sunblind actuator feedback value is copied to *nvoSBStateFB*. • If this is not the case, but the sunblind actuator provides a feedback variable of type *SNVT_setting* and it is bound to *nviSBSettingFB* the value is copied to *nvoSBStateFB* with the *cmd_source* field copied from the *nvoSblndState*. • If none of the above is the case, the value of *nvoSblndState* is copied to *nvoSBStateFB*.

8.2.9.12 Configuration Properties

All configuration properties listed in this section are implemented as Configuration Parameters using configuration files (CPT, CPV, and CPC files). File access is provided via LONMARK FTP.

nciLocation	
Type	SCPTLocation (SNVT_str_asc)
Valid Range	31 ASCII characters
Default Value	Sunblind Ctrl X
Description	This configuration property can be used to specify the name of the sunblind controller.

nciMaxSendTime	
Type	SCPTmaxSendTime (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	0 sec
Description	Maximum time between updates for the *nvoSblndSetting* and *nvoSblndState* outputs (heartbeat functionality).

nciMaxRcvTime	
Type	SCPTmaxRcvTime (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	0.0 sec (disabled)
Description	This configuration property specifies the expected update interval of the *nviWindowContact* network variable input. If the time elapses without an update of the variable an alarm is generated and the sunblind adopts the safe position defined by the *nciWinConFailPos*.

nciWinConFailPos	
Type	SCPTdefaultSetting (SNVT_setting)
Valid Range	.function: SET_NUL (-1): Invalid Value SET_DOWN (2) Decrease setting by specified value SET_UP (3) Increase setting by specified value SET_STOP (4) Stop action SET_STATE (5) Setting on at specified value .setting: 0..100, resolution 0.5 .rotation: -359.98 .. 360.00, resolution 0.02
Default Value	SET_STATE, 0%, 0 deg
Description	This configuration property defines the safety position of a sunblind when a failure of the window contact (*nviWindowContact*) has been detected.

nciBypassTime	
Type	SCPTbypassTime (SNVT_time_min)
Valid Range	0..65,535 minutes
Default Value	0
Description	This configuration property defines the amount of time that the sunblind controller is in bypass (occupancy) mode following a single bypass request via *nviOccManCmd*. Additional bypass requests can restart the timer. Setting this configuration property to zero disables the bypass function and no bypass takes place.

nciOverrideValue	
Type	SCPTdefaultSetting (SNVT_setting)
Valid Range	.function: SET_NUL (-1): Invalid Value SET_DOWN (2) Decrease setting by specified value SET_UP (3) Increase setting by specified value SET_STOP (4) Stop action SET_STATE (5) Setting on at specified value .setting: 0..100, resolution 0.5 .rotation: -359.98 .. 360.00, resolution 0.02
Default Value	SET_NUL
Description	This configuration property defines the override value which is adopts by the sunblind when the *nviOverride* is at ON.

nciMaintValue	
Type	SCPTdefaultSetting (SNVT_setting)
Valid Range	.function: SET_NUL (-1): Invalid Value SET_DOWN (2) Decrease setting by specified value SET_UP (3) Increase setting by specified value SET_STOP (4) Stop action SET_STATE (5) Setting on at specified value .setting: 0..100, resolution 0.5 .rotation: -359.98 .. 360.00, resolution 0.02
Default Value	SET_NUL
Description	This configuration property defines the maintenance value which is adopts by the sunblind when the *nviMaintenance* is at ON.

nciDriveDelay	
Type	UCPTdriveDelay (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	0.0 sec (no delay)
Description	This configuration property defines the delay which the sunblind controller waits before it updates its outputs. If different drive delays are configured for the different sunblind controllers, the sunblinds will not drive at the same time avoiding a power peak. No drive delay is applied for local control (*nviLocalControl*).

nciSlHiLimGlare	
Type	UCPTsunLuxHighLimit (SNVT_lux)
Valid Range	0..65,335 lux
Default Value	30000
Description	This configuration property defines the high limit for the sun lux level input used by the automatic glare detection algorithm (see Section 8.2.9.1)

nciSlLoLimGlare	
Type	UCPTsunLuxHighLimit (SNVT_lux)
Valid Range	0..65,335 lux
Default Value	10000
Description	This configuration property defines the low limit for the sun lux level input used by the automatic glare detection algorithm (see Section 8.2.9.1)

nciMinAzimuth	
Type	UCPTminAzimuth (SNVT_angle_deg)
Valid Range	-359.98..360.00 deg Resolution: 0.02 deg
Default Value	0
Description	This configuration property defines the minimum azimuth for the sun elevation input used by the automatic glare detection algorithm (see Section 8.2.9.1).

nciMaxAzimuth	
Type	UCPTmaxAzimuth (SNVT_angle_deg)
Valid Range	-359.98..360.00 deg Resolution: 0.02 deg
Default Value	0
Description	This configuration property defines the maximum azimuth for the sun elevation input used by the automatic anti-glare algorithm (see Section 8.2.9.1).

nciBladeDist	
Type	UCPTbladeDistance (SNVT_length_mil)
Valid Range	0 ... 6533.5 mm Resolution: 0.1 mm
Default Value	7 cm
Description	This configuration property defines the distance between sunblind blades (see Section 8.2.9.1).

nciBladeWidth	
Type	UCPTbladeWidth (SNVT_length_mil)
Valid Range	0 ... 6533.5 mm Resolution: 0.1 mm
Default Value	9 cm
Description	This configuration property defines the width of the sunblind blades (see Section 8.2.9.1).

nciRotOffset	
Type	UCPTrotationOffset (SNVT_angle_deg)
Valid Range	-359.98..360.00 deg Resolution: 0.02 deg
Default Value	0
Description	This configuration property defines the rotation offset for the glare protection algorithm (see Section 8.2.9.1).

nciMaxRot	
Type	UCPTmaxRotation (SNVT_angle_deg)
Valid Range	-359.98..360.00 deg Resolution: 0.02 deg
Default Value	90
Description	This configuration property defines the maximum rotation for the glare protection algorithm (see Section 8.2.9.1).

nciMinRot	
Type	UCPTminRotation (SNVT_angle_deg)
Valid Range	-359.98..360.00 deg Resolution: 0.02 deg
Default Value	0
Description	This configuration property defines the minimum rotation for the glare protection algorithm (see Section 8.2.9.1).

nciDeltaTemp	
Type	SCPTminDeltaTemp (SNVT_temp_p)
Valid Range	-273.17°C...327.66°C Resolution: 0.01°C
Default Value	1°C
Description	This configuration property defines the amount by which the indoor temperature (*nviIndoorTemp*) value must change before it can take in account.

nciHoldTime	
Type	SCPTholdTime (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	600 sec
Description	This configuration property defines the hold time for the occupied state. When the OC_OCCUPIED value is received via *nviOccupSensor* the constant light controller remains in the occupied state until the hold time has expired and then changes to the unoccupied state. If set to 0 the unoccupied state is entered when OC_UNOCCUPIED is received.

nciSblndEvtPrio	
Type	UCPTsunblindEventPriority
Valid Range	.heat_cool_req: -128..127 .up_if_occupied: -128..127 .constant_light_req: -128..127 .view_protect: -128..127 .glare: -128..127 .scene_sel: -128..127 .global_req: -128..127 .group_req: -128..127 .local_req: -128..127 .window: -128..127 .override_req: -128..127 .set_override_req: -128..127 .rain: -128..127 .wind: -128..127 .frost: -128..127 .window_sensor_alarm: -128..127 .weather_sensor_alarm: -128..127 .maintenance_req: -128..127 .set_maintenance_req: -128..127
Default Value	All 0
Description	This configuration property defines the priorities of sunblind events (see Section 8.2.9.9). Higher values render higher priority. If multiple events have the same priority then they are prioritized with the default priority. Events can be disabled by setting a negative value for its priority. Events *maintenance_req* and *set_maintenance_req* will always be the highest priority events. Their value is used to define which of the two events has the higher priority.

nciAutoEvMinTm	
Type	UCPTautoDelay (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	600 sec
Description	Use to specify the minimum time for the following events: • Weather sensor alarm • Window sensor alarm • Frost condition • Wind • Rain condition • Window open • Anti-glare • View protection • Constant light request • Up if occupied • Heating/cooling request The event will stay active for at least the configured time even if the event condition disappears before this time expires. In this way the sun blind controller ensures that the sunblind does not drive too frequently.

nciCLNumber	
Type	UCPTobjectIndex (SNVT_count)
Valid Range	0..15 0xFFFF invalid
Default Value	0xFFFF (no Constant Light Controller linked)
Description	This configuration property specifies the index of the constant light controller that is linked to the sunblind controller (see Section 8.2.8.7 und 8.2.8.11). Object indices start with index 0.

8.2.10 Globals #0005

The L-DALI provides one Globals object for each DALI channel. It provides global network variable inputs and configuration properties which are not specific for a certain constant light controller or sunblind controller.

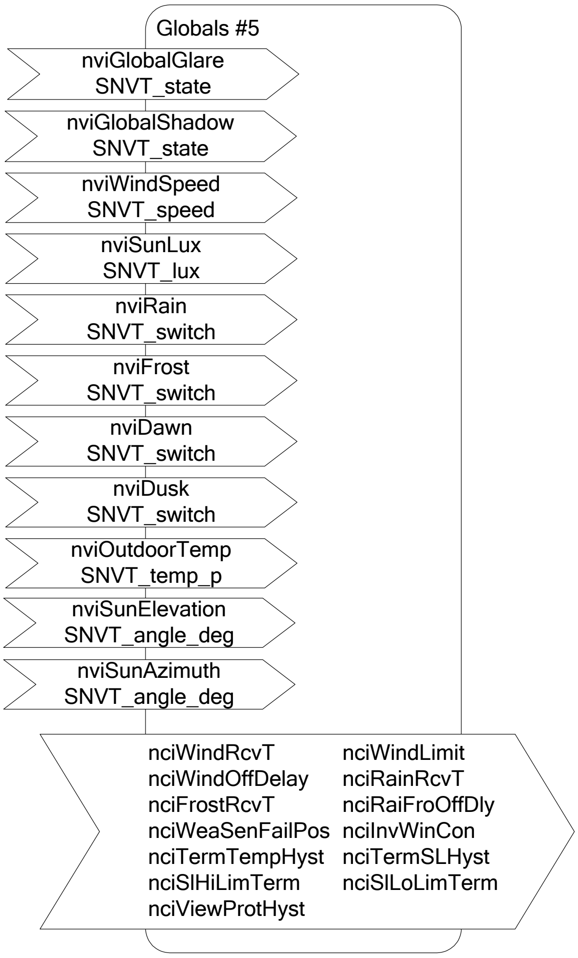


Figure 227: Global Object

This object type is enabled in the default interface. It will be disabled with the Sunblind Controller objects.

8.2.10.1 Network Variable Inputs

nviGlobalGlare	
Type	SNVT_state
Valid Range	.bit0 0, 1 … .bit15 0, 1
Default Value	
Description	This network input permits an external calculation of glare condition for all of the 16 sunblind controllers per channel (see Section 8.2.9.1). Automatic glare protection for the sunblind controller object x is deactivated if the bit with index x is not set.

nviGlobalShadow	
Type	SNVT_state
Valid Range	.bit0 0, 1 … .bit15 0, 1
Default Value	
Description	This network input permits an external calculation of shadow condition for all of the 16 sunblind controllers per channel (see Section 8.2.9.1). Automatic glare protection for the sunblind controller object x is deactivated if the bit with index x is set.

nviWindSpeed	
Type	SNVT_speed
Valid Range	0..6553.5 m/s Resolution: 0.1 m/s
Default Value	
Description	This network variable input provides the current wind speed measured by a wind sensor. If the wind speed is higher than the value configured in *nciWindLimit*, the sunblinds must go up.

nviSunLux	
Type	SNVT_lux
Valid Range	0..65,335 lux
Default Value	
Description	This network variable input provides the current outdoor illumination measured by a light sensor.

nviRain	
Type	SNVT_switch
Valid Range	.value: 0..100% .state: 0, 1 Rain: .state=1 and .value>=0.5% No Rain: .state=0 and .value=0
Default Value	
Description	This network variable input provides the rain status. If it is raining, the sunblinds must go up.

nviFrost	
Type	SNVT_switch
Valid Range	.value: 0..100% .state: 0, 1 Frost: .state=1 and .value>=0.5% No Frost: .state=0 and .value=0
Default Value	
Description	This network variable input provides the frost status. If a frost condition is detected, the sunblinds must not move.

nviDawn	
Type	SNVT_switch
Valid Range	.value: 0..100% .state: 0, 1 Dawn: .state=1 and .value>=0.5% No Dawn: .state=0 and .value=0
Default Value	
Description	Currently not used.

nviDusk	
Type	SNVT_switch
Valid Range	.value: 0..100% .state: 0, 1 Dusk: .state=1 and .value>=0.5% No Dusk: .state=0 and .value=0
Default Value	
Description	Currently not used.

nviOutdoorTemp	
Type	SNVT_temp_p
Valid Range	-273.17°C..327.66°C Resolution: 0.01°C
Default Value	
Description	This network variable input provides the current outdoor temperature. This input is used when the sunblinds are used to heat/cool a room (see Section 8.2.9.3).

nviSunElevation	
Type	SNVT_angle_deg
Valid Range	-359.98..360.00 deg Resolution: 0.02 deg
Default Value	
Description	This network variable input provides the current sun elevation. The elevation is the angle between the horizon and the middle of the sun, considered out of the viewpoint of an observer. The values 0 – 90° are typically used (0= Sunrise/Sunset; 90 = zenith position).

nviSunAzimuth	
Type	SNVT_angle_deg
Valid Range	-359.98..360.00 deg Resolution: 0.02 deg
Default Value	
Description	This network variable input provides the current sun azimuth. The values 0 – 359° are typically used (0 = NORTH, 90 = EAST, 180 = SOUTH, 270 = WEST).

8.2.10.2 Configuration Properties

All configuration properties listed in this section are implemented as Configuration Parameters using configuration files (CPT, CPV, and CPC files). File access is provided via LONMARK FTP.

nciWindRcvT	
Type	SCPTmaxRcvTime (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	0.0 sec (disabled)
Description	This configuration property specifies the expected update interval of the *nviWindspeed* network variable input. If the time elapses without an update of the variable an alarm is generated and the sunblinds adopt the safe position defined by *nciWeaSenFailPos*.

nciWindLimit	
Type	UCPTwindLimit (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	0.0 sec
Description	This configuration property defines the maximum wind for the sunblinds. If the current wind speed (*nviWindspeed*) is greater than the configured values the sunblinds must go up.

nciWindOffDelay	
Type	UCPTweaSenOffDelay (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	0.0 sec
Description	This configuration property defines the time for which the current wind speed (*nviWindspeed*) must be below the critical wind speed (*nciWindLimit*) before the sunblind controller return to normal operation.

nciRainRcvT	
Type	SCPTmaxRcvTime (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	0.0 sec (disabled)
Description	This configuration property specifies the expected update interval of the *nviRain* network variable input. If the time elapses without an update of the variable an alarm is generated and the sunblinds adopt the safe position defined by *nciWeaSenFailPos*.

nciFrostRcvT	
Type	SCPTmaxRcvTime (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	0.0 sec (disabled)
Description	This configuration property specifies the expected update interval of the *nviFrost* network variable input. If the time elapses without an update of the variable an alarm is generated and the sunblinds adopt the safe position defined by *nciWeaSenFailPos*.

nciRaiFroOffDly	
Type	UCPTweaSenOffDelay (SNVT_time_sec)
Valid Range	0..6553.4 sec Resolution: 0.1 sec
Default Value	0.0 sec
Description	This configuration property defines the time for which the rain/frost condition (*nviRain*/*nviFrost*) has to be inactive before the sunblind controller returns to normal operation.

nciWeaSenFailPos	
Type	SCPTdefaultSetting (SNVT_setting)
Valid Range	.function: SET_NUL (-1): Invalid Value SET_DOWN (2) Decrease setting by specified value SET_UP (3) Increase setting by specified value SET_STOP (4) Stop action SET_STATE (5) Setting on at specified value .setting: 0..100, resolution 0.5 .rotation: -359.98 .. 360.00, resolution 0.02
Default Value	SET_STATE, 0%, 0 deg
Description	This configuration property defines the safety position of sunblinds when a failure of a safety relevant weather sensor (e.g. *nviWindspeed*, *nviRain* or *nviFrost*) is detected.

nciInvWinCon	
Type	UCPTautoOffBreak (SNVT_lev_disc)
Valid Range	ST_ON, ST_OFF Inverted: ST_ON Not Inverted: ST_OFF
Default Value	ST_OFF (not inverted)
Description	This configuration property allows inverting the window contact inputs of the sunblind controllers (*nviWindowContact*).

nciTermTempHyst	
Type	SCPTtemperatureHysteresis (SNVT_temp_diff_p)
Valid Range	-327.68°C..327.66°C Resolution: 0.01°C
Default Value	1°C
Description	This configuration property defines the hysteresis for the outdoor temperature (*nviOutdoorTemp*). It is used for the heating/cooling algorithm (see Section 8.2.9.3).

nciTermSLHyst	
Type	UCPTterminalLoadSLhysteresis (SNVT_lev_cont)
Valid Range	0..100% Resolution: 0.5
Default Value	5.0%
Description	This configuration property defines the hysteresis for the sun lux level input (*nviSunLux*). It is used by the heating/cooling algorithm (see Section 8.2.9.3).

nciSlHighLimTerm	
Type	UCPTsunLuxHighLimit (SNVT_lux)
Valid Range	0..65,335 lux
Default Value	30000
Description	This configuration property defines the high limit for the sun lux level input (*nviSunLux*). It is used by the heating/cooling algorithm (see Section 8.2.9.3).

nciSlLowLimTerm	
Type	UCPTsunLuxLowLimit (SNVT_lux)
Valid Range	0..65,335 lux
Default Value	10000 lux
Description	This configuration property defines the low limit for the sun lux level input (*nviSunLux*). It is used by the heating/cooling algorithm (see Section 8.2.9.3).

nciViewProtHyst	
Type	UCPTviewProtectionSLhysteresis (SNVT_lev_cont)
Valid Range	0..100% Resolution: 0.5
Default Value	5.0%
Description	This configuration property defines the hysteresis for the sun lux level input (*nviSunLux*). It is used by the view protection algorithm (see Section 8.2.9.2).

8.2.11 Button #0001

The L-DALI provides 64 button objects for each DALI channel to represent DALI buttons, switches, remote controls, panels etc. supported by the L-DALI. See Section 8.4.4 for supported devices.

This object type is disabled in the default interface. If required these objects must be enabled as described in the project settings (see Section 7.16.3). The network variables available in the button objects depend on the template selected in the project settings.

In any case the object contains the following configuration properties:

nciLocation	
Type	SCPTLocation (SNVT_str_asc)
Valid Range	31 ASCII characters
Default Value	Button X
Description	This configuration property can be used to specify the name of the button.

The following object templates are available:

8.2.11.1 SNVT_switch

This template has to be selected when a DALI button device allowing button function configuration is used and the state of the button has to be available on the LONMARK interface. In this case each Button object contains the following network variables:

nvoButtonState	
Type	SNVT_state
Valid Range	-
Default Value	0
Description	If a button input is configured with the button function *Generic* the state of this input is mapped to the corresponding bit of the *SNVT_state* (bit0 to input 1, bit1 to input 2, etc.). The bit is set to 1 if the button is pressed or the contact is closed and 0 otherwise.

8.2.11.2 Philips OccuSwitch

This template has to be selected when expanding an IRT 8080 IR-remote control the Philips OccuSwitch DALI LRM2090/20 to controlling sunblinds, fan speed, and temperature setpoint in a room. In this case each Button object contains the following network variables:

nvoSetPtOffset	
Type	SNVT_temp_p
Valid Range	-3°C to +3°C
Default Value	-
Description	This network variable output allows adjusting the HVAC setpoint by supplying an offset to a preconfigured setpoint. Its value is controlled by the temperature setpoint button on the IRT 8080.

nvoFanSpeedCmd	
Type	SNVT_switch
Valid Range	.value: 0%, 33%, 66%, 100% .state: 0, 1 On: .state = 1 and .value > 0 Off: .state = 0 and .value=any or .state=1 and .value = 0
Default Value	-
Description	This network variable output allows control the speed of a ventilation fan. Its value is controlled by the temperature fan speed button on the IRT 8080.

nvoSblndControl	
Type	SNVT_setting
Valid Range	.function: SET_NUL (-1): Invalid value. SET_DOWN (2) Decrease setting by specified value. SET_UP (3) Increase setting by specified value. SET_STOP (4) Stop action. SET_STATE (5) Setting on at specified value. .setting: 0..100, resolution 0.5 .rotation: -359.98 .. 360.00, resolution 0.02
Default Value	-
Description	This network variable output allows control the position and the rotation of a sunblind (via a sunblind controller or sunblind actuator object). Its value is controlled by the sunblind button on the IRT 8080. A long press (> 1 second) of the button will set the network variable to {SET_STATE; 0,0; 65534} (up) or {SET_STATE; 100,0; 65534} (down), respectively. This allows the blinds to drive up and down. A short press (< 1 second) while driving the blinds will set the network variable to {SET_STOP; 0,0; 0,00}. This allows stopping the blinds at the current position. A short press (< 1 second) will set the network variable to {SET_UP; 127,5; 15} (up) or {SET_DOWN; 127,5; 15} (down), respectively. This allows rotating the blades of the sunblinds (open/close).

8.3 BACnet Interface (LDALI-20X only)

Similar to the CEA-709 interface, the BACnet interface allows controlling DALI ballasts and access information from DALI ballasts and sensors via BACnet objects and properties. Additional properties allow configuring the built-in light controller applications (constant light, staircase lighting, etc.).

The objects can be grouped in the following categories:

DALI ballasts

- *Analog Output objects (see Section 8.3.3.10)*:
  - Control output light level.
  - Configure DALI ballast parameters.
  - Configure light application parameters (staircase lighting, etc.).
- *Analog Input objects (see Section 8.3.3.11)*: Output light level feedback.

DALI groups

- *Analog Output objects (see Section 8.3.3.10)*:
  - Control output light level.
  - Configure light application parameters (staircase lighting, etc.).
- *Analog Input objects (see Section 8.3.3.11)*: Output light level feedback.
- *Multi-State Output objects (see Section 8.3.3.12)*: Scene control.
- *Accumulator objects (see Section 8.3.3.13)*: Accumulated energy usage (calculated).
- *Analog Input objects (see Section 8.3.3.14)*: Status and health state of group.

DALI channels

- *Analog Output objects (see Section 8.3.3.10)*:
    - Control output light level.
    - Configure light application parameters (staircase lighting, etc.).
- *Analog Input objects (see Section 8.3.3.11)*: Output light level feedback.
- *Multi-State Output objects (see Section 8.3.3.12)*: Scene control.
- *Accumulator objects (see Section 8.3.3.13)*: Accumulated energy usage (calculated).
- *Analog Input objects (see Section 8.3.3.14)*: Status and health state of channel.

DALI sensors

- *Analog Input objects (see Section 8.3.4.2)*: Lux level sensor value.
- *Binary input objects (see Section 8.3.4.3)*: Occupancy sensor value.

Constant Light Controller

- *Loop objects (see Section 8.3.5.10)*

All properties not specifically mentioned in the following sections behave as described by the BACnet standard.

8.3.1 Interface Version

The BACnet interface can be used in two different modes:

Firmware version 1.x compatible interface

This interface is provided for backward compatibility with projects set up with a L-DALI with firmware version before 2.0.

Full featured interface

This interface is recommended for new projects.

The main difference between the two interface versions is the instance number layout. The "Full featured interface" offers better human readability. Further, new features in future firmware versions may be only available via the full featured interface.

The interface version can be configured via the LINX Configurator project settings (see Section 7.16.5).

8.3.2 Device Object

The BACnet interface provides one device object as shown in Table 26. The following Sections describe the device object's properties in detail, subsuming related properties in a single Section in order to provide a coherent overview.

Property Identifier	Property Datatype	Conformance Code
Object_Identifier	BACnetObjectIdentifier	R
Object_Name	CharacterString	R
Object_Type	BACnetObjectType	R
Vendor_Name	CharacterString	R
Vendor_Identifier	Unsigned16	R
Model_Name	CharacterString	R
Firmware_Revision	CharacterString	R
Application_Software_Version	CharacterString	R
Location	CharacterString	W
Description	CharacterString	W
Protocol_Version	Unsigned	R
Protocol_Revision	Unsigned	R
Protocol_Services_Supported	BACnetServicesSupported	R
Protocol_Object_Types_Supported	BACnetObjectTypesSupported	R
Object_List	BACnetARRAY[N] of BACnetObjectIdentifier	R
Database_Revision	Unsigned	R
Max_APDU_Length_Accepted	Unsigned	R
Segmentation_Supported	BACnetSegmentation	R
Max_Segments_Accepted	Unsigned	R
APDU_Segment_Timeout	Unsigned	W
APDU_Timeout	Unsigned	W
Number_Of_APDU_Retries	Unsigned	W
Max_Master	Unsigned(1..127)	R
Max_Info_Frames	Unsigned	R
System_Status	BACnetDeviceStatus	R
Device_Address_Binding	List of BACnetAddressBinding	R
Active_COV_Subscriptions	List of BACnetCOVSubscription	R
UTC_Offset	Integer	W
Daylight_Savings_Status	Boolean	R
Local_Date	Date	R
Local_Time	Time	R
Time_Synchronization_Recipients	List of BACnetRecipient	W
UTC_Time_Synchronization_Recipiens	List of BACnetRecipient	W
Time_Synchronization_Interval	Unsigned	W
Align_Interval	Boolean	W
Interval_Offset	Unsigned	W
Configuration_Files	BACnetARRAY[N] of BACnetObjectIdentifier	R
Last_Restore_Time	BACnetTimeStamp	R

Table 26: Properties of the Device Object.

8.3.2.1 Device Name and ID

The following properties of the Device object, which are part of every BACnet object, identify the device uniquely.

Object_Identifier (Read-Only). This property, of type *BACnetObjectIdentifier*, is a numeric code that is used to identify the object. For the Device object, the object identifier must be unique internetwork-wide.

The *Object_Type* part of the *Object_Identifier* of the Device object is 8 (= DEVICE). The instance part of the *Object_Identifier* of the Device object is configurable via the configuration UI (see Section 5.2.12). The default value is 17800.

Object_Name (Read-Only). The name of the object. The value of *Object_Name* of the Device object is configurable via the configuration UI (see Section 5.2.12). For the Device object, this name shall be unique within the BACnet internetwork.

Object_Type (Read-Only). The object's type. For the Device object, the value of this property is 8 (= DEVICE).

8.3.2.2 Device Information

A whole set of properties provides general purpose information about the device.

Vendor_Name (Read-Only). The value of this property is "LOYTEC electronics GmbH".

Vendor_Identifier (Read-Only). A numerical value identifying the BACnet vendor. The value of this property is 178.

Model_Name (Read-Only). The value of this property is equal to the product code of the device ("LDALI-ME204").

Firmware_Revision (Read-Only). The value of this property gives the current BACnet module version used on the device.

Application_Software_Version (Read-Only). The value of this property gives the build date and the version of the current application on the device.

Location (Read-Writable). A string intended to be used to describe the physical location of the device, e.g., "1st floor". This property can be set via the configuration UI (see Section 5.2.12). The default value is "unknown".

Description (Read-Only). A string intended to be used to describe the device's purpose. This property can be changed via the configuration UI (see Section and 5.2.12).

Protocol_Version (Read-Only). The BACnet protocol version supported by the device. The value of this property is 1.

Protocol_Revision (Read-Only). The BACnet protocol revision of the BACnet version supported by the device. The value of this property is 6.

Protocol_Services_Supported (Read-Only). A string of bits marking which BACnet services can be executed by the device. For a detailed list of the BACnet services supported, please refer to the product's PICS document.

Protocol_Object_Types_Supported (Read-Only). A string of bits identifying which BACnet object types are supported by the device. For a detailed list of supported object types, please refer to the product's PICS document.

8.3.2.3 Object Database

The following properties provide information about the BACnet objects contained in the device.

Object_List (Read-Only). This property holds a *BACnetARRAY* of object IDs (object type, object instance pairs), one object ID for each object within the device that is accessible through BACnet services.

Database_Revision (Read-Only). This property, of type *Unsigned*, is a logical revision number for the device's object database. It is incremented when an object is created, an object is deleted, an object's name is changed, an object's Object_Identifier property is changed, or a restore is performed.

8.3.2.4 Protocol Parameters

BACnet protocol parameters are accessible via the properties listed below.

Max_APDU_Length_Accepted (Read-Only). The maximal size of an APDU (Application Protocol Data Unit) accepted by the device. The value of this property is 487 if BACnet MS/TP is used and 1476 if BACnet/IP is used.

Segmentation_Supported (Read-Only). The value of this property indicates whether and which kind of segmentation is supported by a device. The value of this property is SEGMENTED_BOTH.

Max_Segments_Accepted (Read-Only). The maximum numbers of segments accepted by a device. The value of this property is 16.

APDU_Segment_Timeout (Read-Writable). Timeout in milliseconds allowed between segments. The value of this property is 2000 milliseconds by default. On MS/TP networks, this value should be increased to 40000 (40 sec).

APDU_Timeout (Read-Writable). Time in milliseconds the device waits for an answer before retrying or giving up on a request; also see *Number_Of_APDU_Retries*. The value of this property is 3000 milliseconds. On MS/TP networks, this value should be increased to 60000 (1 min).

Number_Of_APDU_Retries (Read-Writable). The number of times the device will try to re-send a packet before giving up on a request; also see *APDU_Timeout*. The value of this property is 3 by default.

Max_Master (Read-Writable). This property is only present if BACnet MS/TP is enabled. It defines maximal MS/TP MAC number at which the device expects an MS/TP master. The value of this property is configurable via the configuration UI (see Section 5.2.12) and must be in the range 1-127. The default value of this property is 127.

Max_Info_Frames (Read-Writable). This property is only present if BACnet MS/TP is enabled. It defines the maximal number of MS/TP packets the device can send when it holds the MS/TP token. Increasing this value will increase latency on the MS/TP network. The value of this property is configurable via the configuration UI (see Section 5.2.12). The default value of this property is 1.

8.3.2.5 Diagnostics

Several properties provide run-time information about the device.

System_Status (Read-Only). The value of this property is always OPERATIONAL.

Device_Address_Binding (Read-Only). This property contains a list of bindings between BACnet device instance numbers (the instance number part of the Device object ID) and BACnet addresses. This property tells a user which BACnet address the device will actually use when trying to communicate with another device known only by its device instance number. This information can be helpful when diagnosing network configuration problems.

Important! ***A BACnet address consists of the BACnet network number, which is 0 for the local network, and the BACnet MAC address of the device.***

In particular problems exist, if two or more devices in the network have been wrongly assigned the same device instance number. In this case two BACnetAddressBinding entries with the same instance number but different BACnet addresses will be listed—provided the ambiguous instance number is in some way required by the device (e.g., by a client mapping).

Important! ***Bindings between device instance numbers and BACnet addresses are only listed in Device_Address_Binding if they are actually required by a given configuration, and are currently known or ambiguous.***

Active_COV_Subscriptions (Read-Only). This property lists currently active COV subscriptions. Each entry of type *BACnetCOVSubscription* provides information about the recipient address, the monitored property ID, whether notification are confirmed or unconfirmed, the remaining time of the subscription, and optionally the COV increment.

Whenever the device receives a COV subscription via one of the services SubscribeCOV or SubscribeCOVProperty, a new entry is added to the list or an existing entry is updated (re-subscription). An entry is removed from the list when a subscription terminates, either because it times out or because it was actively unsubscribed by the subscriber.

8.3.2.6 Date & Time

The device's time and date are exposed to the network via the following set of properties.

UTC_Offset (Read-Writable). This *Integer* value specifies the time difference between local time and UTC in minutes. The value of this property is configurable via the configuration UI (see Section 5.2.11).

Important! ***Note that UTC_Offset is relative to local time and not relative to UTC, i.e., a time zone offset of GMT+1 (Berlin, Paris, Vienna) corresponds to UTC_Offset being set to -60 (minutes).***

Daylight_Savings_Status (Read-Only). This *Boolean* value indicates whether (TRUE) or not (FALSE) daylight saving correction of the local time is currently active. The daylight saving scheme is configurable via the configuration UI (see Section 5.2.11).

Local_Date (Read-Only). The current date according to the device's clock. The value of this property can be changed via the configuration UI (see Section 5.2.11).

Local_Time (Read-Only). The current time according to the device's clock. The value of this property can be changed via the configuration UI (see Section 5.2.11).

8.3.2.7 Time Master

The device can serve as a BACnet time master, i.e., it can issue TimeSynchronization and UTCTimeSynchronization request on time synchronization events. A time synchronization event occurs after rebooting, when the device's clock changes, or, if so configured, the event is generated periodically. The following properties are used to configure the time master. Use a BACnet operator workstation to write these properties over the BACnet network.

Time_Synchronization_Recipients (Read-Writable). This list of recipients will receive TimeSynchronization requests on time synchronization events. A recipient is either specified by its device ID (the object ID of its Device object), or its BACnet address. By default, this list is empty.

UTC_Time_Synchronization_Recipients (Read-Writable). This list of recipients will receive UTCTimeSynchronization requests on time synchronization events. A recipient is either specified by its device ID (the object ID of its Device object), or its BACnet address. By default, this list is empty.

Time_Synchronization_Interval (Read-Writable). The *Unsigned* value of this property specifies the time interval in minutes in which periodic time synchronization events are created. If set to zero, no periodic time synchronization events are generated.

The actual clock time at which periodic time synchronization events are generated is determined by the properties *Time_Synchronization_Interval*, *Align_Interval*, and *Interval_Offset*; Table 27 outlines how these properties interact.

Time_Synchronization_Interval	Align_Intervals	Periodc Time Synchronization Event At…
Multiple of 1440 (minutes), i.e., multiple of one day	TRUE	*Interval_Offset* minutes after midnight, every (*Time_Synchronization_Interval*/1440) days
Multiple of 60 (minutes) but *not* multiple of 1440 (minutes), i.e., multiple of one hour	TRUE	*Interval_Offset* minutes from the current* hour, every (*Time_Synchronization_Interval*/60) hours
Multiple of 1440 (minutes), i.e., multiple of one day	FALSE	*Interval_Offset* minutes from the current* minute, every (*Time_Synchronization_Interval*/1440) days
Multiple of 60 (minutes), but *not* multiple of 1440 (minutes), i.e., multiple of one hour	FALSE	*Interval_Offset* minutes from the current* minute, every (*Time_Synchronization_Interval*/60) hours
Neither multiple of 60 or 1440, but greater than zero	TRUE or FALSE	*Interval_Offset* minutes from the current* minute, every *Time_Synchronization_Interval* minutes
Zero	TRUE or FALSE	Never

Table 27: Periodic time synchronization events are parameterized by the properties *Time_Synchronization_Interval*, *Align_Interval*, and *Interval_Offset*.
* Current hour or minute refers to the hour or minute at which one of the properties *Time_Synchronization_Interval*, *Align_Interval*, and *Interval_Offset* is written, e.g., the hour or minute the device completes the boot process or one of these properties is modified via BACnet services.

By default, the value of *Time_Synchronization_Interval* is 1440 (minutes), i.e., one day.

Align_Intervals (Read-Writable). The *Boolean* value of this property determines whether or not periodic time synchronization events shall be anchored at the start of a day or hour (TRUE) or not (FALSE), provided *Time_Synchronization_Interval* is a multiple of a day (1440 minutes) or hour (60 minutes). Table 27 details on how this property influences generating periodic time synchronization events. The default value of this property is TRUE.

Interval_Offset (Read-Writable). While *Time_Synchronization_Interval* specifies the period in which time synchronization events are generated, the *Unsigned* value of this property determines the point of time in minutes within this interval at which the time synchronization event is actually triggered. If the value of *Interval_Offset* is larger than the value of *Time_Synchronization_Interval*, the remainder of *Interval_Offset* divided by *Time_Synchronization_Interval* is used. The default value of this property is 0.

8.3.2.8 Backup & Restore

The following properties are related to backup & restore procedures.

Configuration_Files (Read-Only). The contents of this property is an array of object IDs of File objects that can backed-up or restored during a BACnet backup or restore procedure. Outside a BACnet backup or restore procedure, this property is empty. After a BACnet backup or restore procedure has been initiated, it contains the object ID *(File, 0),* i.e., the File object whose instance number is 0.

Last_Restore_Time (Read-Only). The *BACnetTimeStamp* of the last restore procedure.

8.3.3 Light Output Objects

There are three different types of light output objects:

- Objects related to DALI lamps (64 per channel).
- Objects related to DALI groups (16 per channel).
- Objects related to the DALI channels (1 per channel).

8.3.3.1 Control

The L-DALI offers different methods to control lamps.

Individual Control

Each lamp can be controlled individually via the *Present_Value* property of the corresponding Analog Output objects (see Section 8.3.3.10). Any new value received is transmitting to the corresponding DALI lamp in the dimming mode configured with the *Mode* property of the object (property *Fade_Time* or *Ramp_Rate* respectively is used to determine dim speed).

Group Control

Lamps belonging to a group can be controlled together by the following means:

- ***Dimming:*** Any new value received via the *Present_Value* property of the corresponding Analog Output objects (see Section 8.3.3.10) is transmitting to the corresponding DALI group in the dimming mode configured with the *Mode* property of the object (property *Fade_Time* or *Ramp_Rate* respectively is used to determine dim speed).
- ***Scene control:*** Using the *Present_Value* property of the corresponding Multi-State Output objects (see Section 8.3.3.12) up to 16 DALI scenes can be saved and recalled. The states "*STORE SCENE x*" store the current light levels of all lamps of the group in the specified scene number 'x'. By setting the state to "*GO TO SCENE x*" previously stored scenes can be recalled. Scenes can be deleted by a using the states "*REMOVE SCENE x*". When a new scene is selected the lamp values are adjusted in fading mode (property *Fade_Time* is used).

Channel Control

All lamps on the same DALI channel can be controlled together by the following means:

- ***Dimming:*** The *Present_Value* property of the corresponding Analog Output objects works in the same way as for lamps and groups but affects all lamps on a channel.

- ***Scene control:*** The *Present_Value* property of the corresponding Multi-State Output objects works in the same way as for groups but affects all lamps on a channel.

8.3.3.2 Maximum and Minimum Light Level

For a dimmable lamp the maximum and minimum light level of a lamp can be configured via the properties *Min_Level* and *Max_Pres_Value* of the corresponding Analog Output object.

For a non-dimmable lamp set *Min_Level* = *Max_Pres_Value* = 100%.

8.3.3.3 Timing Parameters

Figure 228 shows the behavior of the lamp actuator if the light is switched on/off via one of the Analog Output objects. When the ON command is received, the lamp is switched to the specified value after the time *On_Delay_Time* has expired. When the lamp is already on and a new ON command is received the lamp is switched to the new value immediately. If the lamp is on and an OFF command is received the lamp is switched off after the time *Off_Delay_Time*.

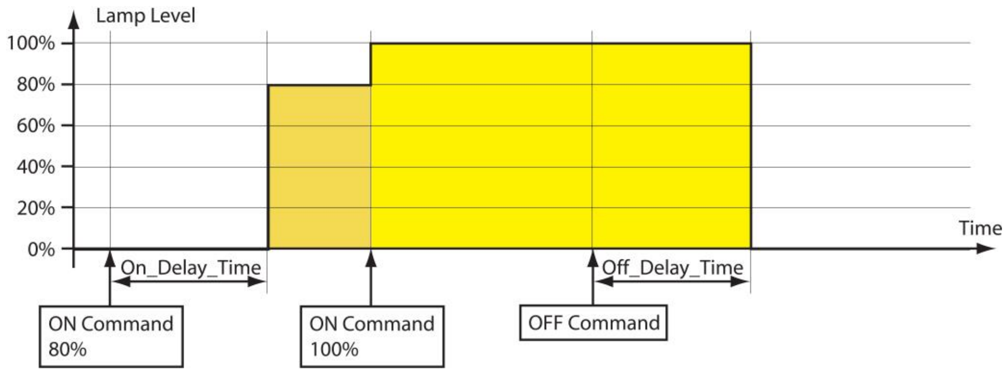


Figure 228: Timing Parameters.

For staircase lighting an automatic cutoff can be configured in *Auto_Off_Time* as shown in Figure 229. When this time expires, the lamp switches off automatically. To warn the user that the light is going to turn off, an off-warning delay (*Warn_Delay*) can be configured. During this time the light is dimmed to 50% of the previous level. For non-dimmable lights (*Min_Level*=*Max_Pres_Value*) the light is blinking.

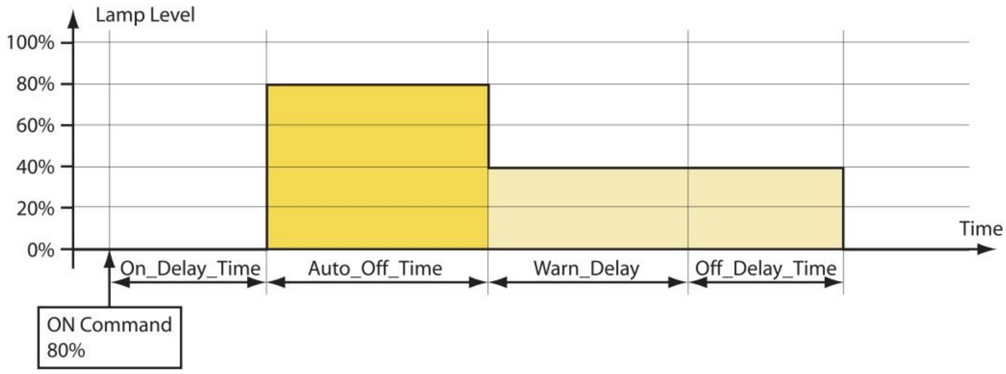


Figure 229: Timing Parameters with *Auto_Off_Time* and *Warn_Delay.*

The behavior of the auto-off timer can be modified with the property *Auto_Off_Mode*. If *Auto_Off_Mode* is set to ENABLE_RESTART or ENABLE_RESTART_BREAK, the auto-off timer can be restarted by a new ON command (see Figure 230). If *Auto_Off_Mode* is set ENABLE_BREAK or ENABLE_RESTART_BREAK the auto-off timer can be stopped before the time has expired by switching the light off via the corresponding *Present_Value* property (see Figure 231).

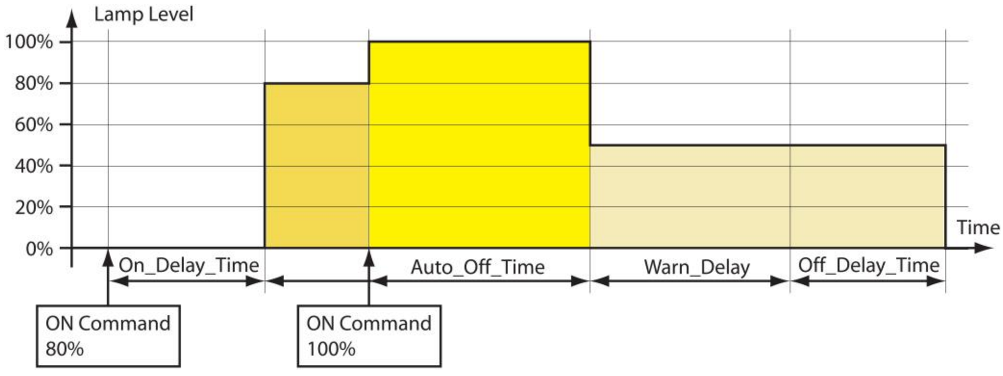


Figure 230: Timing Parameters (restart enabled).

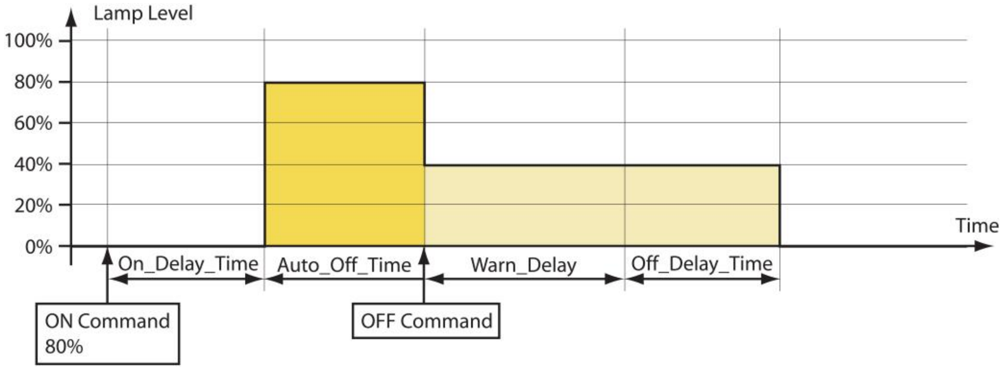


Figure 231: Timing Parameters (break enabled).

8.3.3.4 Feedback

The DALI lamp, group and channel each report the current state via the *Present_Value* of a feedback Analog Input (see Section 8.3.3.11).

All Analog Output objects report if they are in manual override mode by setting the OVERRIDDEN flag in the *Status_Flags* property.

8.3.3.5 Emergency Lights

The following additional, emergency light related properties are available via the Channel Health Analog Input object (see Section 8.3.3.14):

- *Battery_Failure*: Report battery failure for each emergency light ballast.
- *Function_Test_Failure*: Report whether last function test failed for each emergency light ballast.
- *Duration_Test_Failure*: Report whether last function test failed for each emergency light ballast.

8.3.3.6 Status Monitoring

The L-DALI monitors the DALI lamps and reports any failure in the property *Device_Failure* of the Channel Health Analog Input object (see Section 8.3.3.14). In this property of type BIT STRING each bit corresponds to the lamp with the same index. In case of a failure the bit is set. Similar properties *Lamp_Failure* and *Ballast_Failure* are used to signal a lamp or a ballast failure reported by a ballast for each DALI ballast.

A device failure is reported if either

- bus power for the related DALI channel fails,
- the DALI ballast is not reachable via the DALI channel, or
- the DALI ballast reports a failure (e.g. lamp failure) in its DALI status register.

The *Present_Value* properties of the group and channel related Analog Input objects (see Section 8.3.3.14) reflect the percentage of failed ballasts in this group or channel respectively.

8.3.3.7 Statistics

The L-DALI reports the run hours and energy consumptions of the DALI lamps.

Run Hours

The L-DALI determines how long each DALI lamp was switched on. The value is reported via the property *Elapsed_Active_Time* of the Analog Output object for each lamp, group, and channel (see Section 8.3.3.10). If a lamp is replaced the run hours can be initialized by writing 0 to this property. The time and date of the last reset is stored in the property *Time_Of_Active_Time_Reset*.

The group related object reports the maximum run hours of all lamps in the groups via the property *Elapsed_Active_Time*.

The channel related object reports the maximum run hours of all lamps on the channel via the property *Elapsed_Active_Time*.

Energy Consumption

The L-DALI device can calculate the energy consumption of the DALI lamps. For that purpose it needs to know the nominal power of all connected DALI lamps. If the configuration property *Nominal_Power* of a lamp related Analog Output object (see Section 8.3.3.10) is set to 0 (AUTO), the L-DALI tries to obtain the energy consumption from the DALI device. However, not all DALI ballasts support this proprietary extension. In this case the *Nominal_Power r* has to be configured with the nominal power of the ballast.

To reduce the number of objects the energy consumption is reported not separately for each lamp but for a whole group or channel. Accumulator objects (see Section 8.3.3.13) are used to report the sum of energy consumed by all lamps in a group or on a channel respectively. The energy consumption can be reset by writing to the *Value_Set* property of the Accumulator object

8.3.3.8 Burn-In Function

Some lamps require a burn-in time during which they must not be dimmed. The burn-in mode can be activated/deactivated via the Web-UI. The burn-in time is defined by the property *Burn_In_Time* available via Analog Output objects corresponding to DALI channels (see Section 8.3.3.10). During this time the lamps will only be switched to on (100%) or off (0%) but not dimmed.

8.3.3.9 Fail Safe Functions

The light level a lamp adopts after power-up is defined in the property *Power_On_Level*. The light level a lamp adopts in case of a DALI system failure is defined in the configuration property *System_Failure_Level*. Both are available via Analog Output objects corresponding to DALI ballasts (see Section 8.3.3.10).

8.3.3.10 Analog Output Object – Control DALI ballast, group, or channel

This object type is used to control the dim level of DALI ballasts (see Table 28), DALI groups (see Table 29), or DALI channel (see Table 30). For DALI devices it also provides access to all its DALI parameters. Further, parameters for light controller applications are available (e.g. staircase lighting, on and off delays, off warning etc.).

Property Identifier	Property ID	Property Datatype	Conformance Code
Object_Identifier	75	BACnetObjectIdentifier	R
Object_Name	77	CharacterString	R
Object_Type	79	BACnetObjectType	R
Present_Value	85	REAL	W
Description	28	CharacterString	W
Device_Type	31	CharacterString	R
Status_Flags	111	BACnetStatusFlags	R
Event_State	36	BACnetEventState	R
Reliability	103	BACnetReliability	R
Out_Of_Service	81	BOOLEAN	W
Units	117	BACnetEngineeringUnits	R
Min_Pres_Value	69	REAL	R
Max_Pres_Value	65	REAL	W
Priority_Array	87	BACnetPriorityArray	R
Relinquish_Default	104	REAL	W
COV_Increment	22	REAL	W
Time_Delay	113	Unsigned	W
Notification_Class	17	Unsigned	W
High_Limit	45	REAL	W
Low_Limit	59	REAL	W
Deadband	25	REAL	W
Limit_Enable	52	BACnetLimitEnable	W
Event_Enable	35	BACnetEventTransitionBits	W
Acked_Transitions	0	BACnetEventTransitionBits	W
Notify_Type	72	BACnetNotifyType	W
Event_Time_Stamps	130	BACnetARRAY[3] of BACnetTimeStamp	R
Profile_Name	168	CharacterString	R
Power_On_Level	512	REAL	W
System_Failure_Level	513	REAL	W
Fade_Time	514	Unsigned	W
Ramp_Rate	515	REAL	W
Min_Level	516	REAL	W
Groups	517	BIT STRING	W
Nominal_Power	518	REAL	W
Dim_Mode	520	BACnetDimMode	W
On_Delay_Time	521	Unsigned	W
Off_Delay_Time	522	Unsigned	W
Warn_Delay	523	Unsigned	W
Auto_Off_Time	524	Unsigned	W
Auto_Off_Mode	526	BACnetAutoOffMode	W
Elapsed_Active_Time	527	Unsigned	W
Time_Of_Active_Time_Reset	528	BACnetDateTime	R

Table 28: Properties of the Analog Output object used to control DALI ballasts.

Property Identifier	**Property ID**	**Property Datatype**	**Conformance Code**
Object_Identifier	75	BACnetObjectIdentifier	R
Object_Name	77	CharacterString	R
Object_Type	79	BACnetObjectType	R
Present_Value	85	REAL	W
Description	28	CharacterString	W
Device_Type	31	CharacterString	R
Status_Flags	111	BACnetStatusFlags	R
Event_State	36	BACnetEventState	R
Reliability	103	BACnetReliability	R
Out_Of_Service	81	BOOLEAN	W
Units	117	BACnetEngineeringUnits	R
Min_Pres_Value	69	REAL	R
Max_Pres_Value	65	REAL	W
Priority_Array	87	BACnetPriorityArray	R
Relinquish_Default	104	REAL	W
COV_Increment	22	REAL	W
Time_Delay	113	Unsigned	W
Notification_Class	17	Unsigned	W
High_Limit	45	REAL	W
Low_Limit	59	REAL	W
Deadband	25	REAL	W
Limit_Enable	52	BACnetLimitEnable	W
Event_Enable	35	BACnetEventTransitionBits	W
Acked_Transitions	0	BACnetEventTransitionBits	W
Notify_Type	72	BACnetNotifyType	W
Event_Time_Stamps	130	BACnetARRAY[3] of BACnetTimeStamp	R
Profile_Name	168	CharacterString	R
Dim_Mode	520	BACnetDimMode	W
On_Delay_Time	521	Unsigned	W
Off_Delay_Time	522	Unsigned	W
Warn_Delay	523	Unsigned	W
Auto_Off_Time	524	Unsigned	W
Auto_Off_Mode	526	BACnetAutoOffMode	W
Elapsed_Active_Time	527	Unsigned	W
Time_Of_Active_Time_Reset	528	BACnetDateTime	R

Table 29: Properties of the Analog Output object used to control DALI groups.

Property Identifier	Property ID	Property Datatype	Conformance Code
Object_Identifier	75	BACnetObjectIdentifier	R
Object_Name	77	CharacterString	R
Object_Type	79	BACnetObjectType	R
Present_Value	85	REAL	W
Description	28	CharacterString	W
Device_Type	31	CharacterString	R
Status_Flags	111	BACnetStatusFlags	R
Event_State	36	BACnetEventState	R
Reliability	103	BACnetReliability	R
Out_Of_Service	81	BOOLEAN	W
Units	117	BACnetEngineeringUnits	R
Min_Pres_Value	69	REAL	R
Max_Pres_Value	65	REAL	W
Priority_Array	87	BACnetPriorityArray	R
Relinquish_Default	104	REAL	W
COV_Increment	22	REAL	W
Time_Delay	113	Unsigned	W
Notification_Class	17	Unsigned	W
High_Limit	45	REAL	W
Low_Limit	59	REAL	W
Deadband	25	REAL	W
Limit_Enable	52	BACnetLimitEnable	W
Event_Enable	35	BACnetEventTransitionBits	W
Acked_Transitions	0	BACnetEventTransitionBits	W
Notify_Type	72	BACnetNotifyType	W
Event_Time_Stamps	130	BACnetARRAY[3] of BACnetTimeStamp	R
Profile_Name	168	CharacterString	R
Burn_In_Time	519	Unsigned	W
Dim_Mode	520	BACnetDimMode	W
On_Delay_Time	521	Unsigned	W
Off_Delay_Time	522	Unsigned	W
Warn_Delay	523	Unsigned	W
Auto_Off_Time	524	Unsigned	W
Auto_Off_Mode	526	BACnetAutoOffMode	W
Elapsed_Active_Time	527	Unsigned	W
Time_Of_Active_Time_Reset	528	BACnetDateTime	R

Table 30: Properties of the Analog Output object used to control DALI channels.

Object_Identifier (Read-Only)

The default instance number is ABCC, where “A” is 0 for the objects associated with DALI devices and 1 for the objects associated with DALI groups, and 2 with DALI channels, “B” is the DALI interface number (0-3) in decimal, and “CC” is the DALI device number (00-63) or the DALI group address (00-15) respectively in decimal (00 for DALI channels).

Examples: Ballast 3 on channel 2 has the instance number 103, Group 5 on channel 1 has the instance number 1005, and channel 4 has the instance number 2300.

In compatibility mode the instance number is 0x0ABBCC, where “A” is 0 for the objects associated with DALI devices and 1 for the objects associated with DALI groups, and 2 with DALI channels, “BB” is the DALI interface number (0-3) in hex, and “CC” is the DALI device number (0-63) or the DALI group address (0-15) respectively in hex. Examples: Ballast 3 on channel 2 has the instance number 0x000303 (771), Group 5 on channel 1 has the instance number 0x010205 (66053), and channel 4 has the instance number 0x020300 (131840).

Object_Name (Read-Only)

This property holds the name of the lamp/group/channel. It can be changed via the Web-UI.

Object_Type (Read-Only)

The value of this property is ANALOG_OUTPUT (1).

Present_Value (Read/Write)

This property, of type REAL, indicates the linearized percentage (0..100.00%) of the device’s desired light output; 0 being off, 1 being dimmest, 100 brightest. It is used to dim the corresponding DALI ballast, group, or channel.

Depending on the *Dim_Mode* property, a write to this property results in a DALI “DIRECT ARC POWER CONTROL” command (fade) or a series of UP or DOWN commands to the device, group or channel respectively.

Description (Read/Write)

This property can be configured via BACnet. It serves documentation purposes only.

Device_Type (Read-Only)

This property contains the DALI device type (e.g. “*low voltage halogen lamp*”) for objects mapping to DALI devices, “*DALI group*” for objects mapping to DALI groups and “*DALI channel*” for objects mapping to DALI channels.

Status_Flags (Read-Only)

Of the four flags the flags IN_ALARM, FAULT, and OUT_OF_SERVICE are linked to the state of the corresponding properties *Event_State*, *Reliability* and *Out_Of_Service* as described by the standard.

OVERRIDDEN	Logical TRUE (1) if the associated DALI channel is controlled via the button interface (see Section 4.5.2), FALSE otherwise.

Reliability (Read-Only)

The *Reliability* property for this object type may have any of the following values:

NO_FAULT_DETECTED	No fault was detected.
NO_OUTPUT	A lamp failure was detected (for groups and channels if at least one lamp failure in the group/on the channel was detected).

COMMUNICATION_FAILURE	Ballast offline (for groups and channels if at least one ballast is offline in the group/on the channel).
UNRELIABLE_OTHER	Other failure detected (e.g. DALI device type specific error, for groups and channels if at least one ballast in the group/on the channel reports a failure)

Units (Read-Only)

The value of this property is 98 (percent).

Min_Pres_Value (Read-Only)

The value of this property is 0.

Max_Pres_Value (Read/Write)

For objects associated to DALI lamps this property is read/write. Its maximum value is 100. It is mapped to the DALI register MAX_LEVEL of the corresponding ballast.

For DALI groups and channels this value is always 100 and read only.

Profile_Name (Read-only)

This property has the following values:

- DALI lamps: "*178-http://www.loytec.com/xsd/DALILampActuator/v1*"
- DALI groups: "*178-http://www.loytec.com/xsd/DALIGroupActuator/v1*"
- DALI channels: "*178-http://www.loytec.com/xsd/DALIChannelActuator/v1*"

Power_On_Level (Read/Write)

The value of this proprietary property (property ID 512) corresponds to the "POWER ON LEVEL" in the DALI device. A value of NaN corresponds to MASK.

Only present if the object is associated with a DALI lamp.

System_Failure_Level (Read/Write)

The value of this proprietary property (property ID 513) corresponds to the "SYSTEM FAILURE LEVEL" in the DALI device. A value of NaN corresponds to MASK.

Only present if the object is associated with a DALI lamp.

Fade_Time (Read/Write)

The value of this proprietary property (property ID 514) corresponds to the "FADE TIME" in the DALI device. It gives the time in milliseconds. The FADE TIME closest to the written value will be used.

Only present if the object is associated with a DALI lamp.

Ramp_Rate (Read/Write)

The value of this proprietary property (property ID 515) corresponds to the "FADE RATE" in the DALI device. It gives the fade rate in percent per seconds for ramping. The FADE RATE closest to the written value will be used.

Only present if the object is associated with a DALI lamp.

Min_Level (Read/Write)

The value of this proprietary property (property ID 516) corresponds to "MIN LEVEL" in the DALI device. It must be in the range 0 < Min_Level <= 100.

Only present if the object is associated with a DALI lamp.

Groups (Read/Write)

The value of this proprietary property (property ID 517) is a 16-bit wide BIT STRING. It corresponds to the "GROUPS" of the DALI device. It defines the group membership of a lamp. Each bit corresponds to the group of the same index. If the bit is set, the lamp is member of the group.

Only present if the object is associated with a DALI lamp.

Nominal_Power (Read/Write)

This proprietary property (property ID 518) is used to specify the nominal power of a DALI ballast. A value of 0 is used to specify AUTO. In this case the nominal power is read from the ballast (if available).

Only present if the object is associated with a DALI lamp.

Burn_In_Time (Read/Write)

This proprietary property (property ID 519) is used to configure the burn in time for the channel. The burn-in mode can be activated/deactivated via the Web-UI.

Only present if the object is associated with a DALI channel.

Dim_Mode (Read/Write)

The value of this proprietary property (property ID 520) is used to specify whether to use ramping (1) or fading (0) when changing the current dim level by writing to the *Present_Value* property.

In case of ramping a constant dim rate is used. Thus, the duration of the dim process depends on the difference between start and target value.

Fading is based on a constant dim duration. Thus, the duration of the dim process does not depend on the difference between start and target value.

On_Delay_Time (Read/Write)

This proprietary property (property ID 521) is used to configure an on-delay time in seconds.

Off_Delay_Time (Read/Write)

This proprietary property (property ID 522) is used to configure an off-delay time in seconds.

Warn_Delay (Read/Write)

This proprietary property (property ID 523) is used to configure an off-warning time in seconds. It defines the time during which the user will be notified that the light will be switched off shortly. During this time, the light is dimmed to 50% of the current level. Lamps which cannot be dimmed (*Min_Level* = *Max_Pres_Value*) blink during this time.

Auto_Off_Time (Read/Write)

This proprietary property (property ID 524) is used to configure an auto-off time in seconds. It determines the delay after which the lamp output is switched off automatically. The timer is started after receiving a lamp ON command via the *Present_Value* property

The property *Auto_Off_Mode* modifies the behavior of the auto-off timer.

Auto_Off_Mode (Read/Write)

The value of this proprietary property (property ID 526) determines the behavior of the auto-off timer. Possible values are:

DISABLE_ALL (0)	ENABLE_RESTART and ENABLE_BREAK not set.
ENABLE_BREAK (1)	The auto-off-timer can be interrupted by writing to the *Present_Value*.
ENABLE_RESTART (2)	The auto-off-timer is restarted whenever the *Present_Value* is written.
ENABLE_RESTART_BREAK (3)	ENABLE_RESTART and ENABLE_BREAK set.

Elapsed_Active_Time (Read/Write)

The value of this proprietary property (property ID 527) gives the accumulated time in seconds, which the corresponding DALI lamp was switched on. For groups or channels it gives the maximum time of all lamps in the group/channel. When writing 0 to this property the value is reset. For groups and channels the value of all lamps in the group/channel are reset.

Time_Of_Active_Time_Reset (Read-Only)

The value of this proprietary property (property ID 528) give the time and date the *Elapsed_Active_Time* property was last reset.

Intrinsic reporting

This object supports all properties required for intrinsic reporting. Per default only fault alarms are enabled.

8.3.3.11 Analog Input Object – Feedback from DALI ballast, group, or channel

This object is used to give feedback on the current dim level of DALI ballasts, DALI groups, or DALI channels (see Table 31).

Property Identifier	Property ID	Property Datatype	Conformance Code
Object_Identifier	75	BACnetObjectIdentifier	R
Object_Name	77	CharacterString	R
Object_Type	79	BACnetObjectType	R
Present_Value	85	REAL	R
Device_Type	31	CharacterString	R
Status_Flags	111	BACnetStatusFlags	R
Event_State	36	BACnetEventState	R
Reliability	103	BACnetReliability	R
Out_Of_Service	81	BOOLEAN	W
Units	117	BACnetEngineeringUnits	R
Min_Pres_Value	69	REAL	R
Max_Pres_Value	65	REAL	R
COV_Increment	22	REAL	W
Profile_Name	168	CharacterString	R

Table 31: Properties of the Analog Input object providing feedback from DALI ballasts, groups, and channels.

Object_Identifier (Read-Only)

The instance number is equal to the instance number of the corresponding Analog Output object (see Section 8.3.3.10).

Object_Name (Read-Only)

This property holds the value of the *Object_Name* of the associated Analog Output object with the addition "*Feedback*".

Object_Type (Read-Only)

The value of this property is ANALOG_INPUT (0).

Present_Value (Read-Only)

This property, of type REAL, indicates the linearized percentage (0..100.00%) of the DALI device's, group's or channel's current light output.

Device_Type (Read-Only)

This property contains the value "*Actual Dim Level*".

Status_Flags (Read-Only)

Of the four flags the flags IN_ALARM, FAULT, and OUT_OF_SERVICE are linked to the state of the corresponding properties *Event_State*, *Reliability* and *Out_Of_Service* as described by the standard.

OVERRIDDEN Always FALSE.

Reliability (Read-Only)

The *Reliability* property for this object type may have any of the following values:

NO_FAULT_DETECTED	No fault was detected.
NO_SENSOR	A lamp failure was detected (for groups and channels if at least one lamp failure in the group/on the channel was detected).
COMMUNICATION_FAILURE	Ballast offline (for groups and channels if at least one ballast is offline in the group/on the channel).
UNRELIABLE_OTHER	Other failure detected (e.g. DALI device type specific error, for groups and channels if at least one ballast in the group/on the channel reports a failure)

Units (Read-Only)

The value of this property is 98 (percent).

Min_Pres_Value (Read-Only)

The value of this property is 0.

Max_Pres_Value (Read-Only)

The value of this property is 100.

Profile_Name (Read-only)

This property has the following values:

- DALI lamps: “*178-http://www.loytec.com/xsd/DALILampActuatorFeedback/v1*”
- DALI groups: “*178-http://www.loytec.com/xsd/DALIGroupActuatorFeedback/v1*”
- DALI channels: “*178-http://www.loytec.com/xsd/DALIChannelActuatorFeedback/v1*”

8.3.3.12 Multi-State Output Object – Scene Control for DALI group or channel

This object is used to control the dim level of DALI groups and DALI channels via scene control. It has states for scene learning, recalling and clearing (see Table 32).

Property Identifier	Property ID	Property Datatype	Conformance Code
Object_Identifier	75	BACnetObjectIdentifier	R
Object_Name	77	CharacterString	R
Object_Type	79	BACnetObjectType	R
Present_Value	85	Unsigned	W
Status_Flags	111	BACnetStatusFlags	R
Event_State	36	BACnetEventState	R
Out_Of_Service	81	BOOLEAN	W
Number_Of_States	74	Unsigned	R
State_Text	110	BACnetARRAY[N]of CharacterString	W
Priority_Array	87	BACnetPriorityArray	R
Relinquish_Default	104	Unsigned	R
Profile_Name	168	CharacterString	R

Table 32: Properties of the Multi-State Output object used for scene control of DALI groups and channels.

Object_Identifier (Read-Only)

The instance number is equal to the instance number of the corresponding Analog Output object (see Section 8.3.3.10).

Object_Name (Read-Only)

This property holds the value of the *Object_Name* of the associated Analog Output object with the addition "*Scene*".

Object_Type (Read-Only)

The value of this property is MULTISTATE_OUTPUT (14).

Present_Value (Read/Write)

The *Present_Value* property can have the following values:

GO TO SCENE X (1-16): Where X is 0-15. Results in recalling the DALI scene X by sending the DALI "GO TO SCENE" command to the DALI group address or DALI broadcast address, which is associated with this Multi-State Output object.

STORE SCENE X (17-32): Where X is 0-15. Results in the current values being stored as scene X by sending the DALI "STORE THE DTR AS SCENE" command to the DALI group address or DALI broadcast address, which is associated with this Multi-State Output object.

REMOVE SCENE X (33-48): Where X is 0-15. Results in scene X being erased by sending the DALI "REMOVE FROM SCENE" command to the DALI group address or DALI broadcast address, which is associated with this Multi-State Output object.

Status_Flags (Read-Only)

Of the four flags the flags IN_ALARM, FAULT, and OUT_OF_SERVICE are linked to the state of the corresponding properties *Event_State*, *Reliability* and *Out_Of_Service* as described by the standard.

OVERRIDDEN	Logical TRUE (1) if the associated DALI channel is controlled via the button interface (see Section 4.5.2), FALSE otherwise.

Reliability (Read-Only)

The *Reliability* property of this object type may have any of the following values:

NO_FAULT_DETECTED	No fault was detected.

Number_Of_States (Read-Only)

This property has the value 48.

State_Text (Read/Write)

Per default this property contains the strings "*GO TO SCENE X*", "*STORE SCENE X*", and "*REMOVE SCENE X*" respectively.

Profile_Name (Read-only)

This property has the following values:

- DALI groups: "*178-http://www.loytec.com/xsd/DALIGroupSceneCtrl/v1*"
- DALI channels: "*178-http://www.loytec.com/xsd/DALIChannelSceneCtrl/v1*"

8.3.3.13 Accumulator Object – Energy usage of DALI group or channel

This object is used to represent the accumulated energy usage of DALI groups and channels (see Table 33).

Property Identifier	Property ID	Property Datatype	Conformance Code
Object_Identifier	75	BACnetObjectIdentifier	R
Object_Name	77	CharacterString	R
Object_Type	79	BACnetObjectType	R
Present_Value	85	UNSIGNED	R
Device_Type	31	CharacterString	R
Status_Flags	111	BACnetStatusFlags	R
Event_State	36	BACnetEventState	R
Reliability	103	BACnetReliability	R
Out_Of_Service	81	BOOLEAN	W
Scale	187	BACnetScale	R
Units	117	BACnetEngineeringUnits	R
Max_Pres_Value	65	Unsigned	R
Value_Change_Time	192	BACnetDateTime	R
Value_Before_Change	190	Unsigned	R
Value_Set	191	Unsigned	W
Time_Delay	113	Unsigned	W
Notification_Class	17	Unsigned	W
High_Limit	45	REAL	W
Low_Limit	59	REAL	W
Deadband	25	REAL	W
Limit_Enable	52	BACnetLimitEnable	W
Event_Enable	35	BACnetEventTransitionBits	W
Acked_Transitions	0	BACnetEventTransitionBits	W
Notify_Type	72	BACnetNotifyType	W
Event_Time_Stamps	130	BACnetARRAY[3] of BACnetTimeStamp	R
Profile_Name	168	CharacterString	R

Table 33: Properties of the Accumulator object representing the accumulated energy usage of DALI groups or channels.

Object_Identifier (Read-Only)

The instance number is equal to the instance number of the corresponding Analog Output object (see Section 8.3.3.10).

Object_Name (Read-Only)

This property holds the value of the *Object_Name* of the associated Analog Output object with the addition "*Energy*".

Object_Type (Read-Only)

The value of this property is ACCUMULATOR (23).

Present_Value (Read-Only)

The value of this property, of type UNSIGNED, indicates the accumulated energy usage in Wh. For objects associated to DALI groups/channels this property reports the sum of the accumulated energy usage of all ballasts in the DALI group/channel.

For calculating the energy usage the nominal power of the affected DALI lamps must be known (see Property *Nominal_Power*, Section 8.3.3.10). The value of this property is just a calculated value and thus is not suitable for billing energy usage.

Device_Type (Read-Only)

This property contains the value "*Accumulated Energy Usage*".

Status_Flags (Read-Only)

Of the four flags the flags IN_ALARM, FAULT, and OUT_OF_SERVICE are linked to the state of the corresponding properties *Event_State*, *Reliability* and *Out_Of_Service* as described by the standard.

OVERRIDDEN Always FALSE.

Reliability (Read-Only)

The *Reliability* property for this object type may have any of the following values:

NO_FAULT_DETECTED No fault was detected.

UNRELIABLE_OTHER Other failure detected.

Scale (Read-Only)

The value of this property is INTEGER 1.

Units (Read-Only)

The value of this property is 18 (watt-hours).

Max_Pres_Value (Read-Only)

The value of this property is 4.294.967.295 (maximum unsigned 32-bit value).

Value_Set (Read/Write)

This property behaves as defined by the BACnet standard. Resetting the accumulator value is possible via the *Value_Set* property (*Value_Set* is writeable, *Value_Before_Change* is not writeable).

Profile_Name (Read-only)

The value of this property is "*178-http://www.loytec.com/xsd/LightEnergyAccumulator/v1*".

Intrinsic reporting

The object supports all properties required for intrinsic reporting. Per default no alarms are enabled.

8.3.3.14 Analog Input Object – Health State of DALI group or channel

This object is used to represent the health state of DALI groups (see Table 34) or DALI channels (see Table 35).

Property Identifier	Property ID	Property Datatype	Conformance Code
Object_Identifier	75	BACnetObjectIdentifier	R
Object_Name	77	CharacterString	R
Object_Type	79	BACnetObjectType	R
Present_Value	85	REAL	R
Status_Flags	111	BACnetStatusFlags	R
Event_State	36	BACnetEventState	R
Reliability	103	BACnetReliability	R
Out_Of_Service	81	BOOLEAN	W
Units	117	BACnetEngineeringUnits	R
Min_Pres_Value	69	REAL	R
Max_Pres_Value	65	REAL	R
COV_Increment	22	REAL	W
Time_Delay	113	Unsigned	W
Notification_Class	17	Unsigned	W
High_Limit	45	REAL	W
Low_Limit	59	REAL	W
Deadband	25	REAL	W
Limit_Enable	52	BACnetLimitEnable	W
Event_Enable	35	BACnetEventTransitionBits	W
Acked_Transitions	0	BACnetEventTransitionBits	W
Notify_Type	72	BACnetNotifyType	W
Event_Time_Stamps	130	BACnetARRAY[3] of BACnetTimeStamp	R
Profile_Name	168	CharacterString	R

Table 34: Properties of the Analog Input object representing the health state of a DALI group.

Property Identifier	Property ID	Property Datatype	Conformance Code
Object_Identifier	75	BACnetObjectIdentifier	R
Object_Name	77	CharacterString	R
Object_Type	79	BACnetObjectType	R
Present_Value	85	REAL	R
Status_Flags	111	BACnetStatusFlags	R
Event_State	36	BACnetEventState	R
Reliability	103	BACnetReliability	R
Out_Of_Service	81	BOOLEAN	W
Units	117	BACnetEngineeringUnits	R
Min_Pres_Value	69	REAL	R
Max_Pres_Value	65	REAL	R
COV_Increment	22	REAL	W
Time_Delay	113	Unsigned	W
Notification_Class	17	Unsigned	W
High_Limit	45	REAL	W
Low_Limit	59	REAL	W
Deadband	25	REAL	W
Limit_Enable	52	BACnetLimitEnable	W
Event_Enable	35	BACnetEventTransitionBits	W
Acked_Transitions	0	BACnetEventTransitionBits	W
Notify_Type	72	BACnetNotifyType	W
Event_Time_Stamps	130	BACnetARRAY[3] of BACnetTimeStamp	R
Profile_Name	168	CharacterString	R
Device_Failure	529	BIT STRING	RV
Ballast_Failure	530	BIT STRING	RV
Lamp_Failure	531	BIT STRING	RV
Battery_Failure	532	BIT STRING	RV
Function_Test_Failure	533	BIT STRING	RV
Duration_Test_Failure	534	BIT STRING	RV

Table 35: Properties of the Analog Input object representing the health state of a DALI channel.

Object_Identifier (Read-Only)

The default instance number is ABCC, where “A” is 3 for the objects associated with DALI groups, and 4 with DALI channels, “B” is the DALI interface number (0-3) in decimal, and “CC” is the DALI group address (00-15) in decimal (00 for DALI channels). Examples: Group 5 on channel 1 has the instance number 3005, and channel 4 has the instance number 4300.

Object_Name (Read-Only)

This property holds the value of the *Object_Name* of the associated Analog Output object with the addition “*Status*”.

Object_Type (Read-Only)

The value of this property is ANALOG_INPUT (0).

Present_Value (Read-Only)

This property reports the percentage of failed ballasts on the corresponding group/channel.

A ballast failure is reported if either

- The bus power for the related DALI channel fails,
- the DALI ballast is not reachable via the DALI channel, or
- the DALI ballast reports a failure (e.g. lamp failure) in its DALI status register.

Status_Flags (Read-Only)

Of the four flags the flags IN_ALARM, FAULT, and OUT_OF_SERVICE are linked to the state of the corresponding properties *Event_State*, *Reliability* and *Out_Of_Service* as described by the standard.

OVERRIDDEN Always FALSE.

Reliability (Read-Only)

The *Reliability* property for this object type may have any of the following values:

NO_FAULT_DETECTED No fault was detected.

Units (Read-Only)

The value of this property is 98 (percent).

Min_Pres_Value (Read-Only)

The value of this property is 0.

Max_Pres_Value (Read-Only)

The value of this property is 100.

Profile_Name (Read-only)

This property has the following values:

- DALI groups: "*178-http://www.loytec.com/xsd/DALIGroupActuatorStatus/v1*"
- DALI channels: "*178-http://www.loytec.com/xsd/DALIChannelActuatorStatus/v1*"

Device_Failure (Read-Only)

The value of this proprietary property (property ID 529) is a 64-bit wide BIT STRING. It contains one bit for each ballast on the channel. If the device fails (offline or failure reported via DALI status bits) the bit is set, otherwise it is cleared.

Only present if the object is associated with a DALI channel.

Ballast_Failure (Read-Only)

The value of this proprietary property (property ID 530) is a 64-bit wide BIT STRING. It contains one bit for each ballast on the channel. If the device reports a ballast failure the bit is set, otherwise it is cleared.

Only present if the object is associated with a DALI channel.

Lamp_Failure (Read-Only)

The value of this proprietary property (property ID 531) is a 64-bit wide BIT STRING. It contains one bit for each ballast on the channel. If the device reports a lamp failure the bit is set, otherwise it is cleared.

Only present if the object is associated with a DALI channel.

Battery_Failure (Read-Only)

The value of this proprietary property (property ID 532) is a 64-bit wide BIT STRING. It contains one bit for each ballast on the channel. If the device reports a battery failure (self-contained emergency lights only) the bit is set, otherwise it is cleared.

Only present if the object is associated with a DALI channel.

Function_Test_Failure (Read-Only)

The value of this proprietary property (property ID 533) is a 64-bit wide BIT STRING. It contains one bit for each ballast on the channel. If the device reports a function test failure (self-contained emergency lights only) the bit is set, otherwise it is cleared.

Only present if the object is associated with a DALI channel.

Duration_Test_Failure (Read-Only)

The value of this proprietary property (property ID 534) is a 64-bit wide BIT STRING. It contains one bit for each ballast on the channel. If the device reports a duration test failure (self-contained emergency lights only) the bit is set, otherwise it is cleared.

Only present if the object is associated with a DALI channel.

Intrinsic reporting

The object supports all properties required for intrinsic reporting. Per default no alarms are enabled.

8.3.4 Sensor Objects

The L-DALI provides objects for 16 DALI sensors per channel. Each sensor is represented by

- an Analog Input object providing the lux level measured by the sensor and
- an Binary Input object providing the occupancy state determined by the sensor.

8.3.4.1 Sensor Calibration

To provide a reliable measurement the light sensor needs to be calibrated. The calibration is best performed using the Web Interface (refer to Section 5.3.4.6).

8.3.4.2 Analog Input Object – DALI lux level sensor

This object is used to represent the light sensor part of a DALI Sensor (see Table 36). There is a maximum of 16 such objects per channel.

Property Identifier	Property ID	Property Datatype	Conformance Code
Object_Identifier	75	BACnetObjectIdentifier	R
Object_Name	77	CharacterString	R
Object_Type	79	BACnetObjectType	R
Present_Value	85	REAL	R
Device_Type	31	CharacterString	R
Status_Flags	111	BACnetStatusFlags	R
Event_State	36	BACnetEventState	R
Reliability	103	BACnetReliability	R
Out_Of_Service	81	BOOLEAN	W
Units	117	BACnetEngineeringUnits	R
Min_Pres_Value	69	REAL	R
Max_Pres_Value	65	REAL	R
COV_Increment	22	REAL	W
Time_Delay	113	Unsigned	W
Notification_Class	17	Unsigned	W
High_Limit	45	REAL	W
Low_Limit	59	REAL	W
Deadband	25	REAL	W
Limit_Enable	52	BACnetLimitEnable	W
Event_Enable	35	BACnetEventTransitionBits	W
Acked_Transitions	0	BACnetEventTransitionBits	W
Notify_Type	72	BACnetNotifyType	W
Event_Time_Stamps	130	BACnetARRAY[3] of BACnetTimeStamp	R
Profile_Name	168	CharacterString	R
Lux_2	557	REAL	R

Table 36: Properties of the Analog Input object representing a DALI lux level sensor.

Object_Identifier (Read-Only)

The default instance number is 5BCC, where “B” is the DALI interface number (0-3) in decimal, and “CC” is the DALI device number (00-15). Example: DALI sensor 4 on channel 2 has the instance number 5104.

Object_Name (Read/Write)

This property holds the name of the DALI sensor. It can be changed via the Web-UI.

Object_Type (Read-Only)

The value of this property is ANALOG_INPUT (0).

Present_Value (Read-Only)

This property reports the current light level reported by the sensor corrected by the calibration function configured via the gain table (see Section 8.3.4.1).

Device_Type (Read-Only)

This property contains the type of DALI sensor (e.g. “Tridonic Sensor”).

Status_Flags (Read-Only)

Of the four flags the flags IN_ALARM, FAULT, and OUT_OF_SERVICE are linked to the state of the corresponding properties *Event_State*, *Reliability* and *Out_Of_Service* as described by the standard.

OVERRIDDEN	Always FALSE.

Reliability (Read-Only)

The *Reliability* property for this object type may have any of the following values:

NO_FAULT_DETECTED	No fault was detected.
NO_SENSOR	No sensor assigned or some failure reported by the DALI sensor (no light value reported).
COMMUNICATION_FAILURE	Sensor offline.
UNRELIABLE_OTHER	Other failure detected.

Units (Read-Only)

The value of this property is 37 (lux).

Min_Pres_Value (Read-Only)

The value of this property is 0.

Max_Pres_Value (Read-Only)

The value of this property is “Infinity”.

Profile_Name (Read-only)

The value of this property is “*178-http://www.loytec.com/xsd/DALILightSensor/v1*”.

Lux_2 (Read-only)

If the assigned sensor provides multiple lux level sensor values this property (property ID 557) reports the current second light level value reported by the sensor. No gain table correction is applied.

Intrinsic reporting

The object supports all properties required for intrinsic reporting. Per default only fault alarms are enabled.

8.3.4.3 Binary Input Object – DALI occupancy sensor

This object is used to represent the occupancy sensor part of a DALI Sensor (see Table 37). There is a maximum of 16 such objects per channel.

Property Identifier	Property ID	Property Datatype	Conformance Code
Object_Identifier	75	BACnetObjectIdentifier	R
Object_Name	77	CharacterString	R
Object_Type	79	BACnetObjectType	R
Present_Value	85	BACnetBinaryPV	R
Status_Flags	111	BACnetStatusFlags	R
Event_State	36	BACnetEventState	R
Reliability	103	BACnetReliability	R
Out_Of_Service	81	BOOLEAN	W
Polarity	84	BACnetPolarity	W
Inactive_Text	46	CharacterString	W
Active_Text	4	CharacterString	W
Profile_Name	168	CharacterString	R
Debounce	535	Unsigned	W
Unoccupied_Delay	536	Unsigned	W
Heartbeat	558	Unsigned	W

Table 37: Properties of the Binary Input object representing a DALI occupancy sensor.

Object_Identifier (Read-Only)

The instance number is equal to the instance number of the corresponding Analog Input object (see Section 8.3.4.2).

Object_Name (Read-Only)

This property holds the value of the *Object_Name* of the associated Analog Input object with the addition "*Occupancy*".

Object_Type (Read-Only)

The value of this property is BINARY_INPUT (3).

Present_Value (Read-Only)

This property reports the occupancy status. If the *Polarity* property is *NORMAL* the occupied state corresponds to *ACTIVE*, while the unoccupied state corresponds to *INACTIVE*.

Status_Flags (Read-Only)

Of the four flags the flags IN_ALARM, FAULT, and OUT_OF_SERVICE are linked to the state of the corresponding properties *Event_State*, *Reliability* and *Out_Of_Service* as described by the standard.

OVERRIDDEN Always FALSE.

Reliability (Read-Only)

The *Reliability* property for this object type may have any of the following values:

NO_FAULT_DETECTED No fault was detected.

NO_SENSOR	No sensor assigned or some failure reported by the DALI sensor (no occupancy value reported).
COMMUNICATION_FAILURE	Sensor offline.
UNRELIABLE_OTHER	Other failure detected.

Profile_Name (Read-only)

The value of this property is "*178-http://www.loytec.com/xsd/DALIOccupancy Sensor/v1*".

Debounce (Read/Write)

The value of this proprietary property (property ID 535) specifies the debouncing time in seconds (see Section 8.2.7.1).

Unoccupied_Delay (Read/Write)

The value of this proprietary property (property ID 536) determines the delay in seconds after which the state changes to unoccupied. The timer is armed each time the value reported by the sensor changes from occupied to unoccupied. A value of 0 disables the unoccupied delay.

Heartbeat (Read/Write)

The value of this proprietary property (property ID 558) determines the maximum period of time that expires before the object automatically updates its *Present_Value*. This allows using the *Present_Value* as occupancy event input to a constant light controller object.

8.3.5 Constant Light Controller

The L-DALI provides 16 constant light controller instances with built in occupancy controller functionality for each DALI channel. Every instance is represented by a BACnet Loop object.

8.3.5.1 Setpoint

The desired setpoint for the illumination level is configured via the property *Setpoint*.

8.3.5.2 Operating Modes

The operating mode of the constant light controller is set via the proprietary property *Mode* (property ID 539). Table 38 shows the different operating modes and under which conditions they are used:

Operating Mode	Description
REGULATOR (1)	A light sensor that measures the indoor illumination is installed. An occupancy sensor is installed.
CONTROL (2)	A light sensor that measures the outdoor illumination is installed. An occupancy sensor is installed.
UPDATER (3)	A DALI sensor with constant light controller functionality shall be mapped to the BACnet network. Only the setpoint shall be adjustable via the BACnet interface.
PRESENCE (4)	Based on occupancy the lights shall be switched on and off.
REGULATOR_NO_OCC (5)	A light sensor that measures the indoor illumination is installed. No occupancy sensor is installed.
CONTROL_NO_OCC (6)	A light sensor that measures the outdoor illumination is installed. No occupancy sensor is installed.
MANUAL_ON_AUTO_OFF (7)	Lights are switched on manually, but shall be switched off based on occupancy.
REGULATOR_AUTO (8)	Similar to REGULATOR, but a manual override is automatically relinquished when the room becomes unoccupied.
CONTROL_AUTO (9)	Similar to CONTROL, but a manual override is automatically relinquished when the room becomes unoccupied.
PRESENCE_AUTO (10)	Similar to PRESENCE, but a manual override is automatically relinquished when the room becomes unoccupied.
AMBIENT (11)	Effect or ambient lighting is switched on based on the measured illumination level (switching on ambient lights does not influence measured illumination level).
REGULATOR_BEDROOM (12)	Similar to REGULATOR_AUTO, but a manual override is not automatically relinquished when the lights are overridden to off.
CONTROL_BEDROOM (13)	Similar to CONTROL_AUTO, but a manual override is not automatically relinquished when the lights are overridden to off.

Table 38: Constant Light Controller Operation Modes

Regulator Mode

The REGULATOR mode has to be selected if the installed light sensor measures the indoor brightness. The constant light controller receives the current indoor lux level via the property *Controlled_Variable_Value* and adjusts the lamp level (from 0% to 100%) by means of a fuzzy control algorithm.

When the light is switched on the control algorithm selects a lamp value which results in a lux level close to the desired setpoint and then adjusts the lamp value in increments once every second until the configured setpoint is reached. The size of increments is adjusted dynamically but will never exceed the percentage value defined in the proprietary property *Step_Value* (property ID 544).

The regulator mode is available in different flavors:

- **With our without occupancy sensor**: If an occupancy sensor is installed and the room becomes unoccupied, the current light level is saved and the light is dimmed to unoccupied level (see Section 8.3.5.7). If the room becomes occupied before the light has reached the unoccupied level, the saved light level is restored.

- **With automatic relinquish**: A manual override will be automatically relinquished when the room becomes unoccupied. In a special bedroom mode the automatic relinquish will only take place if the lights are on. If they were switched off manually, the controller does not return to automatic mode when the area turns unoccupied.

Control Mode

The CONTROL mode has to be selected if the installed light sensor measures the outdoor illumination. The constant light controller receives the current outdoor lux level via the property *Controlled_Variable_Value* and sets the lamp level (from 0% to 100%) accordingly (see Figure 220).

The proprietary property *Step_Value* (property ID 544) ensures that the light level does not change abruptly. Per second the light level is not changed more than the percentage value defined by this configuration property. When the light is switched on or off, *Step_Value* is not taken into account.

To ensure that the setpoint is reached under all weather conditions, the control curve shown Figure 220 has to be adjusted by calibrating the constant light controller.

The control mode is available in different flavors:

- **With our without occupancy sensor**: If an occupancy sensor is installed and the room becomes unoccupied, the current light level is saved and the light is dimmed to unoccupied level (see Section 8.3.5.7). If the room becomes occupied before the light has reached the unoccupied level, the saved light level is restored.

- **With automatic relinquish**: A manual override will be automatically relinquished when the room becomes unoccupied. In a special bedroom mode the automatic relinquish will only take place if the lights are on. If they were switched off manually, the controller does not return to automatic mode when the area turns unoccupied.

Presence Mode

The PRESENCE mode has to be if the constant light controller shall operate as occupancy controller. In this mode the light is switched to the value specified in the proprietary property *Occupied_Level* (property ID 542) if the room is occupied and to the unoccupied level (see Section 8.3.5.7) if the room is unoccupied.

The light is switched to the value *Occupied_Level* only if the lux level is below *Setpoint* and the room is occupied.

The presence mode is available in different flavors:

- **With or without automatic relinquish**: If automatic relinquish is active a manual override will be automatically relinquished when the room becomes unoccupied.

Manual-On/Auto-Off Mode

The MANUAL_ON_AUTO_OFF mode is a variation of the PRESENCE mode. However, in this mode the constant light controller does not switch on the lights, when the room becomes occupied. Rather, it switches the lights to the unoccupied level (see Section 8.3.5.7) when the room becomes unoccupied, using the propertiy *Hold_Time* similar to PRESENCE mode.

Lights can be switched on (or off) via some other DALI master (e.g. a DALI push-button).

Ambient Mode

The AMBIENT mode is used if effect lighting (e.g. lights in a shop window, facade lighting, floor lights etc.) has to be switched on or off depending on a measured light level. Lights are switched to the value specified in the proprietary property *Occupied_Level* (property ID 542) when the lux level is below *Setpoint* and to the unoccupied level (see Section 8.3.5.7) when lux level is higher than *Setpoint.*

Updater Mode

The UPDATER mode has to be selected when a DALI light/occupancy sensor with constant light controller functionality is installed on the DALI network. In this mode no light level computation is performed. The controller simply forwards the parameters to the DALI constant light controller and retrieves lamp setting from the DALI light sensor.

In general, LOYTEC does not recommend using UPDATER mode. Wherever possible use REGULATOR or CONTROL mode instead and use DALI sensors only as lux level and occupancy information source via the corresponding sensor objects. These modes allow improved influence of the constant light controller's parameters via the fieldbus side.

8.3.5.3 Prioritized Control

Manual override of the constant light controller can be accomplished in two ways:

- Using the priority array of the controlled lamp actuator Analog Output object using a higher priority than the constant light controller.

- When internal CLC bindings are used to control DALI groups an override can also be performed using a DALI push-button device controlling the same groups as the constant light controller (see Section 8.3.5.8).

8.3.5.4 Controlling Multiple Light Bands

The L-DALI constant light controller allows controlling two light bands (groups). The primary light band (Light Band 1) is near the inside of the building, the secondary light band (Light Band 2) is near the window front. Depending on the outdoor light intensity the primary light band has to be brighter than the secondary light band to illuminate the room evenly.

The primary light band is controlled by the *Present_Value* property of the Loop object. Further for both light bands internal bindings to local DALI groups and their corresponding Analog Output objects can be done (see Section 5.3.2).

The maximum difference between the two light bands can be configured via the configuration property *Lamp_2_Offset* as shown in Figure 232. The configuration property *Lamp_2_Limit* defines the light level above which the output values of light band 1 and 2 are identical.

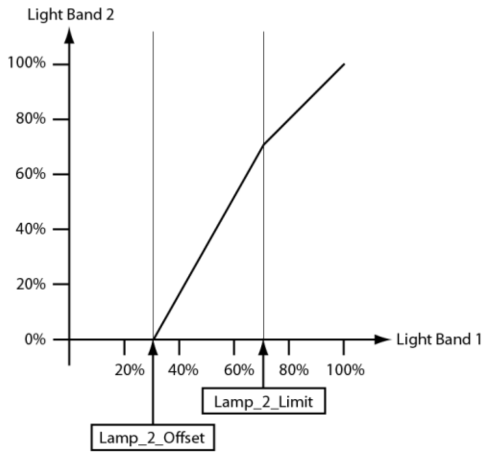


Figure 232: Secondary Light Band

8.3.5.5 Occupancy Detection

The constant light controller receives the occupancy information via the proprietary property *Occupancy_Variable_Value* (property ID 538). How this information is evaluated depends on the hold time configured in the proprietary property *Hold_Time* (propery ID 540).

If a hold time is configured (*Hold_Time* > 0) occupancy evaluation is event based. That is, the constant light controller changes to the occupied state when the value OCCUPIED (1) is received. When the UNOCCUPIED (2) value is received the constant light controller remains in the occupied state until the hold time configured in *Hold_Time* has expired and then changes to the unoccupied state. Further, if no OCCUPIED is received for the time configured in *Hold_Time* the constant light controller changes to the unoccupied state, too.

This behavior typically requires the occupancy sensor to send OCCUPIED with a regular interval (heartbeat). It allows using multiple occupancy sensors with the same constant light controller (fan-in). It is the recommended mode if the occupancy information is received via the local DALI occupancy objects (see Section 8.3.4.3).

If no hold time is configured (*Hold_Time* = 0) occupancy evaluation is state based. That is, the constant light controller changes to the occupied state when the value OCCUPIED is received and to unoccupied state when the value UNOCCUPIED is received.

This behavior is typically required if the occupancy information is only propagated if the state changes (no heartbeat functionality). It does not allow using multiple occupancy (fan-in) sensors without additional logic (e.g. an Occupancy Controller). It is the recommended mode, when using occupancy information from other BACnet devices.

After the light has been switched off any updates on the *Occupancy_Variable_Value* input are ignored for the time specified in *Ignore_Time* (property ID 541).

8.3.5.6 Timing Parameters

To avoid that the constant light controller switches the light on and off repeatedly if the measured lux value is near the setpoint, a hysteresis can be configured. The proprietary property *Off_Hysteresis* (property ID 546) defines the hysteresis for switching off the lamp and the proprietary property *On_Hysteresis* (property ID 545) defines the hysteresis for switching on the lamp. In addition a delay can be configured before the lamp is switched on (*On_Delay*) or off (*Off_Delay*). Figure 233 illustrates these configuration parameters.

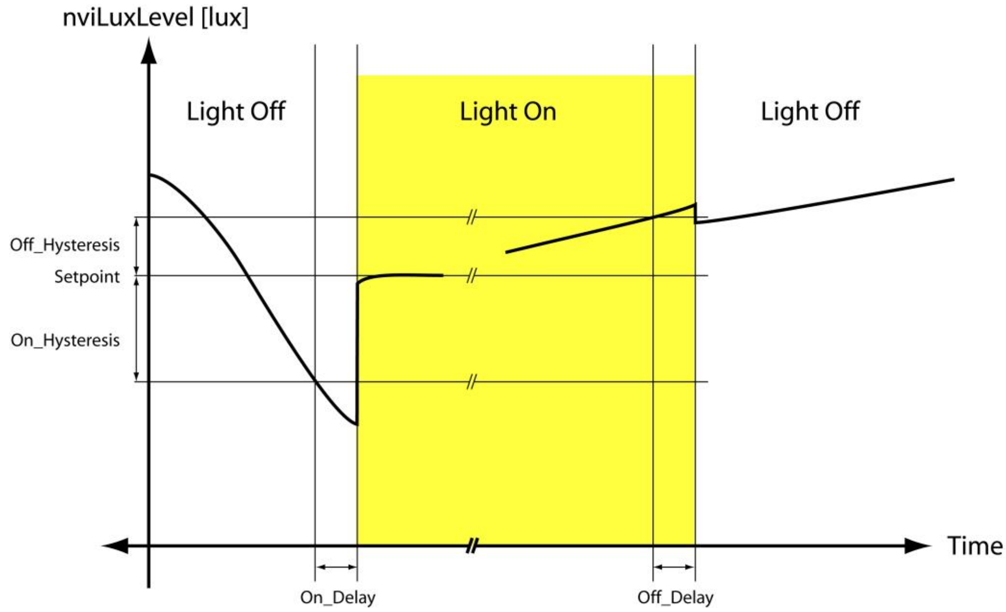


Figure 233: Constant Light Controller Timing.

8.3.5.7 Unoccupied Level

If the constant light controller enters unoccupied state (see Section 8.3.5.5) *Unoccupied_Level* (property ID 543) and *Off_Delay* (property ID 547) determine the behavior as shown in Table 39.

Unoccupied_Level	***Off_Delay***	**Behavior**
0	0	Dim down and switch off lights
> 0	0	Dim down lights to value defined in *Unoccupied_Level*.
0	> 0	Dim down to 0.5% and switch off lights after *Off_Delay*.
> 0	> 0	Dim down lights to value defined in *Unoccupied_Level* and switch off lights after *Off_Delay*.

Table 39: Behavior of constant light controller when entering unoccupied state.

8.3.5.8 Interaction with DALI Buttons

When internal CLC bindings are used to control DALI groups an override can also be performed using a DALI push-button device like the LDALI-BM1 (or some other DALI master) controlling the same groups as the constant light controller.

Whenever the light level of one of the ballasts controlled by the constant light controller is changed by operating a DALI button the constant light controller will detect this and stop controlling the group and enter override mode. Furthermore, the *Present_Value* of the Constant Light Controllers Loop Object will be updated with the new feedback value of the group controlled by the DALI button.

See Section 5.3.4.7 and 7.6 on how to set up DALI button functions.

8.3.5.9 Interaction with other BACnet Objects

The Loop Object allows automatically retrieving its input values from and writing its output values to other BACnet server objects. To reference the property and server object it provides properties of type *BACnetObjectPropertyReference*. Specifically these are:

- *Manipulated_Variable_Reference*: Writing *Present_Value* (Constant Light Controller output, e.g. to Analog Output controlling DALI lights, see Section 8.3.3.10).

- *Controlled_Variable_Reference*: Reading lux level input (e.g. from DALI lux level sensor, see Section 8.3.4.2).

- *Setpoint_Reference*: Reading setpoint value.

- *Occupancy_Variable_Reference* (property ID 537): Reading occupancy input (e.g. from DALI occupancy sensor, see Section 8.3.4.3).

Note, that the referenced server object must be local.

Alternatively internal bindings can be configured as described in Section 5.3.2.

8.3.5.10 Loop Object – Constant Light Controller application

This object is used to represent a constant light controller instance (see Table 40). There are up to 16 instances per DALI channel available.

Property Identifier	Property ID	Property Datatype	Conformance Code
Object_Identifier	75	BACnetObjectIdentifier	R
Object_Name	77	CharacterString	R
Object_Type	79	BACnetObjectType	R
Present_Value	85	REAL	R
Description	28	CharacterString	W
Status_Flags	111	BACnetStatusFlags	R
Event_State	36	BACnetEventState	R
Reliability	103	BACnetReliability	R
Out_Of_Service	81	BOOLEAN	W
Output_Units	82	BACnetEngineeringUnits	R
Manipulated_Variable_Reference	60	BACnetObjectPropertyReference	W
Controlled_Variable_Reference	19	BACnetObjectPropertyReference	W
Controlled_Variable_Value	21	REAL	W
Controlled_Variable_Units	20	BACnetEngineeringUnits	R
Setpoint_Reference	109	BACnetSetpointReference	W
Setpoint	108	REAL	W
Action	2	BACnetAction	R
Priority_For_Writing	88	Unsigned	W
COV_Increment	22	REAL	W
Profile_Name	168	CharacterString	R
Occupancy_Variable_Reference	537	BACnetObjectPropertyReference	W
Occupancy_Variable_Value	538	BOOLEAN	W
Mode	539	Unsigned	W
Hold_Time	540	Unsigned	W
Ignore_Time	541	Unsigned	W
Occupied_Level	542	REAL	W
Unoccupied_Level	543	REAL	W
Step_Value	544	REAL	W
On_Hysteresis	545	REAL	W
Off_Hysteresis	546	REAL	W
Off_Delay	547	Unsigned	W
On_Delay	548	Unsigned	W

Table 40: Properties of the Loop object representing a Constant Light Controller Application.

Object_Identifier (Read-Only)

The default instance number is BCC, where "B" is the DALI interface number (0-3) in decimal, and "CC" is the Constant Light Controller instance on the channel (00-15).

Object_Name (Read-Only)

This property holds the name of the constant light controller.

Object_Type (Read-Only)

The value of this property is LOOP (12).

Present_Value (Read-Only)

This property, of type REAL, indicates the linearized percentage (0..100.00%) of the constant light controller output.

Description (Read/Write)

This property can be configured via BACnet. It serves documentation purposes only.

Reliability (Read-Only)

The *Reliability* property for this object type may have any of the following values:

NO_FAULT_DETECTED	No fault was detected.
OPEN_LOOP	The sensor value does not change when the output is controlled (CONTROL mode only).
COMMUNICATION_FAILURE	Communication with sensor or light output not working.
UNRELIABLE_OTHER	Other failure detected.

Output_Units (Read-Only)

The value of this property is 98 (percent).

Manipulated_Variable_Reference (Read/Write)

Specifies the object and property the value of the *Present_Value* property is written to.

Controlled_Variable_Reference (Read/Write)

Specifies the object and property the light sensor value is read from.

Controlled_Variable_Value (Read/Write)

This property holds the current light sensor value as read from Controlled_Variable_Reference.

Controlled_Variable_Units (Read-Only)

The value of this property is 37 (lux).

Setpoint_Reference (Read/Write)

Specifies the object and property the setpoint for the constant light controller algorithm is read from.

Setpoint (Read/Write)

This property holds the setpoint for the constant light controller algorithm.

Action (Read-Only)

The value of this property is DIRECT.

Profile_Name (Read-only)

The value of this property is "*178-http://www.loytec.com/xsd/ConstantLightController/v1*".

Occupancy_Variable_Reference (Read/Write)

This proprietary property (property ID 537) specifies the object and property the occupancy sensor value is read from.

Occupancy_Variable_Value (Read/Write)

This proprietary property (property ID 538) holds the current occupancy sensor value as read from Occupancy_Variable_Reference.

Mode (Read/Write)

This proprietary property (property ID 539) holds the constant light controller operating mode (see Section 8.3.5.2).

Hold_Time (Read/Write)

This proprietary property (property ID 540) holds the hold time for the occupancy state in seconds. When the OCCUPIED (1) value is received via *Occupancy_Variable_Value* the constant light controller remains in the occupied state until the hold time has expired and then changes to the unoccupied state. If set to 0 the unoccupied state is entered when UNOCCUPIED (2) is received.

Ignore_Time (Read/Write)

This proprietary property (property ID 541) holds the occupancy ignore time after switching off the light in seconds.

Occupied_Level (Read/Write)

This proprietary property (property ID 542) holds the output levels adopted in the presence modes in occupied state.

Unoccupied_Level (Read/Write)

This proprietary property (property ID 543) holds the output levels adopted in the presence modes in unoccupied state.

Step_Value (Read/Write)

This proprietary property (property ID 544) holds the step value for constant light controller algorithm in percent. It defines the maximum step size that the constant light controller will take to approach the target illumination. Per second the light level is not changed more than the percentage value defined by this configuration property.

On_Hysteresis (Read/Write)

This proprietary property (property ID 545) holds the constant light controller algorithm hysteresis for switching lights on (percent).

The hysteresis is used in CONTROL or REGULATOR mode.

Off_Hysteresis (Read/Write)

This proprietary property (property ID 546) holds the constant light controller algorithm hysteresis for switching lights off (percent).

The hysteresis is used in CONTROL or REGULATOR mode.

Off_Delay (Read/Write)

This proprietary property (property ID 547) holds the constant light controller off delay in seconds.

On_Delay (Read/Write)

This proprietary property (property ID 548) holds the constant light controller on delay in seconds.

8.3.6 Button Objects

Currently no button objects are supported on the L-DALI BACnet interface.

8.3.7 Alarming, Scheduling and Trending Objects (AST)

The BACnet interface provides the following AST objects:

- 4 Notification Class objects per channel (Alarming)
- 8 Calendar objects
- 16 Scheduler objects per channel
- 16 Trendlog object per channel

To configure the trended and scheduled data points a standard BACnet Operator Workstation has to be used, to setup the corresponding properties of the Scheduler and Trendlog objects respectively.

8.3.8 Client Mapping CSV File

Client functionality for the BACnet server objects can be defined by so-called *client mappings*. These mappings basically specify whether present value properties shall be written to or polled from the BACnet network, and what the destination address and objects are. These definitions can be downloaded as a CSV file onto the device using FTP.

The CSV file must be named 'bacclnt.csv' and stored in the directory '/var/lib/bacnet' on the device. The file is read when the device boots. If any errors occur they are reported in '/tmp/bacclnt.err'.

The column format is shown in Table 41. Lines beginning with a hash ('#') sign are comment lines. The example values in Table 41 setup a client mapping named "Lamp Room 302", which writes (mapping type 2) the present value of the local object AI,4 to the remote object AO,1 on the device with the instance number 17801.

Column	Field	Example	Description
A	Description	Lamp Room 302	User-defined description of this client mapping. Can be left empty. Don’t use commas or semi-colons in the text!
B	Local Object-Type	AI	The BACnet object type of the local server object (AI, AO, AV, BI, BO, BV, MI, MO, MV, ACCM, LOOP)
C	Local Object Instance Number	4	The object instance number of the above object.
D	Remote Device Instance	17801	The device object instance number of the remote BACnet device
E	Remote Object-Type	AO	The BACnet object type of the remote server object (AI, AO, AV, BI, BO, BV, MI, MO, MV, ACCM, LOOP)
F	Remote Object Instance Number	1	The object instance number of the above object.
G	Map Type	2	Defines the type of the mapping: 0=Poll, 1=COV, 2=Write, 3=Value
H	Interval	60	Defines the poll interval in seconds for poll/value mappings and the COV lifetime in seconds for COV mappings. Note: In previous versions this column was also used to specify the write priority for write mappings. This usage of column H is deprecated and column I should be used to specify priority.
I	Priority	8	For write and value mappings this defines the write priority (1..16). Omit this field or set it to ‘-1’ to write w/o priority.
J	Local Property ID	45	Specifies the property ID of the local object, which is mapped to the remote object. If omitted, the Present_Value of the local object is mapped.
K	Remote Property ID	45	Specifies the property, which is written/read on the remote object. If omitted, the remote property ID is the Present_Value.

Table 41: CSV Columns of the BACnet Client Mappings File.

8.3.9 EDE Export of BACnet Objects

The BACnet server object configuration of the device is accessible as a set of CSV files following the EDE format convention. They can be downloaded via FTP from the directory ‘/data/ede’ on the device. The files are

- ldali.csv: This is the main EDE sheet with the list of BACnet objects.
- ldali-states.csv: This is the state text sheet. For each state text reference in the main sheet, a line contains the state texts for this multi-state object.
- ldali-types.csv: This is the object types text sheet. The file contains a line for each object type number. Note, that lines for standard object types can be omitted.
- ldali-units.csv: This is the unit text sheet. The file contains a line for each engineering unit enumerator value. Note that lines for standard units can be omitted.

8.4 DALI Interface

On it’s up to four DALI interfaces the L-DALI acts as a DALI master controller. DALI stands for “Digitally Addressable Lighting Interface” and is the name commonly used for the communication protocol defined in the international standard IEC 62386[2]. For further information regarding DALI please refer to http://www.dali-ag.org/.

8.4.1 DALI Wiring

DALI wiring is typically run together with the mains wiring using normal mains rated wire (2 wires). Table 42 shows the recommended conductor size depending on the length of the DALI wires. A total length of 300 m must not be exceeded.

DALI cable length	Recommended min. conductor size
< 100 m	0.5 mm^2
100-150 m	0.75 mm^2
150-300 m	1.5 mm^2

Table 42: Recommended minimum conductor size for DALI wiring

Though the signal is only 16 V (typical), DALI is not SELV rated and should therefore treated as mains voltage wiring.

DALI connections are not polarity sensitive[3].

8.4.2 DALI Channel Limitations

The number of devices per channel is limited by the following aspects:

Limits of L-DALI:

The L-DALI supports a maximum of **64 ballasts**, **16 sensors** and **64 buttons** per DALI channel (which can be assigned).

Addresses:

A maximum of **64 addresses** are available per DALI channel. Each device (ballast, sensor, buttons) uses one DALI address with the following exceptions:

- The **L-DALI** does not use a DALI address.
- **LOYTEC sensor and button** devices use their own address range (**64 addresses each**). Therefore, their number does not count into the number of 64 addresses mentioned above, allowing up to 64 ballasts and up to 16 LOYTEC sensors and up to 64 LOYTEC button devices per channel.

Power consumption:

The power drawn by the devices via the DALI line must be covered by the DALI bus power supply. Table 43 shows the typical power drawn via the DALI line depending on device types. For DALI devices not listed in the table see the corresponding datasheet or contact the device vendor on the power drawn via the DALI line.

---

[2] Previous versions of the DALI standard were defined in IEC 60929 Annex E.

[3] Except for at the bus power supply, in case more than one bus power supply is connected to the channel.

Device type	Power usage
Mains supplied devices (ballasts)	2 mA
LOYTEC LDALI-MS1 multi-sensor	3.5 mA
LOYTEC LDALI-BM1 button coupler	4 mA
LOYTEC LDALI-RM1 relay module	2,1 mA

Table 43: DALI bus power usage for different device types

LOYTEC DALI bus power supplies supply 116 mA or 232 mA depending on the model. Two 116 mA bus power supplies can be connected to the DALI channel in parallel resulting in a combined DALI bus power supply of 232 mA. The DALI bus power **supply must not exceed 250 mA**.

8.4.3 Multi-Master Operation

The L-DALI is capable of multi-master operation. That is, it can be installed in parallel to one or more other DALI master controllers on the same DALI network. However, all other DALI master controllers must be multi-master capable in order to render a working DALI system. Other DALI masters may be DALI multi-sensors with built in constant light or occupancy controller functionality, DALI switches, buttons, and touch panels, as well as other DALI controllers.

8.4.4 DALI Device Types

The L-DALI distinguishes between the following DALI device types:

DALI Ballasts

DALI ballasts are specified in IEC 60929 and the newer IEC 62386. The L-DALI supports IEC 62386 parts 101 and 102 and the DALI device types specified in parts 201 to 209 (Edition 1 of all parts). See [9] for DALI ballasts tested with the L-DALI.

In addition the L-DALI support some vendor specific features (e.g. reading serial number, determining nominal power) for OSRAM and Tridonic ballasts.

DALI Sensors

Currently the DALI standard does not specify DALI sensors. Thus, any “DALI sensors” available use proprietary extensions of the DALI protocol. Nevertheless, the L-DALI supports the following DALI multi-sensors[4]:

- LOYTEC LDALI-MS1 (see Section 8.4.4.1)
- Tridonic MSensor 02 5DPI 41 (see Section 8.4.4.3)
- Schneider Electric LA-11

[4] *LOYTEC electronics GmbH assumes no responsibility for any errors contained in this list. LOYTEC makes no representation and offers no warranty of any kind regarding any of the third-party components mentioned in this list. These components are suggested only as examples of usable devices. The use of these components or other alternatives is at the customer’s sole discretion.*

- Helvar Digidim 311/312/313/314/317
- OSRAM DALI Professional Sensorcoupler series
- OSRAM DALI HIGHBAY Adapter
- OSRAM LS/PD LI
- Philips OccuSwitch DALI LRM2090/20 and LRM2090/30 (occupancy only, see Section 8.4.4.4)
- ThebenHTS PlanoSpot 360 DALI (see Section 8.4.4.6).

Typically these devices offer occupancy and light level sensor functionality. In addition they usually also offer occupancy and constant light controller functionality. The L-DALI allows utilizing the sensor functionality. If it is intended to also use the controller functionality the installation software of the devices vendor must be used to install and parameterize the device as this is not supported by the L-DALI.

In addition the following special sensor is supported:

- Lunatone DALI Temp-Sensor (see Section 8.4.4.7).

If you need support for a DALI sensor not listed above please contact LOYTEC.

DALI Buttons

As for sensors the “DALI switches”, “DALI buttons”, “DALI touch panels”, or remote controls connected to DALI sensors use proprietary extensions of the DALI protocol.

DALI buttons can be used together with the L-DALI in two different way:

- Manual control of DALI groups (dimming lights, scene recall). See Section 5.3.4.7 and 7.6 on how to configure which button input performs witch function.
- Mapping DALI button inputs to network variables on the LONMARK interface of the L-DALI (LDALI-10x only, see Section 8.2.11).

As of now the L-DALI supports the following DALI buttons[4]:

- LOYTEC LDALI-BM1 (see Section 8.4.4.2).
- LOYTEC LDALI-MS1 IRT (infrared receiver, see Section 8.4.4.1)
- Philips IRT 8080 IR-remote control with Philips OccuSwitch DALI LRM2090 (mapping to LONMARK interface only).

Other DALI devices

Different other proprietary DALI devices are currently not supported by the L-DALI. That is, they cannot be commissioned or parameterized using the L-DALI and they are not mapped to LONMARK or BACnet objects like DALI ballasts or supported DALI sensors and DALI buttons.

However, such devices may be operated in a network controlled by the L-DALI. In this case they must be commissioned and parameterized using the installation software of the device’s vendor.

8.4.4.1 LOYTEC LDALI-MS1 Multi-Sensor

The LDALI-MS1 performs motion detection with a PIR sensor and allows to measure lux level in the range of 0 – 2500 lux. It is designed to be mounted in rooms of up to 5 m. For corridors and similar applications the sensor head can be tilted to detect movement in a distance of up to 12 m.

Figure 234: LDALI-MS1 Multi-Sensor with integrated IR receiver.

Note: *Currently the dial on the back of the sensor has no function. Please leave at default position 0!*

In addition the LDALI-MS1 is equipped with an IR receiver to be used with infrared remote controls.

Note: *When scanning the DALI channel, each LDALI-MS1 device will appear twice in the scan results. In the sensor section as **LDALI-MS1** and in the button section as **LDALI-MS1 IRT**. The latter represents the infrared receiver and can be ignored if not infrared remote control is used.*

The LDALI-MS1 infrared receiver is compatible with the Apple Remote. That is, the Apple remote control can be used to control lights in the room. See Figure 235 for how the RC buttons are mapped to the button inputs of the LDALI-MS1 IRT. Optionally pairing is supported.

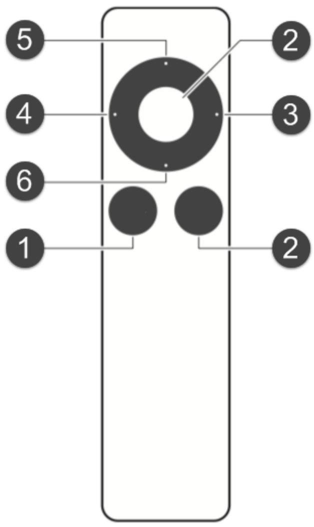


Figure 235: Mapping of Apple Remote buttons.

The LDALI-MS1 IRT supports remote controls with up to 11 buttons using the NEC IR protocol. The vendor code must be 0x43f and the command page 0xe. For unpaired devices the device ID is 0. The command code is equal to the button input index (1-11). Please contact LOYTEC if requiring a custom IR remote control unit (larger quantities).

8.4.4.2 LOYTEC LDALI-BM1 push button coupler

The LDALI-BM1 push button coupler integrates customary light pushbuttons and switches into a DALI channel. It provides 4 free programmable digital inputs. When the push button is pressed or a switch changes its state, DALI commands are sent to a DALI group within the DALI channel. See Section 5.3.4.7 and 7.6 on how to configure which button input performs witch function.

Figure 236: LDALI-BM1 push button-coupler

8.4.4.3 Tridonic MSensor 02 5DPI 41

Important: ***Only Tridonic MSensor 02 5DPI 41 is supported!***

The Tridonic MSensor 02 5DPI 41 comes with a built in constant light controller application, which cannot be disabled. To use it as DALI light level and occupancy sensor only all sensors must be configured to control group 14 by setting the dial on back of device to position “F” (see Figure 237). On this channel group 14 and 15 must not be used for any other purposes anymore (do not configure ballasts to be group member in group 14 or 15).

Figure 237: Set MSensor 02 5DPI 41 group dial to position ‘F’.

Important:	***In any case do not set the dial to position "0" as this will make the sensor send broadcasts!***

The dial position is shown in the Web Interface in the DALI Installation tab (see Figure 238).

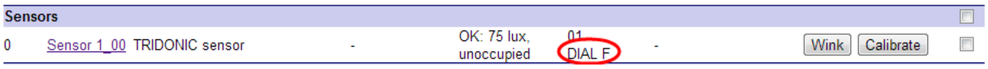


Figure 238: The MSensor 02 5DPI 41 dial position is shown in the Web Interface in the DALI Installation tab.

8.4.4.4 Philips OccuSwitch DALI LRM2090 with IRT 8080 IR-remote control

The Philips OccuSwitch DALI LRM2090 is represented as three devices on the DALI channel (see Figure 239):

- **Philips OccuSwitch lights**: This part represents the ballast functionality of the OccuSwitch. Similar to a ballast the OccuSwitch reports the current dim level of the ballast it controls. By dimming this device the lighting control application of the OccuSwitch can be temporarily overridden. Further it provides the information on the ballast status (lamp failure etc.) of the ballasts controlled by the OccuSwitch. Only this part of the OccuSwitch can be assigned to groups. If the ballast part is assigned to a group this automatically affects all other parts as well.

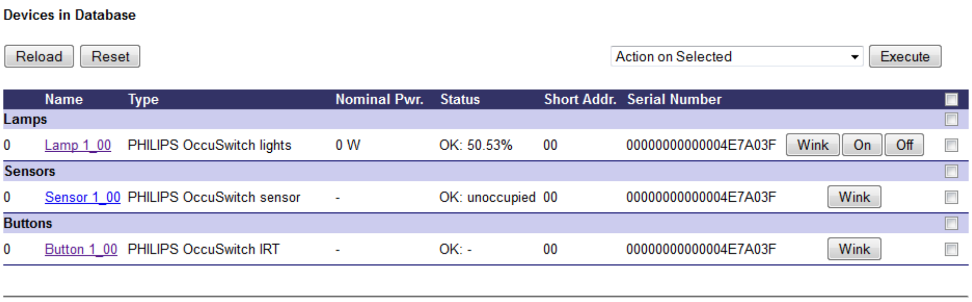


Figure 239: Philips OccuSwitch DALI LRM2090/20 is represented as three DALI devices.

- **Philips OccuSwitch sensor**: This part represents the occupancy sensor functionality of the OccuSwitch. The OccuSwitch does not provide any light level information (lux). Note, that if multiple OccuSwitch devices are assigned to a group only one of these sensors needs to be assigned to a LONMARK or BACnet Occupancy Sensor object, as all sensors in a group will share their occupancy status (if one detects occupancy all will report occupancy).
- **Philips OccuSwitch IRT**: This part represents the IRT 8080 IR-remote control functionality as described in Section 8.2.11.2. It allows controlling sunblinds, fan speed, and HVAC setpoint in a room. Currently this functionality is only supported in the LDALI-3E10X models. If no IRT 8080 is available, these parts need not be assigned.

All three instances will have an identical short address and serial number.

If the Philips OccuSwitch DALI was assigned to DALI groups using the IRT 8099/10 omniprog the group number assigned will be used auto-assignment of the devices. Example: If an OccuSwitch was given the group number 8 pressing the Auto-Assign button will assign the ballast representation of the device to Lamp Actuator object 8, the occupancy sensor representation to Occupancy Sensor object 8, and the button representation to Button object 8. If these objects are not available, a different object will be assigned in case of the ballast and button part, while no assignment will be done in case of the sensor part.

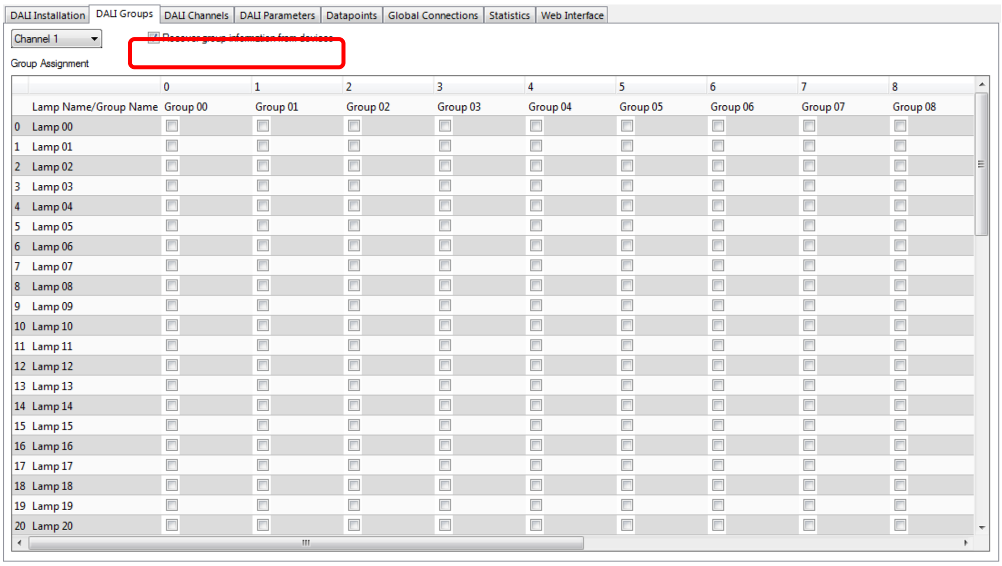


Figure 240: Be sure to check the **Recover group information from devices** checkbox if group addresses were already assigned with the IRT 8099/10 omniprog.

Be sure the **Recover group information from devices** checkbox (see Section 7.5.2) is checked to keep the group information set with the omniprog when commissioning the DALI channel with the LINX Configurator (see Figure 240).

8.4.4.5 Schneider Electric DALI Sensor LA-11

For proper lux sensor value readings ensure that the position of the dial labeled **Brightness** is **not** set to **Test** (see Figure 241).

Figure 241: Ensure position of Brightness dial is not set to Test.

8.4.4.6 ThebenHTS PlanoSpot 360 DALI

This sensor reports multiple lux level readings. The value reported via the assigned Light Sensor objects is the (primary) integral value. If the lux level sensor object provides a second lux level value (BACnet version only) this is the value measured in the direction “window”.

8.4.4.7 Lunatone DALI Temp-Sensor

When using the Lunatone DALI Temp-Sensor the temperature value reported by the sensor is provided via the lux level output (*nviLuxLevel*, *Present_Value*) of the corresponding Light Sensor object.

8.4.5 Power Failure Recovery

If a ballast signals that it has come back from power failure the L-DALI will restore the ballast's last known dim level. This behavior ensures that a consistent lighting situation is reestablished after a power failure.

8.4.6 DALI Channel Bridging

The L-DALI allow connecting two or more physical DALI channels to one virtual DALI channel. This operation mode is called DALI bridging. It is used if a DALI group shall contain more than 64 ballasts, that is, more ballasts than it is possible to accommodate on one DALI channel.

In the DALI bridging mode the L-DALI will forward any dim commands addressed to DALI groups and any DALI broadcasts received on one channel to the other bridged channels.

Note: *DALI bridging is only required if the large group shall be controlled by other DALI masters like DALI switches, buttons, or touch panels. If no such devices are used, it is recommended to use fan-out binding to the group's network variables (one on each CEA-709 node representing a DALI channel) instead of the bridging mode.*

8.4.7 Reducing ballast standby energy consumption

To reduce the standby energy consumed by DALI ballasts, the L-DALI allows switching off the mains power of all ballasts on a DALI channel using an external relay, whenever they all have a dim level of 0% (OFF).

Important: ***Do not use this functionality in case the channel contains DALI emergency lights or HID lamps!***

To use this feature the mains power of all DALI ballasts on a DALI channel must be switchable via a relay controlled by a BACnet or LONMARK IO-module (e.g. LOYTEC L-IOB, see Figure 242). The relay must switch off power, whenever the channel feedback of that DALI channel is 0% and must switch power on otherwise. Depending on the devices interface the channel feedback is available via the following data points:

- **CEA-709**: *nvoCHValueFb* in the corresponding Channel Actuator object (see Section 8.2.5.5).
- **BACnet**: *Present_Value* of the channels feedback Analog Input object (see Section 8.3.3.4).

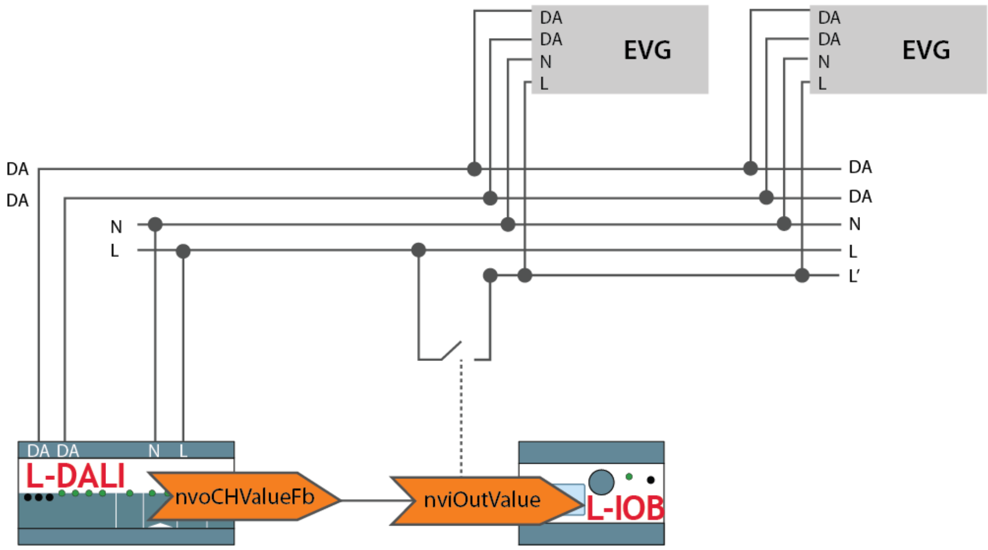


Figure 242: Wiring example for switching ballast mains using L-IOB relay output.

Important:	***The channels DALI bus power must not be switched off by the relay!***

The feature is enabled by setting the **Mains-Off handling** mode of the channel to **L-DALI** either using the LINX Configurator software (see Section 7.5.1.1) or the Web-Interface (see Section 5.3.4.10).

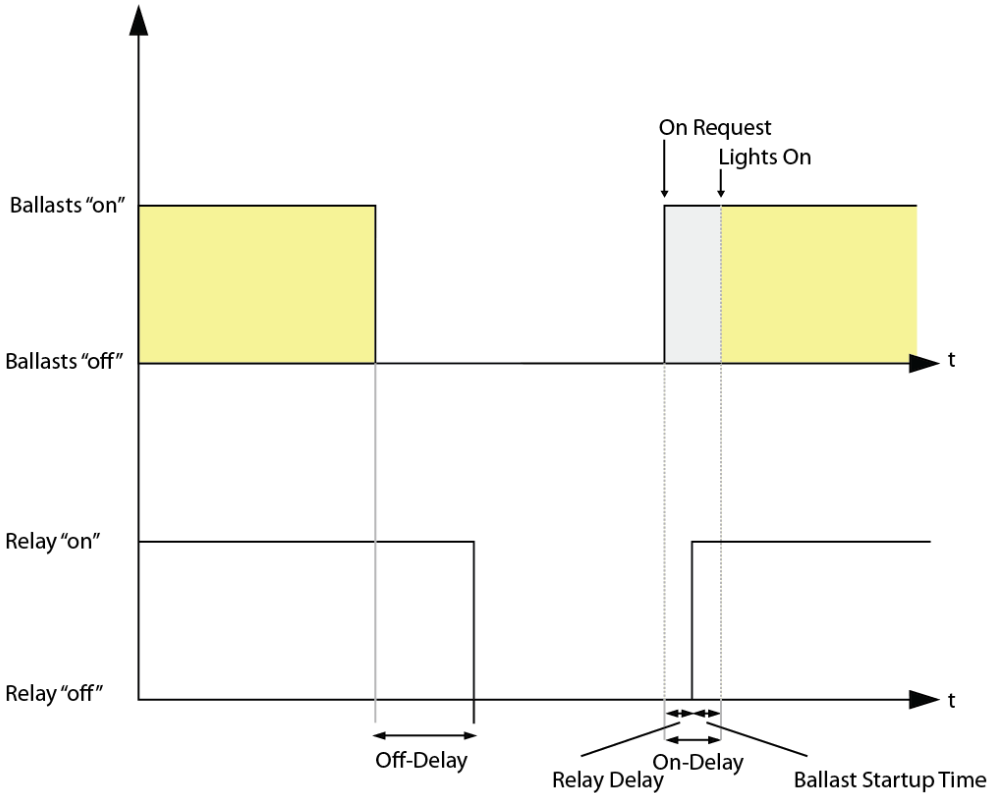


Figure 243: Timing parameters of Mains-Off handling function.

The function has two parameters (see Figure 243):

- **Off-Delay**: Delay between detection that all DALI ballasts are at 0% and entering standby mode (switching off mains). Using smaller value increases the energy saving potential, but might decrease ballast life-time.

- **On-Delay**: Typical delay between commanding the mains to be switched on (via the corresponding data point) and the time the ballast is powered up and reacting to dim commands. Modify this value if you observe all ballasts on the channel briefly switching on when mains are turned on.

Note: *The On-Delay includes the delay due to communication via BACnet or CEA-709 respectively, the delay in the IO-module for switching the relay and the ballast startup time. The ballast startup time typically is below 500 ms (required by IEC 62386). If the On-Delay is too short, some ballasts might not be ready. They therefore will miss the startup sequence and will go to their POWER ON LEVEL. If the On-Delay is too long, switching on lights when returning from standby mode will be delayed longer than necessary.*

Alternatively, the same functionality is offered by special DALI devices (e.g. Tridonic PS2 standby). Set the **Mains-Off handling** mode of the channel to **External** if using such a device.

8.5 XML-DA OPC Server

8.5.1 Access Methods

LOYTEC devices with the built-in OPC server can expose data points over a Web service. The OPC tag namespace is built from the data point hierarchy, which has been configured by the Configurator software. The OPC server on the device implements the data access standard via the Web service interface XML-DA. The OPC XML-DA Web service is accessible via the URI

http://192.168.1.100/DA

where the IP address has to be replaced with the actual IP address of the device. The Web service is accessible over the same TCP port as the Web server. The default TCP port is 80. The Web server port can only be changed via the device configuration tab in the Configurator (see Section 7.16.2) or in the L-Config tool (see NIC User Manual [3]).

Since the Web service is easily routable on the Internet, the embedded OPC server implements the basic authentication method to protect the system from unauthorized access. The basic authentication involves the operator user and the password configured for this user. On how to configure the operator’s password, please refer to Section 5.1.

To disable the basic authentication, clear the operator’s password.

Note: *It is highly recommended to use basic authentication when exposing crucial data points over the Web service.*

To use the exposed OPC data points, there exist several possibilities:

- Use LOYTEC’s L-WEB visualization tool that comes free with any LOYTEC device,

- use LOYTEC’s L-VIS device as OPC XML/DA client, or

- use a standard OPC client or SCADA package, or

- create your own Web service client with custom Web Pages.

For L-WEB visualization tools and L-VIS OPC XML/DA client functionality see the corresponding manuals ([4] and [6]).

Standard OPC clients and SCADA packages, which shall visualize the device's data points, must conform to the OPC XML-DA standard. This means they must support the OPC Web service and not only the COM/DCOM protocol. If your SCADA package does not support OPC XML-DA, a PC-based bridge from XML-DA to the COM-based protocol can be used. The bridge software is running on a PC and translates from COM/DCOM requests into XML-DA Web service requests. The system is depicted in Figure 244.

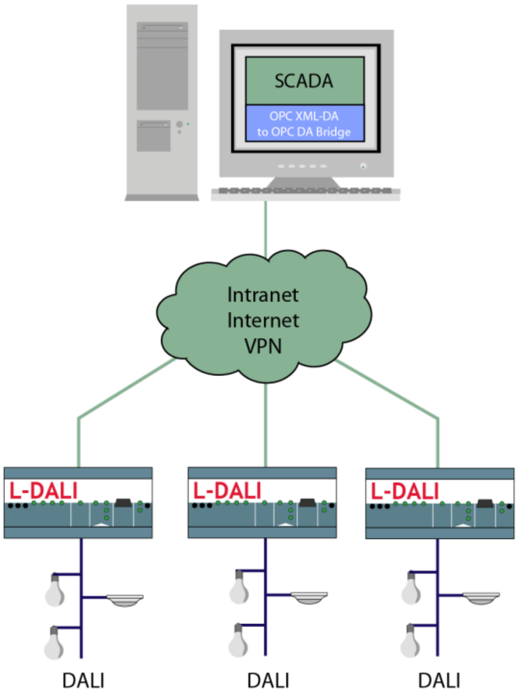


Figure 244: Using a XML-DA/DCOM bridge.

With the bridge is configured to access a number of OPC devices, the COM-based SCADA application can access a COM-based OPC server for each of those devices. The bridge software needs to be purchased from an OPC bridge software vendor.

If L-WEB is not used, customers can create their own XML-DA clients based on the WSDL for OPC XML-DA.

8.5.2 Data Points

The data point hierarchy as configured by the Configurator software is exposed to the OPC tag namespace by the device. This is done internally for all data points, which are marked for OPC exposure (i.e., have the OPC check-mark set).

Folders are translated into OPC nodes. Any of the data point classes, analog, binary, multi-state, string, and user, are exposed as OPC tags. Each OPC tag contains the value of the data point and some of its meta-data. An example of browsing the OPC tags on the device is shown in Figure 245.

The OPC quality property of a given OPC tag is coupled to the data point status. If a data point is offline or unreliable, the OPC quality property changes to *uncertain*.

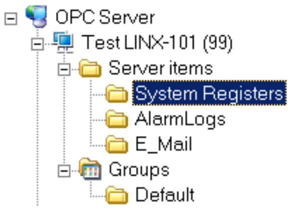


Figure 245: Client browsing the OPC tag namespace on a L-INX.

8.5.2.1 Analog

Analog data points are exposed as a one-to-one mapping to OPC tags. For each analog data point, an OPC tag is created. The OPC tag contains a number of OPC properties, which are derived from the data point’s properties:

- Item Canonical Data Type (SmallInt): This property indicates the data type ‘5’ (Double).
- Item Value (Double): The present data point value.
- Item Quality (SmallInt): The value quality. It is “good” if the data point is in normal state, or “uncertain” if the data point has an off-normal state, e.g., offline or unreliable.
- Item Timestamp (Date): This property contains the timestamp of the last value update.
- Item Access Rights (Integer): This property defines whether the tag is read-only or read/write.
- Item Description (String): This is the description of the data point.
- Item EU Type (Integer): This property is ‘1’.
- High EU (Double): This is the analog maximum value of the data point.
- Low EU (Double): This is the analog minimum value of the data point.
- EU Units (String): This is the human-readable engineering units text of the data point.

8.5.2.2 Binary

Binary data points are exposed as a one-to-one mapping to OPC tags. For each binary data point, an OPC tag is created. The OPC tag contains a number of OPC properties, which are derived from the data point’s properties:

- Item Canonical Data Type (SmallInt): This property indicates the data type ‘11’ (Boolean).
- Item Value (Boolean): The present data point value.
- Item Quality (SmallInt): The value quality. It is “good” if the data point is in normal state, or “uncertain” if the data point has an off-normal state, e.g., offline or unreliable.
- Item Timestamp (Date): This property contains the timestamp of the last value update.
- Item Access Rights (Integer): This property defines whether the tag is read-only or read/write.
- Item Description (String): This is the description of the data point.

- Contact Close Label (String): This property contains the active text of the binary data point.

- Contact Open Label (String): This property contains the inactive text of the binary data point.

8.5.2.3 Multi-state

Multi-state data points are exposed as a one-to-one mapping to OPC tags. For each multi-state data point an OPC tag is created. The OPC tag contains a number of OPC properties, which are derived from the data point's properties:

- Item Canonical Data Type (SmallInt): This property indicates the data type '3' (Integer).

- Item Value (Integer): The present data point value.

- Item Quality (SmallInt): The value quality. It is "good" if the data point is in normal state, or "uncertain" if the data point has an off-normal state, e.g., offline or unreliable.

- Item Timestamp (Date): This property contains the timestamp of the last value update.

- Item Access Rights (Integer): This property defines whether the tag is read-only or read/write.

- Item Description (String): This is the description of the data point.

- Item EU Type (Integer): This property is '2' for multi-state.

- Enumerated EU (Array of String): This property contains the state texts of the data point.

8.5.2.4 User Type

User-type data points contain a byte array of user-defined data. Data points of user-type are also exposed as a one-to-one mapping to OPC tags. For each such data point, an OPC tag is created. The item value of the user-defined data is a hex string without whitespace representing the byte array, e.g., "B034". The OPC tag contains a number of OPC properties, which are derived from the data point's properties:

- Item Canonical Data Type (SmallInt): This property indicates the data type '8' (String).

- Item Value (String): A hex string without whitespace representing the byte array.

- Item Quality (SmallInt): The value quality. It is "good" if the data point is in normal state, or "uncertain" if the data point has an off-normal state, e.g., offline or unreliable.

- Item Timestamp (Date): This property contains the timestamp of the last value update.

- Item Access Rights (Integer): This property defines whether the tag is read-only or read/write.

- Item Description (String): This is the description of the data point.

8.5.2.5 String

String data points contain a string of text characters. Data points of string type are also exposed as a one-to-one mapping to OPC tags. For each such data point, an OPC tag is created. The item value of the tag is the string data, e.g., “Room4”. The OPC tag contains a number of OPC properties, which are derived from the data point’s properties:

- Item Canonical Data Type (SmallInt): This property indicates the data type ‘8’ (String).
- Item Value (String): The string value.
- Item Quality (SmallInt): The value quality. It is “good” if the data point is in normal state, or “uncertain” if the data point has an off-normal state, e.g., offline or unreliable.
- Item Timestamp (Date): This property contains the timestamp of the last value update.
- Item Access Rights (Integer): This property defines whether the tag is read-only or read/write.
- Item Description (String): This is the description of the data point.

8.5.2.6 Structured Data Points

Structured data points are modeled as one user-type data point, which contains the entire structure value as a byte array. The respective structure fields are created as sub-data points of appropriate class.

The relation between user-type data point and sub-data points is also exposed to OPC. In this case, an OPC node is created for the user-type data point. In that node, the sub-data points are exposed as OPC tags. The entire structure is also exposed as a user-type OPC tag under the same OPC node.

Important! ***Deselect any un-used structure members from OPC exposure to reduce the number of total OPC tags.***

It is important to note, that when using structured data points the top-level and all its structure members are exposed as OPC tags by default. Using many structured data points may lead to exceeding the OPC tag limit. Please observe this limit in the Configurator’s statistics tab and deselect the **OPC Tag** check box for unwanted structure members. This helps to keep your configuration lean and improves the performance of the OPC server when browsing and subscribing.

8.5.3 AST Objects

The alarming, scheduling, and trending (AST) objects are more complex than regular data points. The OPC XML-DA standard does not have appropriate tags for those objects. Therefore, the device exposes AST objects as a set of OPC tags describing the object. All tags for one AST object are collected under an OPC node representing the AST object.

8.5.3.1 Scheduler Object

The device exposes the scheduler objects to OPC XML-DA tags. Each scheduler object is represented by a node in the OPC name space. The content of the schedule XML document referred to in this section must be compliant to the scheduleCfg schema. This schema can be found at the LOYTEC Web site. The XML documents can refer to the target namespace ‘http://www.loytec.com/xsd/scheduleCfg/1.0/’.

In that node, the following OPC tags are available:

- ServiceType (string, Read-only, const): This is a constant tag of type string, which contains “schedule”. It identifies this folder as a schedule folder. This can be used as an additional identification to the vendor-specific property of the folder tag.

- Schedule (string, read/write): This tag configures the schedule. The data type is string and the format is in XML. The XML document contains the *scheduleCfg* element as the root element.

- Caps (string, read-only): This tag contains the schedule capabilities. The data type is string and the format is in XML. The XML document contains the *scheduleCapabilities* element as the root element.

- CalItemPath (string, Read-only, const): This is an optional tag. If present, it contains the item path to the calendar object, that the schedule references. To read the calendar referenced by the schedule, use this item path and the “Calendar” item name to read the calendar XML document.

- EmbeddedCal (node): This is an optional OPC node. If present, it contains the OPC tags for the embedded calendar. The embedded calendar structure is as defined for calendar objects in Section 8.5.3.2.

8.5.3.2 Calendar Object

The device exposes the calendar objects to OPC XML-DA tags. Each calendar object is represented by a folder in the OPC name space. In that folder, the following OPC tags shall be available:

- ServiceType (string, Read-only, const): This is a constant tag of type string, which contains “calendar”. It identifies this folder as a calendar folder. This can be used as an additional identification to the vendor-specific property of the folder tag.

- Calendar (string, read/write): This tag configures the calendar. The data type is string and the format is in XML. This document contains the *calendarCfg* element as the root element.

- Caps (string, read-only): This tag contains the calendar capabilities. The data type is string and the format is in XML. The XML document contains the *calendarCapabilities* element as the root element.

8.5.3.3 Alarm Objects

The alarm objects on the device provide the *alarm summary* and can be used to acknowledge alarms. The alarm objects are exposed to XML-DA tags. Each alarm is uniquely identified by an XML alarm ID (XAID). The XAID must identify the alarm object and the alarm ID in that object. The XAID is used in the acknowledge service to identify the alarm. The XAID can also be transmitted in e-mail notifications.

Each alarm object is represented by a folder in the OPC name space. In that folder, the following OPC tags shall be available:

- ServiceType (string, Read-only, const): This is a constant tag of type string, which contains “alarm”. It identifies this folder as an alarm folder. This can be used as an additional identification to the vendor-specific property of the folder tag.

- Summary (string, Read-only): Reading from this tag, the current alarm summary can be obtained. The data type is string and the tag contains an XML document. This tag should not be subscribed to as it contains a large document. Subscribe to NotifyNewCnt instead, to get notified about new alarms. The root element of the XML document is the *alarmSummary* element.

- NotifyCnt (unsigned, Read-only): This tag is updated with an incremented notify count for each alarm update notification. This is the case for new or cleared alarm conditions, and for acknowledged alarms. Clients can subscribe to this tag in order to be notified about changes in the alarm summary. The client has then to read the complete alarm summary when notifications occur.

- NotifyNewCnt (unsigned, Read-only): This tag is updated with an incremented notify count each time a new alarm appears. This tag does not update when alarms are acknowledged or go inactive.

- Ack (string, Write): Writing to this tag acknowledges an alarm. The data type is string. The written data is an XML document, which contains the *alarmAck* element. The write must specify the XAID.

8.5.3.4 Trend Log Objects

Each trend log object on the device is represented by a folder in the OPC name space. This folder contains a number of tags describing and controlling the trend log. To retrieve log records, however, the XML-DA tag interface cannot be used. There are two options: retrieve the complete log as a CSV file, or use the LOYTEC proprietary Data Log Web service (XML-DL). That Web service uses the logHandle provided by a tag. The CSV file location can be obtained from a tag also.

- ServiceType (string, Read-only, const): This is a constant tag of type string, which contains “trendLog”, or “alarmLog”. It identifies this folder as a trend log, data log or alarm log folder. This can be used as an additional identification to the vendor-specific property of the node tag.

- Purge (Boolean, read/write): When writing TRUE to this tag, the log is purged.

- TotalCnt (unsignedInt, read-only): This tag contains the total number of logged records. This number can be larger than the BufferSize.

- BufferSize (unsignedInt, read/write): The size in records of the log buffer. Writing to this tag can resize the log buffer, if it is disabled.

- LogHandle (string, read-only, const): This handle specifies the data log. The logHandle must be used with the proprietary Data Log Web service.

- CsvFile (string, read-only, const): This tag specifies the file path and file name of the CSV data log file.

- CentralDL0, CentralDL1 (string, read/write): These tags are obsolete and kept for backward compatibility.

8.5.3.5 E-mail Templates

E-mail templates can be configured in the Configurator software. When an e-mail template is triggered, the corresponding e-mail is transmitted. The e-mail template can also be triggered over the OPC interface. Therefore, a node is added to the OPC name space for each e-mail template under the “E_Mail” node.

Each e-mail node is named after the e-mail template and contains the following OPC tags:

- ServiceType (string, Read-only, const): This is a constant tag of type string, which contains “email”. It identifies this folder as an e-mail template folder.

- Send (Boolean, read/write): When writing TRUE to this tag, the e-mail transmission is triggered.

9 Network Media

9.1 FT (LDALI-10X only)

The L-DALI FT port is fully compatible to the parameters specified by LONMARK for this channel. FT ports can also be used on Link Power (LP-10) channels. However, the L-DALI does not provide the power supply for Link Power channels.

When using the Free Topology Segment feature of the FT, only one termination (Figure 246) is required and can be placed anywhere on the free topology segment. Instead of building the termination, one can order the L-Term module (LT-33) from LOYTEC, which can be used to properly terminate the bus.

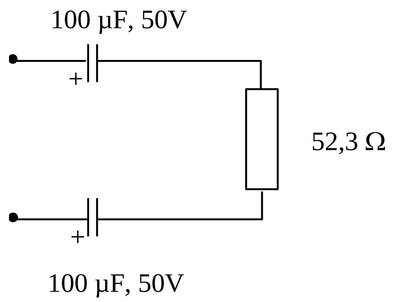


Figure 246: FT Free Topology Termination

In a double terminated bus topology, two terminations are required (Figure 247). These terminations need to be placed at each end of the bus. Here, also L-Term modules can be used at either end.

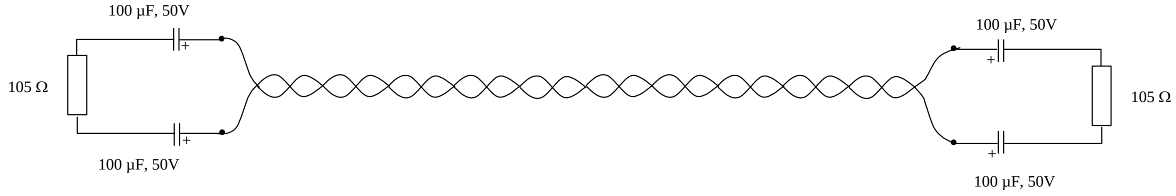


Figure 247: Termination in an FT Bus Topology

9.2 MS/TP (LDALI-ME204 only)

MS/TP is an RS-485 protocol and usually needs three wires (negative, positive, and reference). Polarity must be connected correctly. When using 2-wire MS/TP, earth ground must be connected to the negative terminal of the power supply. Never connect the positive terminal of the power supply to earth ground! See Section 4.9.1 for wiring instructions. Each MS/TP network segment must be properly terminated. Use an LT-04 network terminator connected at each of the two ends of the segment media.

The RS-485 transceiver of the device represents a full-load on the RS-485 bus. Consequently, a minimum of 31 devices are supported on the MS/TP channel. More devices may be possible, if they represent half-load or quarter-load. Please consult the third-party documentation. If more MS/TP devices need to be connected, use an RS-485 repeater to separate them electrically.

Logically, the MS/TP bus supports up to 255 devices. Each MS/TP device must be assigned a unique MAC address. Up to 127 MS/TP masters can be connected. Make sure, that the Max_Master setting includes the highest MS/TP master MAC address.

For operation of some slower devices on the MS/TP network it is recommended to set the following properties of the device object to fine-tune communication on the network:

- APDU_Timeout = 60000 (1 min).
- APDU_Segement_Timeout = 40000 (40 sec).

9.3 Redundant Ethernet (LDALI-E101-U/LDALI-E201-U only)

9.3.1 Ethernet Cabling Options

The L-DALI models LDALI-E101-U and LDALI-E201-U are equipped with two Ethernet ports, which are connected to an internal Ethernet switch. This allows for advanced cabling options to reduce cabling costs or to increase network resilience. For this discussion, the term *upstream* is used to designate the direction towards the network, which the devices are connected to. Likewise, the term *downstream* is used to designate devices more distant to the network which the devices are connected to.

Redundant cabling options are enabled by the Rapid Spanning Tree Protocol (RSTP) which is implemented in most managed switches. Please note, that this is a feature of the switch, not of the L-DALI, so that LOYTEC cannot give a guarantee that this will work with a particular switch model. In no case redundant cabling options will work with unmanaged switches. The older Spanning Tree Protocol (STP) should not be used for this type of application, as it converges too slowly.

Star topology: In the most basic setup, a device is connected to an Ethernet switch with one cable. This is called a star cabling because all devices are connected to a common upstream device. In this setup, the cable and the switch are single point of failures.

Chain topology: Because the L-DALI itself acts as an Ethernet switch, this device can be connected to a chain. This is a special form of the star topology. Its advantage is the reduced cabling costs. The disadvantage is the connection loss to downstream devices when an upstream device is powered-off, reset or removed. Also, the Ethernet bandwidth (100 MBit/s) is shared among all members of the chain. The last device has one unused Ethernet port, as it is not allowed to create Ethernet loops without STP. The recommended maximum number of daisy-chained devices is 20.

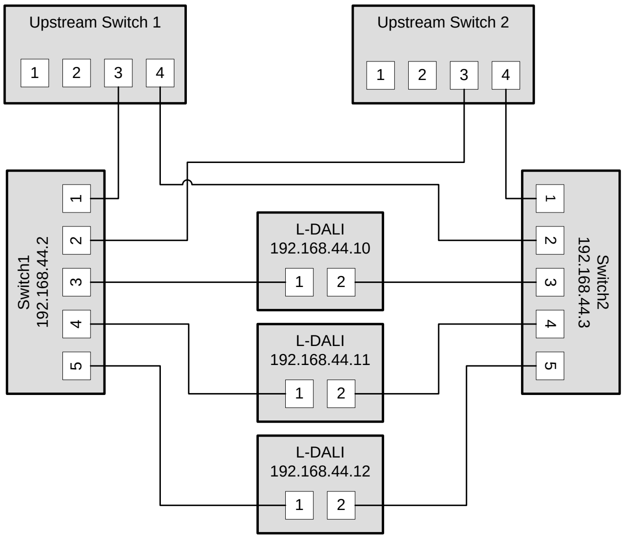


Figure 248: Fully redundant Ethernet topology

Fully redundant topology: Both Ethernet ports are connected to a different upstream switch. Thus, a single cable or upstream switch problem can be tolerated. This topology requires RSTP. In Figure 248, the L-DALI with IP addresses 192.168.44.10 to 192.168.44.12 are connected in this way. This connection scheme increases switch and cabling costs, but increases network resilience. Note that the upstream network is connected via the lowest-numbered ports. If this is not possible, the ports need to be configured to the lowest STP port priority value (which is the highest priority).

Ring topology: In this setup, the devices are connected in a chain and each end of the chain is connected to a different upstream switch. This topology requires RSTP. If a single device is powered off, the RSTP will automatically recalculate the spanning tree so that all other devices in the chain are reachable. Only if two devices are power-off at the same time, the devices between them will not have an Ethernet connection. In Figure 249, the L-DALI devices with IP addresses from 192.168.44.10 to 192.168.44.12 are connected in this way. The recommended maximum number of daisy-chained devices is 20.

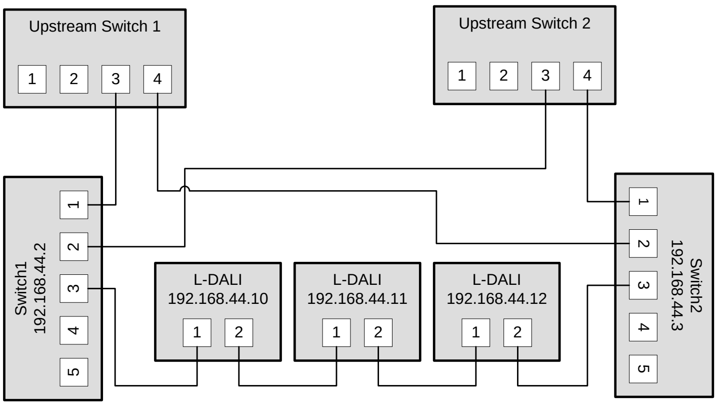


Figure 249: Ring Ethernet topology

9.3.2 Upstream Options

In case of redundant switches, there are two possible upstream topologies:

Single upstream connection: Switch1 (or Switch2, but not both) is connected to the upstream network while Switch2 only provides a redundant path to the Loytec devices. The redundant path is created by a direct Ethernet cable between Switch1 and Switch2 which needs to be plugged into a lower-numbered port than the L-DALI devices are connected to. If this is not possible, the STP port priority for the cross-connection cable needs to be set to a low value. The RSTP domain should be restricted to Switch1 and Switch2. This can be done by enabling a BPDU filter on the port on Upstream Switch 1. This will block all RSTP packets to enter the upstream network. A sample setup for this topology is shown in Figure 250.

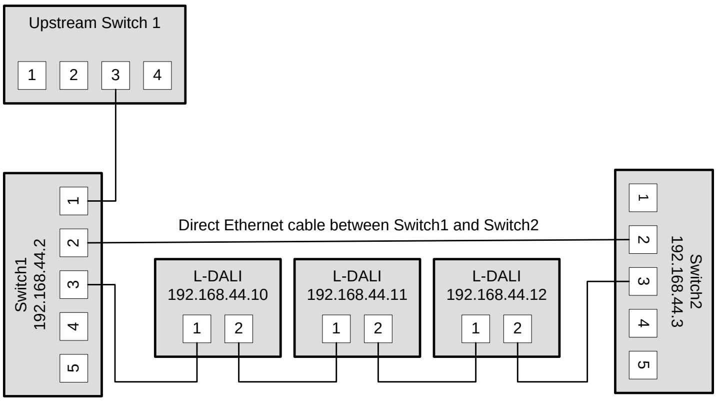


Figure 250: Single upstream connection.

Redundant upstream connection: Switch1 and Switch2 are both connected to the upstream network, either to two ports on the same switch or to two redundant upstream switches. In this case, RSTP is needed to ensure a loop-free topology between the upstream switches, Switch1 and Switch2, so the RSTP domain includes the upstream network and the chained L-DALI devices. The configuration of Switch1 and Switch2 need to ensure that they are not selected as the root bridge. If possible device communication should be bound to a separate VLAN and MSTP (Multiple Spanning Tree Protocol) should be employed to isolate the spanning tree operations. This topology is shown in Figure 248.

9.3.3 Preconditions

For the fully redundant and ring topology, the following preconditions have to be met:

- The upstream switches have to support the Rapid Spanning Tree Protocol (RSTP), as defined in IEEE 802.1w.
- The upstream switches have to provide a broadcast storm filter.
- Two distinct switches are required for each end of the device chain.
- Both upstream switches are connected to the same Ethernet network.

9.3.4 Switch Settings

The switches which connect the devices to the network need the following settings. Note that these are only recommendations or starting points. Each network with redundant connections needs testing and verification to prevent network loops.

- The STP bridge must be enabled.
- The STP bridge priority should be set to the minimum (61440), so that these switches are not elected as root bridges.
- The bridge mode should match the upstream bridge modes, preferable 802.1s or 802.1w.

If the upstream network uses RSTP, the timing parameters of the upstream networks must be used. Else the timing parameters should be set to minimum values for fast convergence:

- Bridge max age time: 6 seconds
- Hello time: 1 seconds
- Forward delay: 4 seconds
- All ports connected to Ethernet rings have to be configured as NON-EDGE ports, so that the RSTP can detect loops
- The switches should be configured to block broadcast storms. A recommended rate is 5% or 3000 packets/seconds.

The upstream switches need the following configuration:

- If a single upstream connection is used, the connected port on the upstream switch should have BPDU filtering enabled.
- If redundant upstream connections are used, the connected ports on the upstream switches should have a BPDU root guard enabled.

9.3.5 Testing

When the switches are configured and the devices are connected, the following tests are recommended. These tests are important to confirm that the STP changes due to topology changes to not interfere with the rest of the network.

- Check that no broadcast storms are sent into the upstream network by capturing traffic between Switch1, Switch2 and the Upstream switch. This test should be done continuously, especially during switch and device power cycles.
- Check that all devices can be reached (ICMP ping).

Execute these tests for these conditions:

- Power up all switches and devices. Wait until all devices are up, then test.
- Power-off Switch1. Wait approx. 10 seconds, then test.
- Power-on Switch2, power-off Switch1. Wait until Switch2 has booted, then test.
- Power-on Switch1. Wait until Switch1 has booted, then test.
- Reboot all L-DALI devices. Wait until the devices have booted, then test.
- Remove a single Ethernet cable. Wait approx. 10 seconds, then test. This test should be repeated for different cables. Make sure that at least the following connections are tested:
    - The connection between Switch1 and the L-DALI directly connected to Switch1.
    - The connection between Switch2 and the L-DALI directly connected to Switch2.
    - A connection in the L-DALI chain which is not connected directly to either Switch1 or Switch2.

9.3.6 Example switch configuration

The following example shows the configuration commands for Switch1, Switch2 and the upstream switch (HP Procurve syntax) in the setup shown in Figure 248.

Upstream switches:

```
config
spanning-tree
spanning-tree priority 8
spanning-tree 3,4 root-guard
spanning-tree hello-time 1
spanning-tree forward-delay 4
spanning-tree maximum-age 6
exit
```

Switch1 and Switch2:

```
config
spanning-tree
spanning-tree priority 15
spanning-tree 1,2 port-priority 0
spanning-tree 3-5 port-priority 8
spanning-tree hello-time 1
spanning-tree forward-delay 4
spanning-tree maximum-age 6
exit
```

10 Firmware Update

The L-DALI firmware supports remote upgrade over the network and the serial console.

To guarantee that the L-DALI is not destroyed due to a failed firmware update, the L-DALI firmware consists of two images:

- L-DALI fallback image,
- L-DALI primary image.

The L-DALI fallback image cannot be changed. Thus, if the update of the primary image fails or the image is destroyed by some other means, the fallback image is booted and allows reinstalling a valid primary image.

When the L-DALI boots up with the fallback image, the CEA-709 port LED and the STATUS LED are flashing red.

10.1 Firmware Update via the LINX Configurator (LDALI-10X only)

The L-DALI primary image can be updated using the LINX Configurator. For this purpose, it is recommended to have the L-DALI connected to the Ethernet and to have a valid IP configuration (see Section 5.2.2). The LINX Configurator must be installed (see Section 7.1.1).

To Update the Firmware using the LINX Configurator

1. Start the LINX Configurator from the Windows Start menu: **Start → Programs → LOYTEC LINX Configurator → LINX Configurator**.

2. Select the menu: **Connection → Connect via FTP**. This opens the FTP connection dialog as shown in Figure 251.

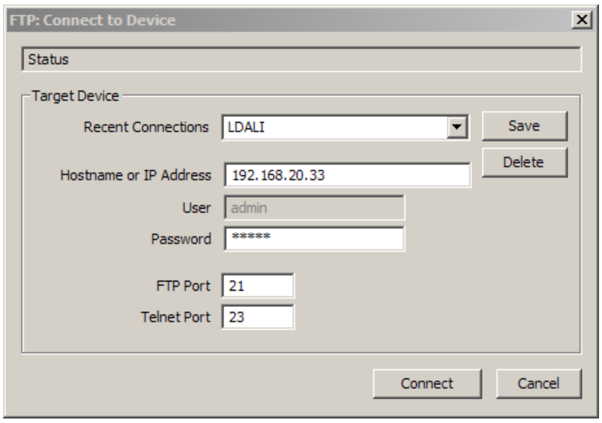


Figure 251: FTP connection dialog.

3. In the FTP connection dialog, enter the IP address of the L-DALI as well as the FTP user name and password. The default user name and password are 'admin' and 'loytec4u'. This can be changed via the Web interface (see Section 5.1) and reset via the console UI (see Section 11.2.2).

4. Click on **Connect**.

Note: *Alternatively, a connection via LNS can be established. However, mind that a firmware update via an FT-10 channel may take considerably longer.*

---

5. Select the menu: **Firmware → Update …**

6. This opens the Firmware Update dialog as shown in Figure 252. Click on the button "**…**" and select the firmware image ("ldali_10x_2_0_0.dl").

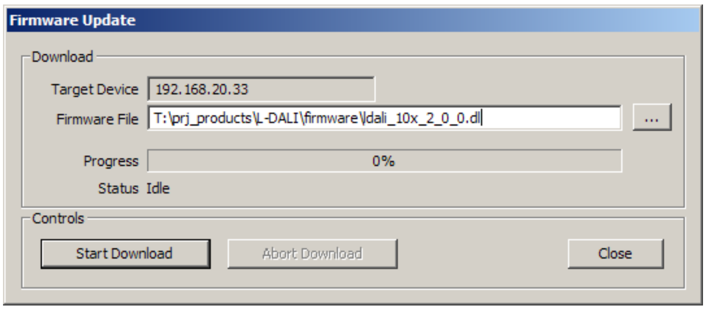


Figure 252: Firmware Update dialog of the LINX Configurator.

7. Click on **Start Download**.

8. Observe the download progress. When the download is complete, the dialog shown in Figure 253 appears.

Figure 253: FTP download success dialog.

9. Click **OK**.

10. In the Firmware Update dialog, click **Close**.

11. The device's firmware has now been successfully upgraded.

10.2 Firmware Update via the Web Interface

The device's firmware can also be upgraded using the Web interface. This option can be found in the **Config** menu under the **Firmware** item. For more details see Section 5.2.19.

10.3 Firmware Update via FTP

The L-DALI primary image can be updated using any FTP client. For this purpose, the L-DALI must be connected to the Ethernet and must have a valid IP configuration (see Section 3.1.2).

The following instructions explain how to update the firmware using the command line FTP client of Windows. However, the process can be accomplished using any other FTP client.

1. Start the command shell (Start->Run, enter “cmd” in the field “Open”, press OK).

2. Change to the directory containing the firmware image (e.g. “ldali_20x_2_2.dl”). Use the command “`cd`”.

3. Start the ftp client. Type “`ftp <ip-addr>`”, where <ip-addr> is the IP address of the L-DALI.

4. When asked for the user enter “admin”.

5. When asked for the password enter the password for the “admin” user. The default password is “loytec4u”.

6. Switch to binary mode by entering “`binary`”.

7. Upload the firmware image to the directory “dev” with the command “`put ldali_20x_2_2.dl dev/ldali_20x_primary.dl`” (LDALI-ME20X) or “`put ldali_10x_2_2.dl dev/ldali_10x_primary.dl`” (LDALI-3E10X).

8. After the firmware image has been successfully uploaded, the L-DALI automatically reboots.

9. Leave the ftp client with the command “`quit`”.

The complete output should look like shown in Figure 254.

```
C:\temp>ftp 192.168.1.254
Connected to 192.168.1.254.
220 RTEMS FTP server (Version 1.1-JWJ) ready.
User (192.168.1.254☹none)): admin
331 User name okay, need password.
Password:
230 User logged in.
ftp> binary
200 Type set to I.
ftp> put ldali_20x_2_2.dl dev/ldali_20x_primary.dl
200 PORT command successful.
150 Opening BINARY mode data connection.
226 Transfer complete.
FTP: 64d Bytes sent in 7,63seconds 88,83KB/s
ftp> quit
```

Figure 254: Typical output of the command line FTP client during a firmware update.

10.4 Firmware Update via the Console

To download the firmware via the console interface, the L-DALI must be connected to the RS-232 port of a PC via its console interface as described in Section 11.2.1. You will need the LOYTEC serial upgrade tool (LSU Tool), which can be downloaded from our homepage at www.loytec.com.

Please make sure that the L-DALI console shows the main menu otherwise navigate to the main menu or simply reset the L-DALI.

To Upgrade via the Console

1. Double click on the *.dlc file that comes with the new firmware package. This should start the LSU Tool and load the firmware image referenced in the dlc file. Please note that the dlc file and the dl file must be stored in the same folder. The start window of the LSU tool is shown in Figure 255.

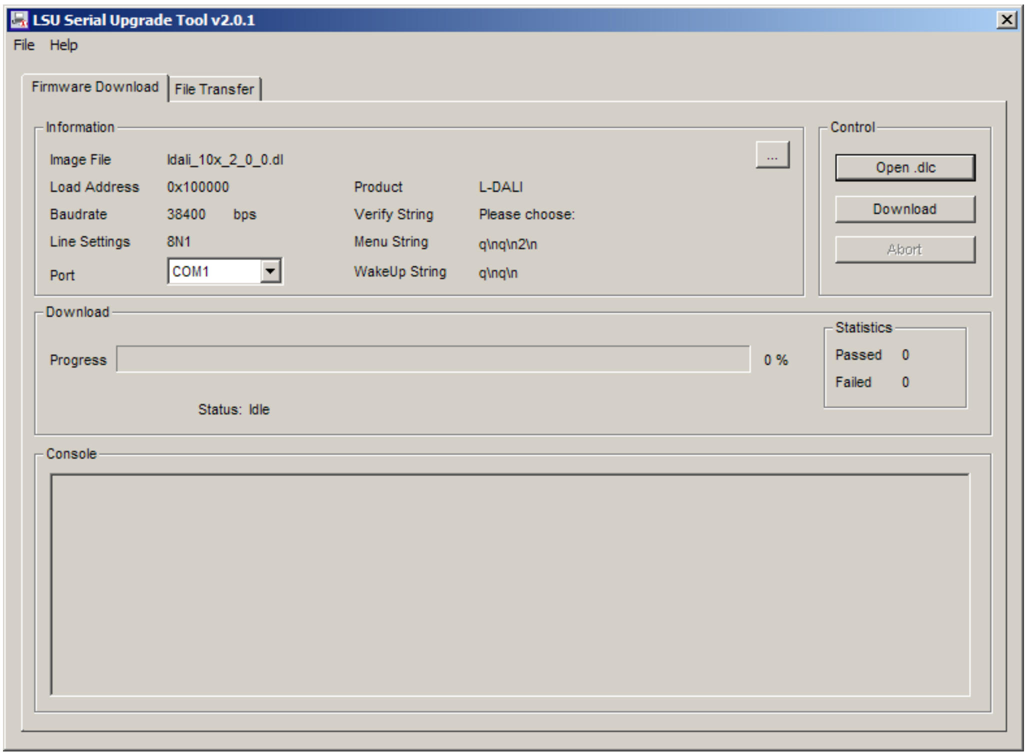


Figure 255: LSU Serial Upgrade Tool in Idle Mode

2. If the L-DALI is not connected to COM1 you can change the port to COM2, COM3, or COM4. Make sure that the product shown under "Product" matches the device you are upgrading. Press **Download** to start the download. A progress bar as shown in Figure 256 can be seen.

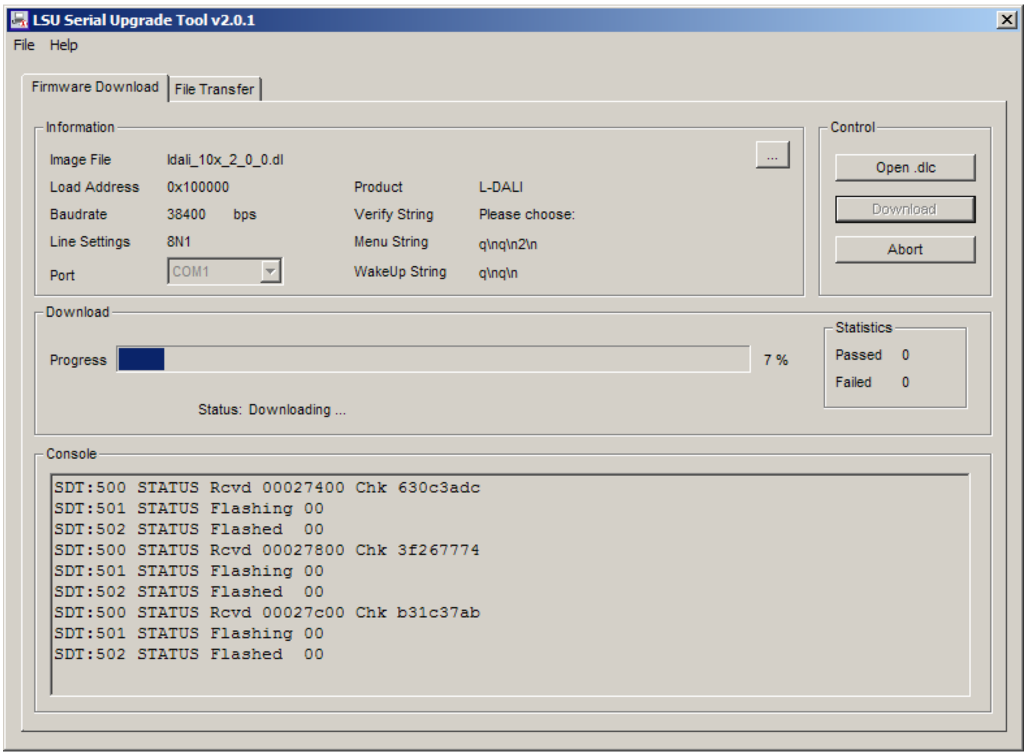


Figure 256: Progress Bar during Firmware Download.

3. If the upgrade is successful, the following window appears (Figure 257).

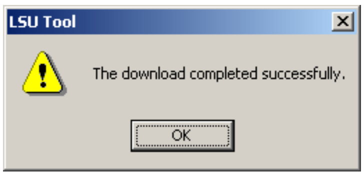


Figure 257: Successful Firmware Upgrade

4. Double check that the new firmware is executed by selecting '1' and pressing **Enter** in the console window. This will bring up the device information which shows the current firmware version.

11 Troubleshooting

11.1 Technical Support

LOYTEC offers free telephone and e-mail support for our L-DALI product series. If none of the above descriptions solves your specific problem please contact us at the following address:

LOYTEC electronics GmbH
Blumengasse 35
A-1170 Vienna
Austria / Europe

email: support@loytec.com
web: http://www.loytec.com
tel: ***+43/1/4020805-100***
fax: ***+43/1/4020805-99***

or

LOYTEC Americas Inc.
N27W23957 Paul Road
Suite 103
Pewaukee, WI 53072
USA

email: support@loytec-americas.com
web: http://www.loytec-americas.com
tel: ***+1 (512) 402 5319***
fax: ***+1 (262) 408 5238***

or

LOYTEC Asia Corporation Ltd.
16F.-3, No. 155, Zhongyang Rd
Xindian District
New Taipei City 23150
Taiwan

email: support-asia@loytec.com
tel: ***+886 (2) 8913 7838***
fax: ***+886 (2) 8913 7830***

11.2 Statistics on the Console

11.2.1 Connecting to the Console

Use a PC terminal program with the communication settings set to 38,400 bps / 8 data bits / no parity / 1 stop bit / no handshake. To connect COM1 of the PC to the Console on the device, use a standard null-modem cable with full handshaking. Power up the device or press **Return** if the device is already running. The menu shown in Figure 258 should appear on the terminal.

```
Device Main Menu
================

[1] Show device information
[2] Serial firmware upgrade
[3] System configuration
[4] DALI maintenance
[5] IP configuration
[7] CEA-709 configuration
[8] Reset configuration (factory defaults)
[9] Device statistics

[a] Data Points

[0] Reset device

Please choose:
```

Figure 258: Console Main Menu.

11.2.2 Reset configuration (load factory defaults)

Select item ‘8’ in the console main menu. This menu item allows resetting the device into its factory default state. The menu appears as shown in Figure 259.

```
Reset Configuration Menu
========================

[1] Reset everything to factory defaults
[3] Reset all passwords
[4] Clear data point configuration

[q] Quit

Please choose:
```

Figure 259: Reset to Factory Defaults Menu.

Select option ‘1’ to reset the entire device to factory defaults (including error log, configuration files, passwords etc.). Select option ‘3’ to reset all passwords (Web interface, FTP server etc.) to factory defaults.

Select option ‘4’ to clear all configured data points, parameters and the DALI configuration. The L-DALI must be rebooted to let the changes take effect.

11.2.3 Device Statistics Menu

Select ‘9’ from the device main menu to get to the device statistics menu. This menu holds relevant information regarding the device statistics of the device. This section describes those statistics, which are not available on the Web UI. The device statistics menu is shown in Figure 260. Use this menu only for debugging purposes. There is no need to access this menu if the network is running smoothly.

```
Statistics Menu
```

```
[2] Show CEA-709 application statistics
[4] Show IP statistics
[5] Show OPC statistics
[8] Show DPAL statistics
[9] Show Reg DPAL statistics

[q] Quit

Please choose:
```

Figure 260: Device Statistics Menu on the Console.

11.2.3.1 IP statistics

A sample console output is shown in Figure 261.

```
*********** INTERFACE STATISTICS ************
***** lo0 *****
Address:127.0.0.1
Flags: Up Loopback Running Multicast
Send queue limit:50  length:0  Dropped:0
***** eth0 *****
Address:192.168.0.2   Broadcast Address:192.168.0.255
Flags: Up Broadcast Running Simplex Multicast
Send queue limit:50  length:0  Dropped:0
Network Driver Stats for CS8900 :
    rx ready len -     50    rx loaded len -     0
     rx packets -    931      tx packets -    165
      rx bytes -   78480       tx bytes -   13627
    rx interrupts -    931    tx interrupts -    165
     rx dropped -     0      rx no mbuf -     0
    rx no custers -     0  rx oversize errors -     0
    rx crc errors -     0    rx runt errors -     0
  rx missed errors -     0         tx ok -    165
    tx collisions -     0    tx bid errors -     0
 tx wait for rdy4tx -     0      tx rdy4tx -     0
 tx underrun errors -     0      tx dropped -     2
     tx resends -     0    int swint req -    2094
    int swint res -    2094      int lockup -     0
     interrupts -    3189

************ MBUF STATISTICS ************
mbufs: 512  clusters: 64  free: 14
drops:  0    waits:  0 drains:  0
   free:461      data:51     header:0      socket:0
    pcb:0      rtable:0      htable:0      atable:0
  soname:0      soopts:0      ftable:0      rights:0
  ifaddr:0     control:0     oobdata:0

************ IP Statistics ************
       total packets received     922
 datagrams delivered to upper level     922
  total ip packets generated here     158

Destination   Gateway/Mask/Hw  Flags   Refs   Use Expire
Interface
default     192.168.0.1    UGS     6    0   0 eth0
62.178.55.77  192.168.0.1    UGH     0    1  3606 eth0
62.178.95.96  192.168.0.1    UGH     0    1  3606 eth0
81.109.145.243 192.168.0.1    UGH     0    1  3606 eth0
81.109.251.36  192.168.0.1    UGH     0    1  3606 eth0
127.0.0.1    127.0.0.1     UH     0    0   0 lo0
130.140.10.21  192.168.0.1    UGH     1    6   0 eth0
192.168.0.0   255.255.255.0   U      0    0   3 eth0
192.168.0.1   00:04:5A:26:96:1F UHL     7    0  1722 eth0
213.18.80.166  192.168.0.1    UGH     1   148   0 eth0
************ TCP Statistics ************

************ UDP Statistics ************
        total input packets     924
        total output packets     158

************ ICMP Statistics ************
```

Figure 261: IP Statistics.

The IP statistics menu has the additional feature of displaying any IP address conflicts. If the device's IP address conflicts with another host on the network, the banner shown in Figure 262 is displayed.

```
WARNING: Conflicting IP address detected!
     IP address 10.125.123.95 also used by device with MAC address
     00 04 5A CC 10 41!

Clear IP conflict history (y/n):
```

Figure 262: IP Address Conflict.

As useful information, the MAC address of the conflicting host is shown. If the information about this conflict shall be cleared, enter 'y'. If 'n' is selected, the conflict will show up again the next time this menu is entered.

11.3 DALI Troubleshooting

If you are experiencing problems with your DALI systems please follow this check list for troubleshooting:

- **Check L-DALI firmware version**: Ensure you are using the latest L-DALI firmware version available via the LOYTEC website (http://www.loytec.com).

- **Test ballast wiring**: On devices without LCD UI press the ON/OFF/AUTO button (see Section 4.5.2) once and verify all DALI luminaires connected to the channels connected to the L-DALI switch on, press it again and verify that they switch off. Finally press the button again to return to normal operation. On device with LCD UI perform the same test by using the Manual Override menu item in the LCD UI sub-menu DALI (see Section 4.6).

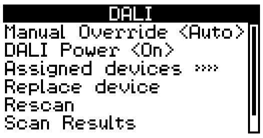


Figure 263: DALI Menu on LCD UI.

  If not all ballasts switch on and off, check the DALI and mains wiring of the affected ballasts. To check the DALI wiring, measure the voltage between the DALI terminals of the luminaire's ballast. It should be between 16 V DC and 22.5 V DC and must not fall below 14 V DC. If wiring is OK, check whether lamp or ballast are broken.

- **Check device limits**: Verify the number of ballast does not exceed the limits given in Section 8.4.2.

- **Check cable length**: Verify DALI lines are no longer than a total of 300 m per DALI channel.

- **Check supported device types**: If your DALI network contains other devices than just DALI ballasts (e.g. sensors, buttons, etc.) check the L-DALI Compatibility list (see [9]) whether those devices are supported by the L-DALI.

- **Check Tridonic MSensor setup**: If your system contains sensors of type Tridonic MSensor please check the dial on the back of the sensor is set up correctly (as described in Section 8.4.4.3).

11.4 DALI Error Codes

Table 44 lists typical error codes which can be observed during a DALI scan or device assignment.

Code	Description	Possible Reason
5	Cannot communicate on DALI bus.	• Problem with bus power: defective or non-standard. • Problem with LDALI transceiver (RMA). Use a different port to verify.
62	Device does not respond to QUERY.	• Communication problem. Maybe due to 116mA bus power supply being too weak. • Device not fully standard compatible. • Bad wiring.
118	Cannot communicate with device. This error typically is non-critical (device is added ok).	• Communication problem. Maybe due to 116mA bus power supply being too weak. • Device not fully standard compatible. • Bad wiring.

Table 44: DALI error codes.

11.5 DALI Protocol Analyzer

A DALI protocol analyzer is available via the console interface and via telnet.

11.5.1 Starting DALI PA via Console

Proceed as follows to start the DALI protocol analyzer via the console UI:

1. Connect to serial interface of L-DALI as described in Section 3.3.1.

2. In the console main menu select menu item “[4] DALI maintenance”. The resulting menu is shown in Figure 264.

```
DALI Maintenance Menu
================================================================

[1]  DALI channel          : 1
[2]  DALI configuration
[3]  DALI statistics
[4]  DALI analyzer
[5]  DALI monitor
[6]  DALI communication

[q]  Quit

Please choose:
```

Figure 264: DALI Maintenance Menu.

3. Use menu item “[1] DALI channel” to select the channel to start the DALI protocol analyzer on.

4. Use menu item “[4] DALI analyzer” to start the DALI analyzer.

5. When asked to enter a filter mask, press RETURN or enter a filter mask as described in Table 45.

Filter flag	Description
0x00000001	Do not show frames sent by other DALI masters.
0x00000002	Do not show frames sent by the L-DALI
0x00000004	Show frames with frame errors.
0x00000008	Only show DALI dim commands.

Table 45: DALI Protocol Analyzer Filter Flags.

6. Press any key to stop the DALI analyzer again.

11.5.2 Starting DALI PA via Telnet

Proceed as follows to start the DALI protocol analyzer via the telnet interface:

1. Connect to your L-DALI using a telnet client and log in with the “admin” user. The login credentials are identical to the ones of the Web Interface (see Section 5.1).

2. Start the DALI protocol analyzer using the command

   ```
   dali_pa <channel> [filter]
   ```

   where <channel> gives the DALI channel (1-4), which has to be analyzed and [filter] is an optional filter mask as described in Table 45.

3. Press any key to stop the DALI analyzer again.

11.5.3 DALI PA Information

Typical output from the DALI protocol analyzer looks like this:

```
11:08:05.284 45.00TE -> REQ  s03 QUERY STATUS
```

Each line contains the following information:

- **Timestamp** (Example: “11:08:05.284”): Local time on the device when the frame was received (end of frame).

- **Settling time** (Example: “45.00TE”): Settling time between this and the previous frame in Te (1 Te = 416.67 µs). The maximum value shown is “99TE”.

- **Direction** (Example: “->”): Frames sent by the L-DALI are marked by “->”, while frames received are marked by “<-“.

- **Frame type** (Example: “REQ”): Type of DALI frame. Some possible frame types are shown in Table 46.

Frame type	Description
REQ	DALI request
CMD	DALI command
RESP	DALI response
ECMD	DALI EFF command
EREQ	DALI EFF request
EFF	DALI EFF event
EVNT	DALI event (Philips)
???	Unknown type

Table 46: DALI frame types.

- **Destination address** (Example: “s03”): Destination address of the frame. Possible address types are:
    - **sXX:** DALI short address, where XX is the short address (00-63).
    - **gXX:** DALI group address, where XX is the group number (00-15).
    - **b*:** DALI broadcast address.
- **Message type & data** (Example: “QUERY STATUS”): Shows the DALI message type and the corresponding data (argument).

11.6 Packet Capture

11.6.1 Configure Remote Packet Capture

Remote packet capture is able to capture packets on the Ethernet. To enable the remote packet capture feature, go to the **Ethernet** port configuration and enable **Remote packet capture** as shown in Figure 265.

Figure 265: Remote packet capture port configuration.

The default **Port** setting may be changed to the desired port. Normally, this can be left at its default. If **No authentication** is selected, the device will allow incoming capture connections without requiring any credentials. If **Username and Password** is selected as authentication method, the client Wireshark will be required to provide valid credentials before the capture session can be started. Note, that only the users **admin** and **operator** are allowed to connect if this authentication method is selected.

Click the **Save Settings** button to save the configuration. The changes take effect and do not require to reboot the device. The remote capture can also be disabled again without a reboot.

11.6.2 Run Wireshark Remote Capture

The remote packet capture requires the use of Wireshark 1.6.11 with WinPCAP 4.1.2. Please update your Wireshark installation to this version or use a newer Wireshark version.

To add a remote capture port

1. Open Wireshark and choose the menu **Capture → Options…** . This opens the **Capture Options** dialog as shown in Figure 266.

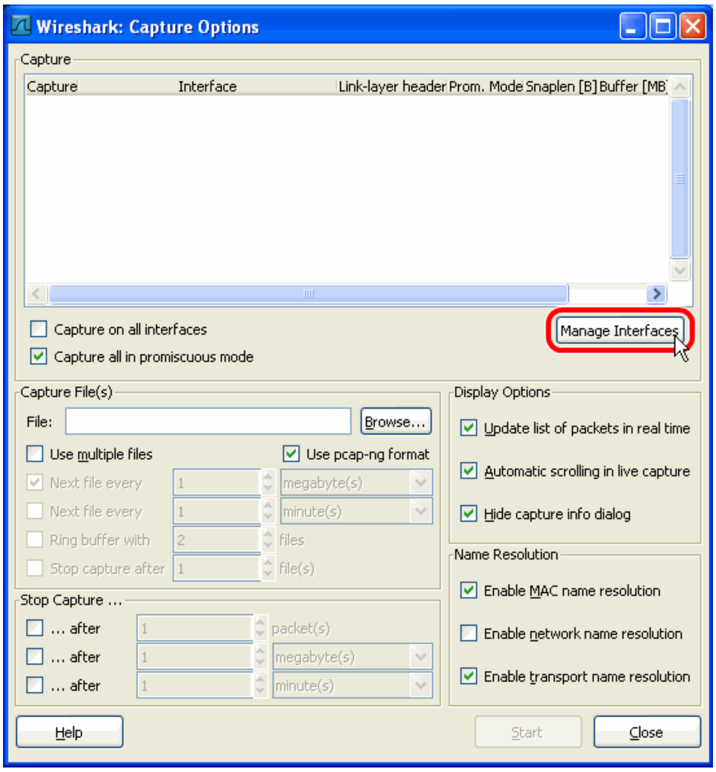


Figure 266: Wireshark Capture Options Dialog.

2. Click the **Manage Interfaces** button to open the **Add new interfaces** dialog.

3. Select the **Remote Interfaces** tab and click **Add** as shown in Figure 267.

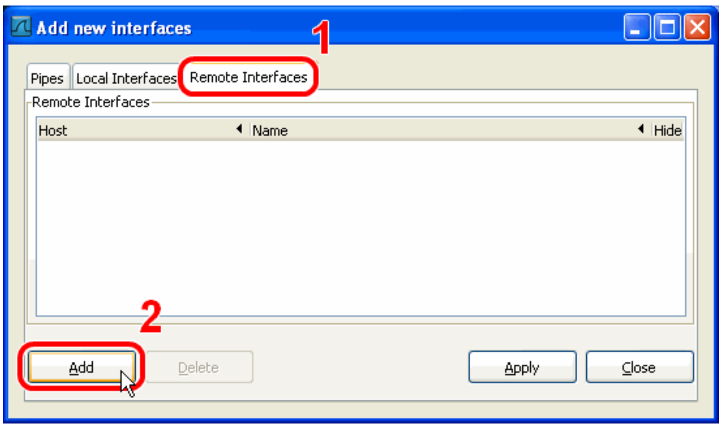


Figure 267: Wireshark Add New Interfaces Dialog.

4. Enter the correct settings for **Host** and **Port** (default 2002) and, if authentication is enabled, enter **Username** and **Password** in the corresponding fields as shown in Figure 268.

5. Note that only the users **admin** and **operator** are allowed to connect.

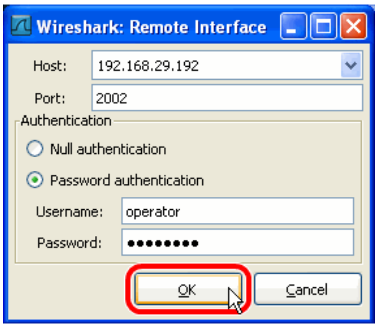


Figure 268: Wireshark Remote Interface Dialog.

6. Click **OK** to retrieve the interface list from the device.

7. If the connection to the device was established successfully, the **Remote Interfaces** list will be updated with information about all capture ports available on the device as shown in Figure 269.

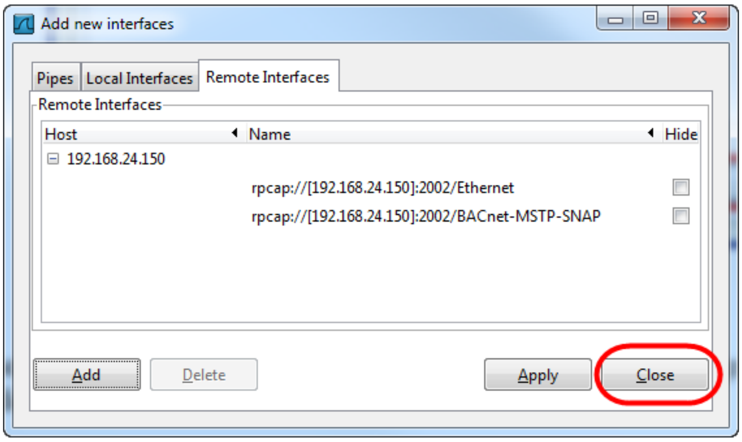


Figure 269: Added new interface to Wireshark.

8. Close the **Add new interfaces** and **Capture Options** dialogs to return to the main window.

To Start a Remote Capture

1. Select the created remote interface from the interface list in the main window. It is named ‘Raw Ethernet traffic’ for remote Ethernet dapture and ‘SNAP encapsulated BACnet MS/TP traffic’ for remote MS/TP capture.

2. Click the **Start** button as shown in Figure 270.

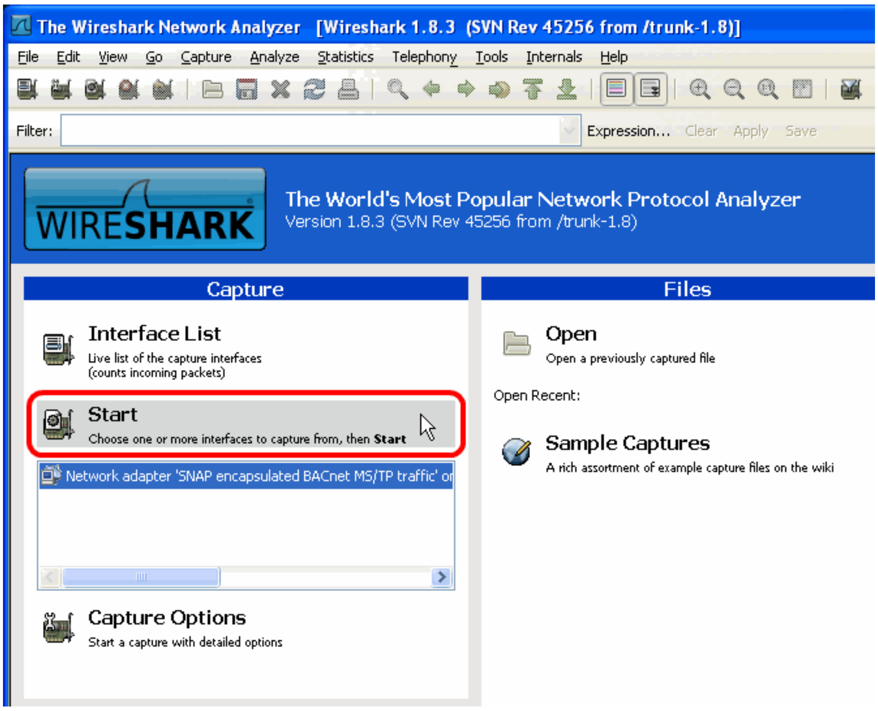


Figure 270: Start Remote Capture in Wireshark.

3. Wireshark will attempt to establish a connection to the device and, if successful, start displaying packets.

12 Security Hardening Guide

This guide contains security-relevant information for operating the product on IT networks. The information refers to the firmware version and the instructions found in the previous chapters of this User Manual.

12.1 Installation Instructions

Install the device over the Web interface:

- Set up the basic device functions and protocol settings as described in Section 3.1.2.
- Disable the FTP, and Telnet servers in the IP port configuration as described in Section 5.2.2.

Connect a serial console cable:

- Connect to the console as described in Section 11.2.1.
- Go to menu [3] system configuration.
- Disable the Web server in option [9].
- Save the settings by hitting [x] for exit and save.

12.2 Firmware

The device is equipped with one piece of software. This is the firmware image and its related firmware version. The firmware is distributed as a downloadable file. The device can be upgraded by placing the firmware image onto the device using the procedure described in Chapter 10.

12.3 Ports

This Section lists all ports, which may be used by the device. The ports are default settings for their respective services. If not stated otherwise, the ports can be changed.

Required Ports:

- 80 tcp: This port is opened by the Web server and the OPC XML-DA server. It can be disabled if OPC XML-DA is not required. The port can be changed.
- 1628 udp/tcp: This is the data exchange port for CEA-852 (LON over IP). It is required for the primary function of the device to exchange control network data between

routers over the IP network. Each device needs this port open. The port can be changed.

- 47808 udp: This is the data exchange port for BACnet/IP. It is required for the primary function of the device to exchange control network data between routers over the IP network. Each device needs this port open. The port can be changed.

Optional ports not necessary for the primary product function. They can be disabled as described in the installation instructions in Section 12.1:

- 21 tcp: This port is opened by the FTP server. The port can be changed and disabled.
- 23 tcp: This port is opened by the Telnet server. The port can be changed and disabled.

12.4 Services

Required services:

- CEA-852 (LON over IP): Primary function of the device. This service is in accordance with the standard ANSI/CEA-852-B.
- BACnet/IP: Primary function of the device. This service is in accordance with the standard ANSI/ASHRAE 135-2010.
- OPC XML-DA: This Web service provides access to data points over the OPC XML-DA standard.

Optional services not necessary for the primary product function. They can be disabled as described in the installation instructions in Section 12.1:

- HTTP: Web server. It provides a Web-based configuration UI. The Web UI can be disabled after setting up the device.
- FTP and Telnet: The FTP and Telnet server is used for connection to the device by the Configurator for configuration, firmware upgrade, and access to the log file. On devices without SSH these services must be enabled during device configuration.

12.5 Logging and Auditing

The device contains a log file, which can be read out over FTP or the Web server. This log contains information when the device started and when crucial communication errors occur. Other information such user log-on are not logged as they are not part of the primary services of this device.

Logged events:

- Time of the last power-on reset of the L-DALI device.
- Time and version of the last firmware upgrade.
- Time when the device configuration has been cleared or the device was reset to factory defaults.
- Commission of the CEA-709 nodes.
- Static errors in the device and data point configuration.
- System overload situations as one-time log messages since last power-on.
- Crucial communication errors as they occur.

13 Specifications

13.1 Physical Specifications

13.1.1 LDALI-3E10X/LDALI-ME204

Operating Voltage	12-35 VDC or 12-24 VAC ±10%
Power Consumption	typ. 3 W
In rush current	up to 950 mA @ 24 VAC
Operating Temperature (ambient)	0°C to + 50°C
Storage Temperature	10°C to +85°C
Humidity (non condensing) operating	10 to 90% RH @ 50°C
Humidity (non condensing) storage	10 to 90% RH @ 50°C
Enclosure	Installation enclosure 9 TE, DIN 43 880
Environmental Protection	IP 40 (enclosure); IP 20 (screw terminals)
Installation	DIN rail mounting (EN 50 022) or wall mounting

13.1.2 LDALI-3101-U/LDALI-E101-U/LDALI-E201-U

Operating Voltage	85-240 VAC 50/60 Hz
Power Consumption	typ. 7.5 W
Operating Temperature (ambient)	0°C to + 50°C
Storage Temperature	10°C to +85°C
Humidity (non condensing) operating	10 to 90% RH @ 50°C
Humidity (non condensing) storage	10 to 90% RH @ 50°C
Enclosure	Installation enclosure 6 TE, DIN 43 880
Environmental Protection	IP 40 (enclosure); IP 20 (screw terminals)

Installation	DIN rail mounting (EN 50 022) or wall mounting

13.2 Resource Limits

Table 47 specifies the resource limits of the different L-DALI models.

Limits \ Model	3E101	3E102	3E104	3101-U	E101-U	ME204	E201-U
OPC Tags	2000	2000	2000	-	2000	2000	2000
User Registers	1000	1000	1000	1000	1000	1000	1000
Address table entries/legacy[1]	512/15	512/15	512/15	512/15	512/15	-	-
Alias NVs[1]	1000	1000	1000	1000	1000	-	-
LONMARK Calendar objects/patterns per object	1/10	2/10	4/10	1/10	1/10	-	-
LONMARK Scheduler objects	16	32	64	16	16	-	-
LONMARK Alarm Servers	1	2	4	1	1	-	-
BACnet client mappings	-	-	-	-	-	1000	1000
BACnet scheduler objects	-	-	-	-	-	64	16
BACnet calendar objects	-	-	-	-	-	8	8
BACnet notification classes	-	-	-	-	-	16	4
Trend Logs	25	50	100	25	25	100	25
Total trended data points	256	256	256	256	256	256	256
Total aggregated size	1MB	1MB	1MB	1MB	1MB	1MB	1MB
E-mail templates	100	100	100	100	100	100	100
Math objects	100	100	100	100	100	100	100
Alarm Logs	10	10	10	10	10	10	10
L-WEB Clients (concurrent)	8	8	8	-	8	8	8
Connections (local)	1000	1000	1000	1000	1000	1000	1000
Connections (global)	250	250	250	-	250	250	250

Table 47: Resource limits of different L-DALI models

[1] Per CEA-709 interface.

14 References

[1] L-IP User Manual 6.0, LOYTEC electronics GmbH, Document № 88065911, September 2012.

[2] LIP-ME201 User Manual 5.0, LOYTEC electronics GmbH, Document № 88073507, January 2015.

[3] NIC User Manual 4.2, LOYTEC electronics GmbH, Document № 88067217, April 2013.

[4] LWEB-802/803 User Manual 2.2, LOYTEC electronics GmbH, Document № 88074215, March 2015.

[5] LWEB-900 User Manual 2.0, LOYTEC electronics GmbH, Document № 88081505, April 2015.

[6] L-VIS User Manual 5.1, LOYTEC electronics GmbH, Document № 88068520, February 2015.

[7] LIOB-10x/x5x User Manual 5.0, LOYTEC electronics GmbH, Document № 88078510, April 2014.

[8] L-INX/L-GATE User Manual 5.1, LOYTEC electronics GmbH, Document № 88073019, January 2015.

[9] AN011E L-DALI Compatibility List, LOYTEC electronics GmbH, Document № 86002004, May 2015.

15 Revision History

Date	Version	Author	Description
07-05-2010	1.0	AD	Initial revision V1.0
15-06-2010	1.0.1	JB	Minor corrections
08-11-2010	1.1	JB	Updated for L-DALI firmware 2.1 and Configurator 1.1
16-09-2011	2.0	JB	Cover LDALI-ME20X (BACnet) firmware 2.2 Minor improvements related to LDALI-3E10X documentation
14-11-2011	2.0.1	JB	Minor corrections and improvements
29-06-2012	3.0	JB	Cover LDALI-3E10X firmware 3.0 and LDALI Configurator 3.0
07-09-2012	3.0.1	JB	Minor corrections
11-10-2013	3.1	JB	Cover LDALI-10X/20X firmware 3.1 and LDALI Configurator 3.1. Cover LDALI-E101-U, LDALI-3101-U and LDALI-E201-U
13.08.2014	3.2	JB	Cover LDALI-10X/20X firmware 3.2 and LDALI Configurator 3.2. Cover LDALI-E101-U, LDALI-3101-U and LDALI-E201-U versions with Dual-Ethernet and internal DALI PS which can be switched off.
27.05.2015	5.2	JB	Cover LDALI-10X/20X firmware 5.2 and LINX Configurator 5.2 (replacing LDALI Configurator). Cover LDALI-MS1 and LDALI-BM1. Added more info on DALI basics and troubleshooting info. Added Property IDs for BACnet standard properties.