LIOB-x8x

L-IOB™ I/O Controller

User Manual

LOYTEC electronics GmbH

Contact

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

Version 6.0

Document № 88080307

Contents

Abbreviations

AST ..... Alarming, Scheduling, Trending
BACnet ..... Building Automation and Control Network
CEA-709 ..... Protocol standard for LONWORKS networks
CEA-852 ..... Protocol standard for CEA-709 over IP networks
COV ..... Change-Of-Value
CP ..... Configuration Property
CS ..... Configuration Server that manages CEA-852 IP devices
DHCP ..... Dynamic Host Configuration Protocol, RFC 2131, RFC 2132
I/O ..... Input/Output
LIOB ..... LOYTEC I/O bus
MAC ..... Media Access Control
NAT ..... Network Address Translation, see Internet RFC 1631
NTC ..... Negative Temperature Coefficient (Temperature Sensor)
NV ..... Network Variable
OPC ..... Open Process Control
PTC ..... Positive Temperature Coefficient (Temperature Sensor)
SCPT ..... Standard Configuration Property Type
SNVT ..... Standard Network Variable Type
UCPT ..... User-Defined Configuration Property Type
UI ..... User Interface

1 Introduction

1.1 Overview

The L-IOB I/O controllers LIOB-18x, LIOB-48x, and LIOB-58x combine physical inputs and outputs (I/Os) with programmable logic (IEC61131) and provide application specific data points through a CEA-709 or BACnet connection. The data points are also subject to AST (alarming, scheduling, and trending) functions of the L-IOB I/O controller. The I/O controllers come in different models with different I/O configurations, including analog outputs, digital outputs, digital inputs, and universal inputs, which can be freely configured. Some models feature a 2-port Ethernet switch. All LIOB-58x models are BTL-certified as B-BC devices.

Initial configuration of the I/O controllers is accomplished through the L-INX Configurator software. The logic application is developed using the provided IEC-61131 compliant design tool. Parameterization of I/Os, self-test, manual modes, override values, etc. can also be performed on the L-IOB LCD display.

1.2 Scope

This document covers LIOB-18x, LIOB-48x, and LIOB-58x controllers with firmware version 6.0 (or higher) and describes specific functions of those device models. Basic device operations are covered in the LOYTEC Device User Manual [1] and data point configuration is covered by the L-INX Configurator User Manual [2]. The usage of logiCAD itself is beyond the scope of this manual. Please refer to the logiCAD online help in case of additional questions.

2 What's New

2.1 New in LIOB-x8x 6.0.0

This section describes the major changes and new features in version 5.0.0. For a full list of changes refer to the Readme file.

New User Manual Structure

The LIOB-X8X User Manual has been split up into three parts: The LIOB-X8X User Manual, which now covers the specific functions of the LIOB-X8X device models. The LINX Configurator User Manual is a common description for using the Configurator software for the L-INX, L-GATE, L-ROC, L-IOB, and L-DALI product line. And the LOYTEC Device User Manual covers hardware, Web interface, LCD display and operating interfaces topics common to all LOYTEC devices.

Flexible CSV Import/Export

The entire CSV import/export engine of the LINX Configurator has been revisited. Previously special export options with fixed columns for CEA-709, Modbus, etc. have been replaced by a general CSV engine that is able to configure columns for any set of data point properties. Default property sets have been added, which can be modified by the user and saved for later use. The new CSV import/export allows modifying existing data point properties by batch edit in Excel or creating new data points. The property sets can be viewed and modifed in a comfortable export editor (see Figure 1). The new engine also integrates CSV data point templates. Learn more about this flexible import/export mechanism in the LINX Configurator User Manual [2].

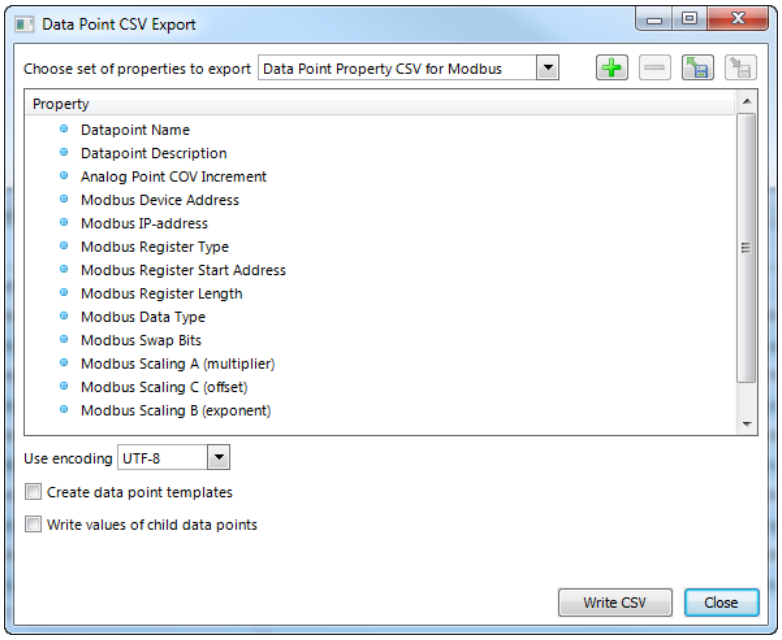


Figure 1: CSV export dialog with configurable columns.

Override Function for Data Points

The data point model in LOYTEC devices has been extended with a manual override function. On the Web interface and on the LCD display data point values can be overridden to manual values. If an override value is set, the running control logic no longer affects the data point value, neither does network communication. The override value is in effect until it is removed again by the user.

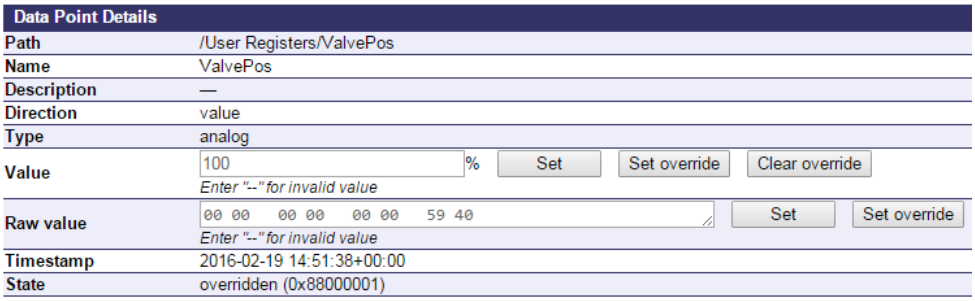


Figure 2: Overriding a data point on the Web interface.

Alarm Messages

The length of alarm messages has been extended to 250. This increased length comes in combination with a new %{path} variable placeholder that can identify the data point path in alarm messages. This is beneficial for mass-engineering alarm conditions on data points that have the same name but are located in different path locations. Alarm logs have also been updated to support the new maximum legnth. BACnet now supports the Event Message Texts property of an alarmed object, which allows retrieving the messages also after a reboot of an alarm client. And analog alarm conditions have been extended to specify a high limit and a low limit alarm message.

Trend Logs

The generic trend logs have been extended to support recording string values. These trends can be operated as polled, COV or triggered. These trend logs can be beneficial to record arbitrary messages The string recording is currently limited to generic trend logs only.

New E-Mail Placeholders

E-Mail configuration has become even more flexible by adding new variable placeholders. The variable placeholders can be augmented by name, description and path specifiers that expand to the respective information in the e-mail. A new date format specifier allows for readable timestamps and the %{last_timestamp} variable expands to the time the last e-mail was transmitted.

BTL Testplan 12 Certified

The BACnet certification of all BACnet models has been updated to comply with protocol revision 12. All new device models are now BTL certified.

2.2 New in LIOB-x8x 5.3.0

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

Extended Support for U.S. Units

An extended support for U.S. units has been built into LOYTEC devices. A device can now be configured to run either in the SI or the U.S. unit system by defining SI and U.S. units per data point as shown in Figure 3. When configured for U.S. units, all data points and L-IOB I/Os process values in the respective U.S. units. This includes the Web UI, the OPC server, the parameter file, global connections, and logiCAD programs. Also the Configurator displays values in U.S. units and conversions are done automatically. A device can even be switched from one unit system to the other. In this case all persistent parameters are reset to their default values in the chosen unit system. A system register shows the currently running unit system. This makes it easy to entirely engineer a device in U.S. units or build devices that can be configured for either European or U.S. markets and meet local requirements at the same time. For more information on units please refer to the LINX Configurator User Manual [2].

Figure 3: Configuration of SI and U.S. units on a data point

Forward Delay in Connections

For applications that require staggered start/stop and randomization, the device implements a configurable forward delay in local and global connections. All receive data point items can be configured with a delay. A received value is written to the data point after the delay expires. The delay can be a fixed time or a time interval, in which the actual delay is randomized. With a resolution of 0.1 seconds the total delay can be up to 100 minutes. Learn more about forward delay in the LINX Configurator User Manual [2].

Auto-Generate and Connect

The auto-connect feature has been extended:

- Support %{folder_descr} as a placeholder in auto-generate templates. This evaluates the description property of the parent folder.
- Generate into existing connections. Epecially when auto-generating the same source data points to different technologies, the same connections are used. This makes it easier to manage auto-generated local connections.

Enhanced Structure Support on the Web UI

The display of data point structures has been improved on the Web UI. The data point listing shows a textual short version of the structured value instead of a hexadecimal Byte buffer. Also CEA-709 structures of NVs without sub-data points are displayed and can be edited.

Project Documentation

A new feature on the device is a Web UI for creating and viewing project documentation on the device. The documentation editor requires admin rights and allows storing files on the device or creating documentation links as URLs. Both items can be viewed by guest users. Examples include storing cabling plans as PDF or adding links to a Web site containing manuals, plans or other useful project documentation. Consult the LOYTEC Device User Manual [1] to learn more about project documentation on the device.

BACnet

All previously extended BACnet features and new features have been BTL-certified in the 5.X firmware series. New features include:

- BACnet object names can be made writeable. With this new option the user can create data point configurations with generic object names and assign location-specific names later in his OWS.
- BACnet simple value objects are now supported. The user can create Large Analog Value, Signed Integer Value, Unsigned Integer Value, Character String Value and Octet String Value objects as needed. Also auto-connect supports these new object types.
- For Trend Log objects, the properties Notify_Type, Event_Enable, Notification_Class can be pre-configured in the data point configuration.

Data Point Creation from CSV Lists

The Configurator implements a new data point template CSV import feature, which can be used by external tools to generate a list of data points. Each line in this CSV references a data point template file that describes all properties of the data point. The CSV supplies name, description and path of the data point to be created. Additionally, data points can be automatically scheduled and trended. Learn more about data point templates in the LINX Configurator User Manual [2].

2.3 New in LIOB-x8x 5.1.0

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

Favorites

Enhancements for using favorites include engineering units, active/inactive texts, multi-state texts and value persistency for unlinked favorites. A new feature are structured favorites. For example, the structure of a SNVT_switch may now be created as a favorite. The top-level part of a structured favorite can be linked to a structured data point of the same type. Alternatively, the individual member elements of a structured favorite can be linked to other, individual data points. This way, one is able to create a structured favorite, that can be linked directly to a matching SNVT or to separate BACnet objects..

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 4.

Figure 4: New trend chart on the Web UI

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.

Alarm Server Ack-All Data Point

Alarm servers now provide a special ackAll property data point. When writing TRUE on that data point all currently active alarms on the alarm server are acknowledged.

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.

Output NVs with Integrated Feedback

Output network variables (NVs) are used to send updates to remote nodes. In order to poll back the actual value of the remote variable, a feedback data point had to be created for the output NV. Now changing the direction on the NV from output to value, an integrated feedback function is activated without the need for an extra data point. This becomes especially useful in data point templates.

2.4 New in LIOB-x8x 5.0.0

This section describes the major changes and new features in version 5.0.0. For a full list of changes refer to the Readme file.

New L-IOB Models

The new models LIOB-484 and LIOB-584 with a 2-port Ethernet switch and a built-in pressure sensor are supported by the firmware. See Sections 11.4, and 11.5.

2-Port Ethernet Switch Support

The new firmware supports all models with a built-in 2-port Ethernet switch. See Section 9.4.

CEA-852 Configuration Server

The LIOB-48x and LIOB-58x models now feature a built-in CEA-852 configuration server..

Duration Mode of Digital Outputs

Digital outputs can be configured in a “Duration” mode. In this mode, the value written to the output is interpreted as a time period, the output shall stay active. When the period ends, the output is automatically switched inactive again.

Support of State-controlled Occupancy Sensors

The L-IOB now also supports occupancy sensors, which do not generate pulses but rather are only switched to the active state as long as occupancy is detected.

Simplified Offset Calibration in the LCD UI

The calibrated life value is now shown next to the offset value for simplified calibration in the LCD UI.

Switching Off Detection of Disconnected Sensors

A new flag is introduced, which allows switching off the detection of disconnected sensors so that every measured value is processed.

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 the LINX Configurator User Manual [2].

Optimum Start for Schedulers

Up to now, optimum start was limited to using a SNVT_tod_event in a CEA-709 scheduler. Now all schedulers (including BACnet and generic schedulers) support timeToNext and nextState data points for implementing optimum start algorithms also for those technologies. Schedulers are extended by property relations, which offer the time to the next state in minutes and the next state in data points. Also, the enable, enableFb (feedback), and presetName information is available as property relations (refer to the LINX Configurator User Manual [2]).

The BACnet scheduler object also has custom properties that expose time to next state and next state to BACnet. If standard BACnet objects are required, the property relations can be linked to any matching BACnet server object as shown in Figure 5 (BV_nextState, AV_timeToNext).

LINX-200 ▸ BACnet Port ▸ Scheduler

Scheduler Name	No.	Direction	OPC	Direction
◢ Schedule_BO_occupied	1	Value	☑	Value
enable	1.1	Value	☑	Value
enableFb	1.2	In	☑	In
nextPresetName	1.3	In	☑	In
nextState -> BACnet Port.Datapoints.BV_nextState	1.4	In	☑	In
presetName	1.5	In	☑	In
timeToNext -> BACnet Port.Datapoints.AV_timeToNext	1.6	In	☑	In
BACnet Port.Datapoints.BO_occupied	1.7			

Figure 5: BACnet scheduler with next state data points.

Favorites

Favorites have been vastly extended to be compatible with all uses that were previously only possible for regular data points. Favorites can now be trended, can have an alarm condition (reporting to a generic alarm server), and be scheduled. Historic filters can be applied to favorites. Favorites can also be used in e-mail templates, math objects and connections like any other data point. Furthermore, it is possible to link favorites to property relations and vice versa, e.g. link a favorite directly to a historic filter data point.

Historic Filters

Historic filters have been extended by a generic delta calculation between any of the defined filter items or the current value. This way, it is no longer necessary to create a separate math object to calculate the consumption of the previous month. Also, a definition for “last day of month” has been added to the monthly period.

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.

Data Point Polling

The receive timeout on input and value data points has been generalized. A receive timeout can now be defined on all technologies for input and value data points. Writing updates to those data points from any source (e.g. a global connection) resets the receive timeout.

Background polling can be enabled in the project settings, which allows a slow polling of all data points even if no pollcycle or dynamic polling has been activated on those data points. For more details refer to the LINX Configurator User Manual [2].

BACnet

All LIOB-58x models are now BTL-certified as B-BC devices. Previously extended BACnet features and new features have been BTL-certified in the 5.X firmware series. New features include:

- Intrinsic alarming for the Accumulator object.

- Option to keep OWS settings in BACnet properties also after a new configuration has been downloaded.

Conversion to Value Data Points

Firmware versions since 4.9.0 support value data points. The default behavior of the Configurator can be controlled in the project settings. This defines whether new value data points or the old "_Read/_Write" data point combinations shall be created. When using templates in L-WEB or L-VIS it is often an all-or-nothing approach, a mixture is not practical.

For making the transition in old projects to value data points, the Configurator now offers a conversion tool. Multi-select old read/write data points and choose the item **Convert to value** from the data point context menu. This converts the selected data points to the new value data points, leaving the IDs, default values and data point usages intact.

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.

Configurator Usability

The folder tree of the data point manager has been brought to state-of-the-art user concepts such as multi-select, drag-and-drop of folders, moving folders, deleting multiple folders and their contents. A name filter can be applied to quickly find folders in the folder tree.

The property tab has been extended by a property name filter. This makes it easy to find the desired property by typing a sub-string of its name in the filter. For a detailed description on the data point properties refer to the LINX Configurator User Manual [2].

Data point link navigation has been made easy by using a **Go to data point** context menu and speed button everywhere data point references are displayed. A data point usage report dialog shows all references to the selected data point and allows jumping to selected objects.

The new **PLC conflicts** tab shows all PLC write conflicts with other write usage of data points such as math objects, connection receivers, etc., while editing the project. This allows finding problems prior to the configuration download. The conflicts tab also provides easy navigation to data points listed as problematic. For more information refer to Section 6.4.7.

3 Quick-Start Guide

This chapter shows step-by-step instructions on how to configure the L-IOB I/O controller for a simple application.

3.1 Hardware Installation

A LIOB-18x I/O controller is connected to a CEA-709 network using the FT port of the L-IOB device. A LIOB-48x I/O controller is connected to a CEA-852 network using the Ethernet/IP port of the L-IOB device. A LIOB-58x I/O controller is connected to a BACnet network using the Ethernet/IP port of the L-IOB device. In all cases, the device must be powered e.g. using an LPOW-2415A power supply.

3.2 Commissioning or BACnet Setup

For LIOB-18x/48x models, use the appropriate L-IOB template of your LNS™ based network management tool (e.g. NL-220™ or LonMaker™) to create and commission a L-IOB device in the database. Once the device is created, the 'Configure' function of the network management tool can be used to invoke the Configurator Software.

For LIOB-58x models, the initial IP and BACnet setup must be done in the LCD UI or Web UI (see LOYTEC Device User Manual [1]).

The main page of the LCD UI of a LIOB-18x is shown in Figure 6. It displays the device name, project name, Programmable Logic (PLC) status, CPU load, supply voltage, and system temperature. In the row where the CPU load is displayed, there is a language symbol on the right side, where the LCD language can be chosen. Observe that changing the language requires a system reboot. In the LIOB-48x/58x LCD UI, the IP address and Ethernet status is shown instead of the PLC status.

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. The **I/O »»** menu is described in the LOYTEC Device User Manual [1]. The **Datapoints »»** menu allows browsing through the data points on the device.

Figure 6: Main Screen of the LCD UI

LIOB-48x/58x devices can additionally host one LIOB-45x/55x device in LIOB-IP mode. In this case, there will be an additional **LIOB-IP** menu on the main screen which is described in the LOYTEC Device User Manual [1].

The **Device Settings »»** menu allows configuring basic device settings. Navigate e.g. to the **Device Management »»** sub-menu, which is displayed in Figure 7.

Figure 7: Device Management Menu on the LCD UI

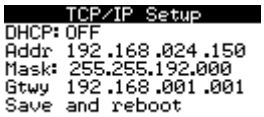
This menu gives you e.g. the following options for basic device configuration:

- **TCP/IP Setup (LIOB-48x/58x)**: IP Configuration page (IP-Address, etc.).
- **CEA-709 over IP (LIOB-48x):** CEA-852 setup page. Here the user can setup the LIOB-48x device to become part of a CEA-852 (CEA-709 over IP) channel. Further settings can be made in the Web interface.
- **Send ID messages**: send out a service pin message (LIOB-18x/48x) or I-Am message (LIOB-58x).
- **Reboot system**: By choosing this item, the device performs a full reboot.
- **Clear DP config**: By choosing this item, the user can clear the devive's entire data point configuration.
- **Factory Defaults**: By choosing this item, the user can reset the entire device to its factory default.
- **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**: changes the display contrast.
- **Language**: changes the LCD language. Observe that this requires a device reboot.
- **Reset I/O counters**: resets all I/O counters like pulse count values.
- **Device Mode (LIOB-18x/48x)**: if switched to "LONMARK Device", the LIOB-18x/48x device is configured to behave like a LONMARK® certified LIOB-15x/45x device, see Section 5.2.

To Set the IP Address on the LCD Display

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

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



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

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

To Configure the BACnet Device ID over the LCD Display

1. On the LCD main screen navigate to **Device Settings »»**.

2. Then navigate to the menu **BACnet »»**.

3. In that menu navigate to the **ID** input for entering the device ID. The field is split into two controls, one for the thousands and one for singles, to simplify entering big numbers.

4. After the device ID has been entered the device name is automatically assembled using that device ID, if no other name has been configured on the Web UI.

5. To let the changes take effect, the device needs to be rebooted. For doing this now you may select the menu item **Save and reboot**.

3.3 Getting Started with the L-INX Configurator

Before setting up a working IEC61131 program, the data points of the L-IOB device need to be set up. These can be data points of I/Os, network variables, registers, etc. Before executing the steps below, install the L-INX Configurator Software from the 'setup.exe'. This file can be downloaded from www.loytec.com.

To Start a Configurator Project

1. Start the L-INX Configurator software by selecting Windows **Start → Programs → LOYTEC LINX Configurator → LOYTEC LINX Configurator**. The application starts up and displays the data point manager screen as shown in Figure 8.

2. When the device is online, connect to the device by clicking on the LNS or device connect speed button as indicated by the red rectangle in Figure 8.

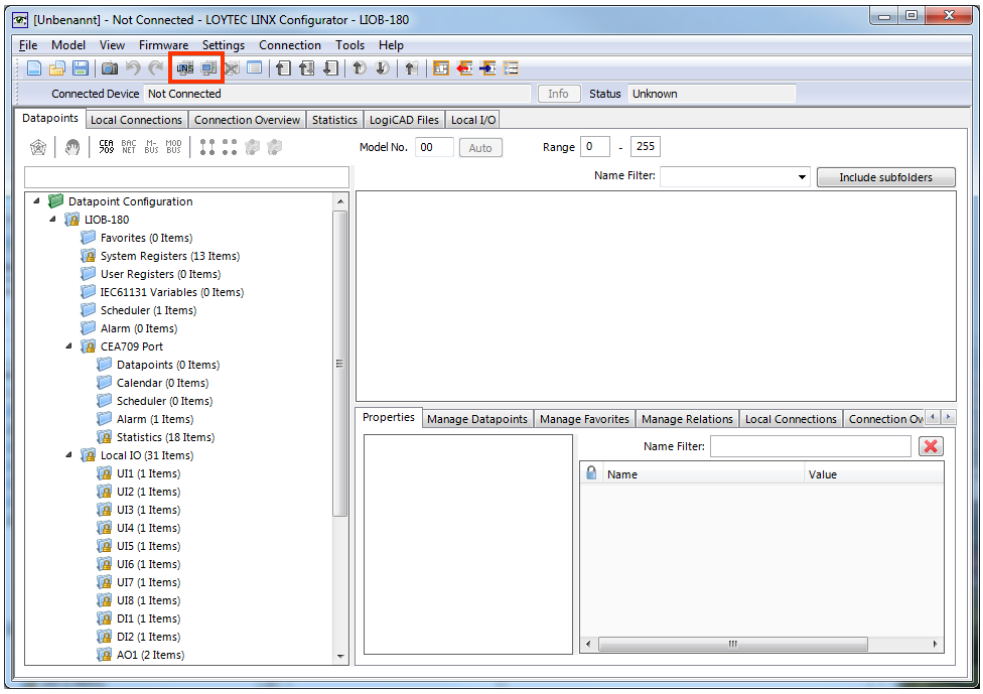


Figure 8: L-INX Configurator main screen

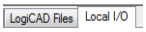
3. For detailed information on how to create data points etc. please refer to LINX Configurator User Manual [2].

3.4 Configuration of the I/Os

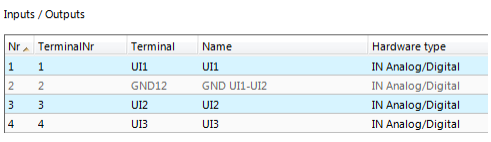
The L-INX Configurator uses a separate tab to configure the I/Os. The I/O configuration can be done off-line and is shown in the following steps.

To Configure I/Os

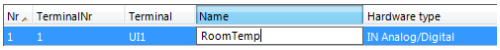
1. For LIOB-18x models, select the **Local I/O** tab. For LIOB-48x/58x models, select the **L-IOB** tab and then **LIOB-LOCAL**.



2. The I/Os available on that L-IOB device are displayed in the **Inputs / Outputs** table.



3. To adapt the I/O name, double-click the name in the **Name** column and edit it, e.g., 'RoomTemp'.



4. Select (or multi-select) an I/O in the **Inputs/Outputs** list and observe the **Object parameters** list below. These parameters can be used to configure the I/O.

Object parameters

Nr	DP Create	OPC	PLC In	PLC Out	Parameter name	Parameter value	Unit	Range	Description
0	☐	☐	☐	☐	Name	RoomTemp			Terminal name
1	☐	☐	☐	☐	HardwareType	IN Analog/Digital			Terminal type
2	☐	☐	☐	☐	SignalType	Voltage 0-10V			Type of the input/output signal

5. On the **Datapoints** tab, the data points for the I/Os have been created. These data points can be used, e.g., in the logiCAD IEC61131 program. For inputs the data point L1_x_UIy_**Input** will be used to read an input value and for outputs the data point L1_x_DOy_**Output** will be used to set an output value.

3.5 Getting started with logiCAD

For developing IEC61131 programs with logiCAD the following components must be installed:

- L-logiCAD setup package. This package installs the logiCAD software, which is needed to design PLC programs for the device.
- L-INX Configurator. This software is required to configure the device to provide the necessary data points to the PLC and integrate the device into the network.
- logiCAD license for using logiCAD on the PC. The license is available as a softlock version or as a hardlock version with a USB dongle. On virtual machines it is mandatory to use the hardlock license.

A detailed guide on how to install the software components described above and upgrade of an older license can be found in Section 6.2.

To Start a logiCAD Project

1. After installing the necessary software components start logiCAD from the L-INX Configurator by clicking the **Start LogiCAD** speed button.

2. The project wizard starts automatically as shown in Figure 9.

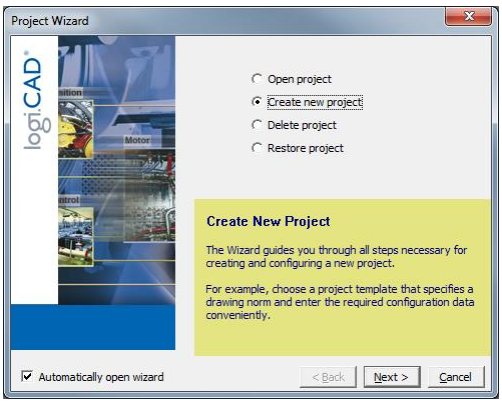


Figure 9: logiCAD project wizard

3. Select **Create new project** and press **Next**.

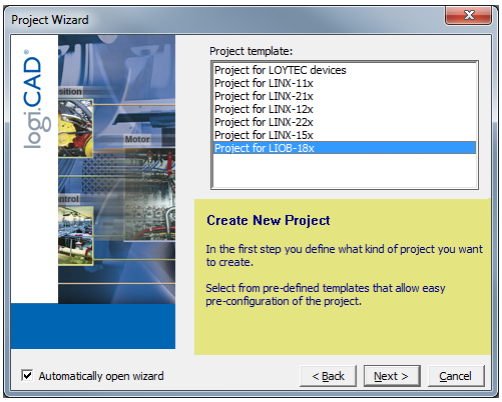


Figure 10: Available project templates

4. Select the project template for the L-IOB device (e.g. LIOB-18x).

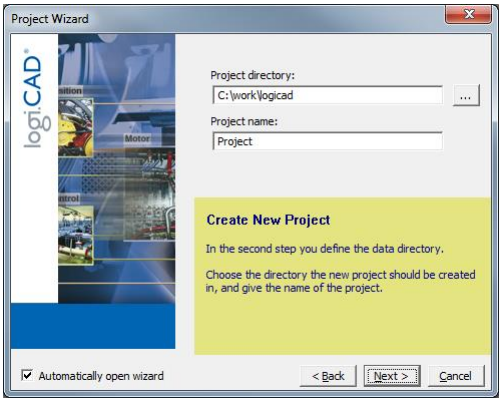


Figure 11: Project name and path

5. Specify the name of the project and the path where to store the project files, see Figure 11.

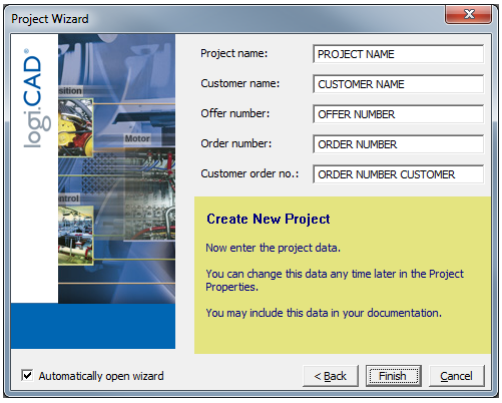


Figure 12: Additional information

6. After specifying additional information the new project is created by pressing the **Finish** button.

7. As shown in Figure 13 below, expand the tree element **Functionplans** and double click **Plan_1** in order to start editing the plan.

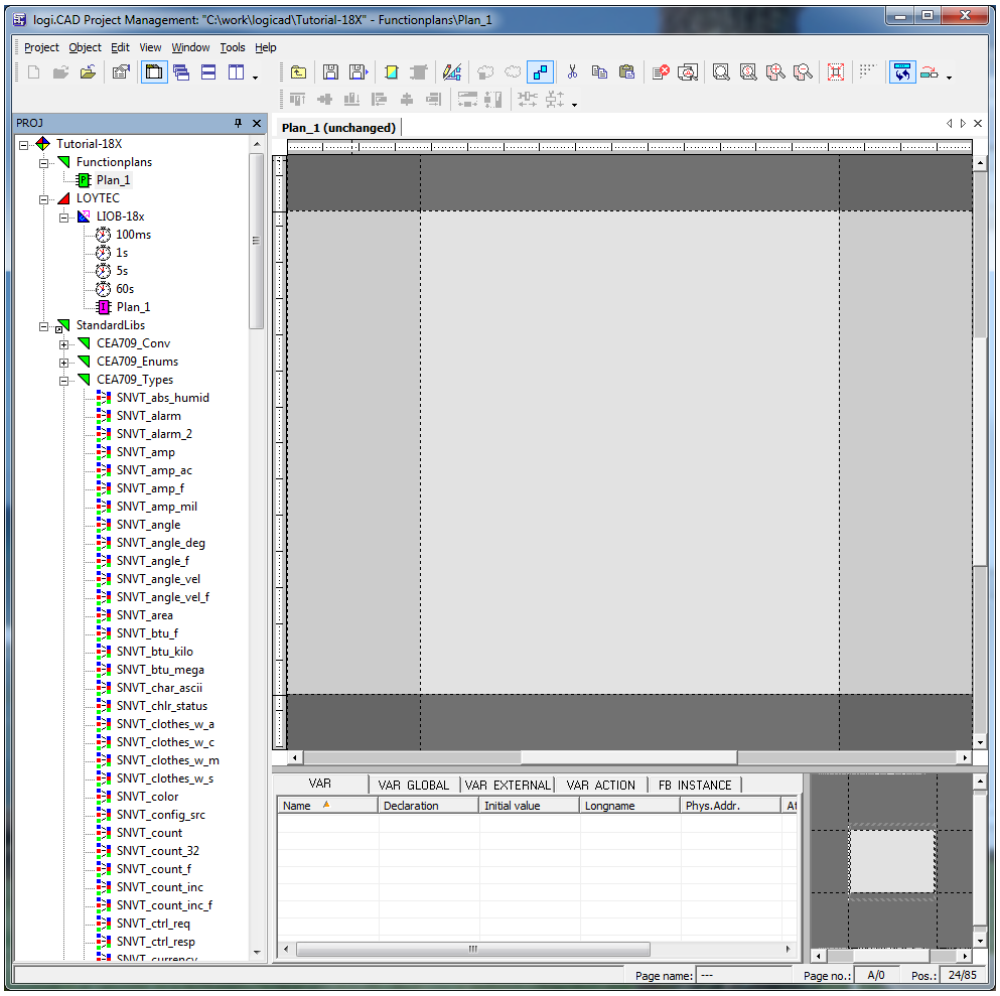


Figure 13: Edit Plan_1

8. In the new LogiCAD project there are no external variables yet. To expose data points from the L-IOB device to the logic program, activate the **PLC** check box of the corresponding data points in the Configurator, e.g., an I/O data point, a network variable, a BACnet server object, or a user register.

9. When you have completed selecting the PLC data points, click the **Export variables to LogiCAD** speed button while LogiCAD is running.

10. The data points now appear as variables in LogiCAD in a folder under the device folder. The folder is named specific to the technology of the data points, e.g. 'Local IO' for all local I/Os that are exposed as a PLC variable. An example is shown in Figure 14.

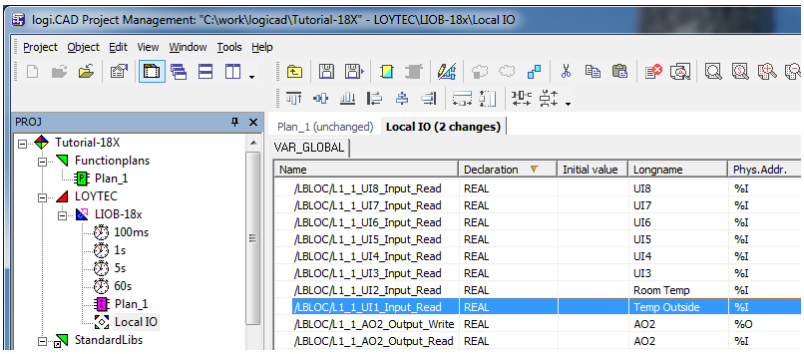


Figure 14: Exposed PLC data points appear in LogiCAD

11. Now the logic can be developed on the function plan.

12. For later debugging, it is good practice to add online test fields to the drawing, to display the current value of the signals during online test. To do this, right-click on the value output of the left function block and select **Create OLT Field** from the context menu, as shown in Figure 15.

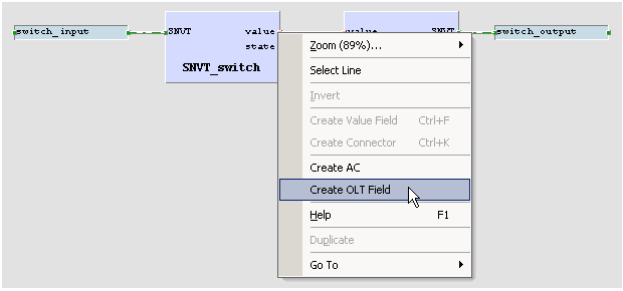


Figure 15: Create online test fields

13. Place the fields above and below the drawing as shown in Figure 16, then press the **Save** button to save your changes.

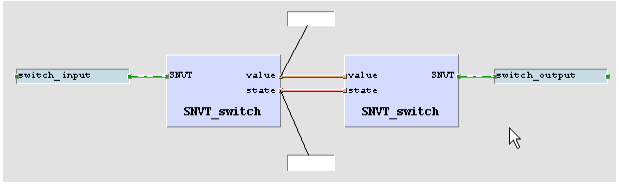


Figure 16: Online Test fields

14. Finally, open the context menu of the **LIOB-18x** (48x/58x) device again and select **Code Generation**. In the dialog, press the button **Start** to start the code generation process. On success, the code generation window reports Errors=0 and Warnings=1.

15. Close the window by pressing the **OK** button. Now the compiled IEC61131 program can be downloaded to the device. Right-click the tree element **LIOB-18x** and select **Download** from the context menu. A connection dialog will appear and ask for the type of connection and additional information.

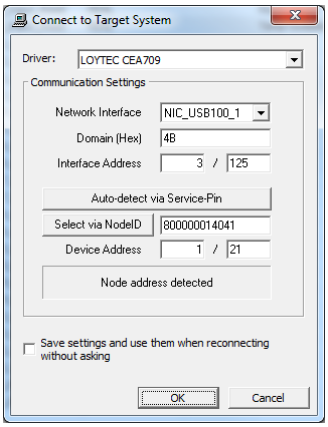


Figure 17: IEC61131 program download

16. Select the LOYTEC CEA709 communication driver and enter the requested communication settings as described in Section 6.4.2. Start the download process by pressing the **OK** button.

17. After the download completed, check the PLC Status in the LCD display. If it is not "Running", the logic is not yet running. Reboot the device to start the IEC61131 program.

4 Hardware Installation

4.1 Enclosure

The enclosure of the product and its terminal layout are shown on the installation sheet found in the product's box.

4.2 Product Label

The product label on the side of the L-IOB I/O controller contains the following information (see Figure 18):

- L-IOB order number (e.g.: LIOB-180),
- Date Code, which defines the production week and year,
- Serial number with bar-code (SER#),
- Node ID of the device. Models with an Ethernet/IP interface additionally contain the MAC address on the label.

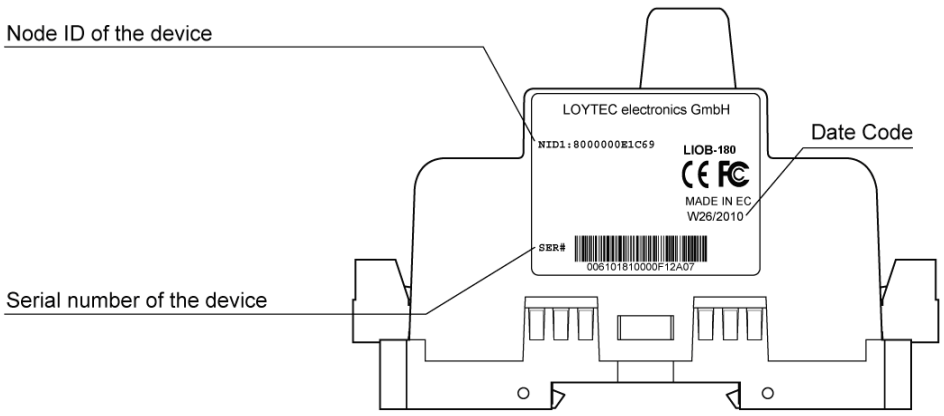


Figure 18: L-IOB product label

Unless stated otherwise, all bar codes are encoded using "Code 128". An additional label is also supplied with the L-IOB for documentation purposes. The specific contents of the product label are shown on the installation sheet found in the product's box.

4.3 Mounting

All L-IOB devices come prepared for mounting on DIN rails following DIN EN 50 022. The devices can be mounted in any position. However, an installation place with proper airflow must be selected to ensure that the L-IOB's temperature does not exceed the specified range (see Section 11.3).

4.4 Power Supply and Wiring

There are four ways of connecting LIOB-18x/48x/58x devices along with all other nodes in a network:

- CEA-709 Free Topology (LIOB-18x),
- CEA-709 Bus Topology (LIOB-18x),
- CEA-852 (LIOB-48x),
- BACnet/IP (LIOB-58x).

All rules of CEA-709 / CEA-852 / LONMARK® devices or BACnet devices regarding network installation, management, and maintenance apply. For using external (non-LOYTEC) power supplies, please refer to Section 9.1. For connecting sensors and actuators to the L-IOB I/Os, please refer to Sections 9.2 and 9.3. The next sections describe the different power supply and cabling variants in detail.

4.4.1 CEA-709 Network Connection in Free Topology

Figure 19 shows the connection of LIOB-18x controllers in free topology, which can be used for cable lengths of up to 500 m between any two devices. As an example for other nodes in the network, a L-INX device is connected on top.

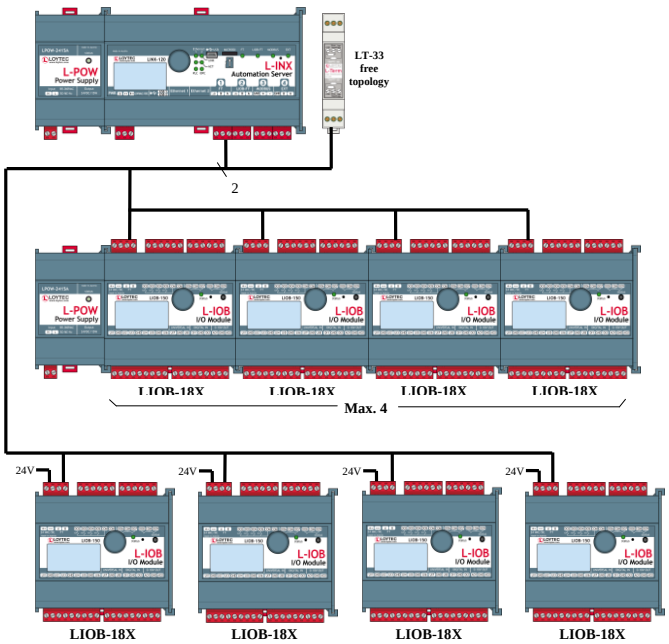


Figure 19: LIOB-18x in Free Topology

The L-IOB devices can either be powered by L-POWs (middle part of Figure 19) or other 24 V power supplies (bottom part of Figure 19). One LT-33 terminator (free topology terminals) must be placed somewhere in the network.

4.4.2 CEA-709 Network Connection in Bus Topology

Figure 20 shows the connection of LIOB-18x controllers in bus topology, which must be used for a cable length of above 500 m. As an example for other nodes in the network, a L-INX device is connected on the top end of the bus. The L-IOB devices can either be powered by L-POWs (middle part of Figure 20) or other 24 V power supplies (bottom part of Figure 20). An LT-33 terminator (bus topology terminals) must be placed at both ends of the bus.

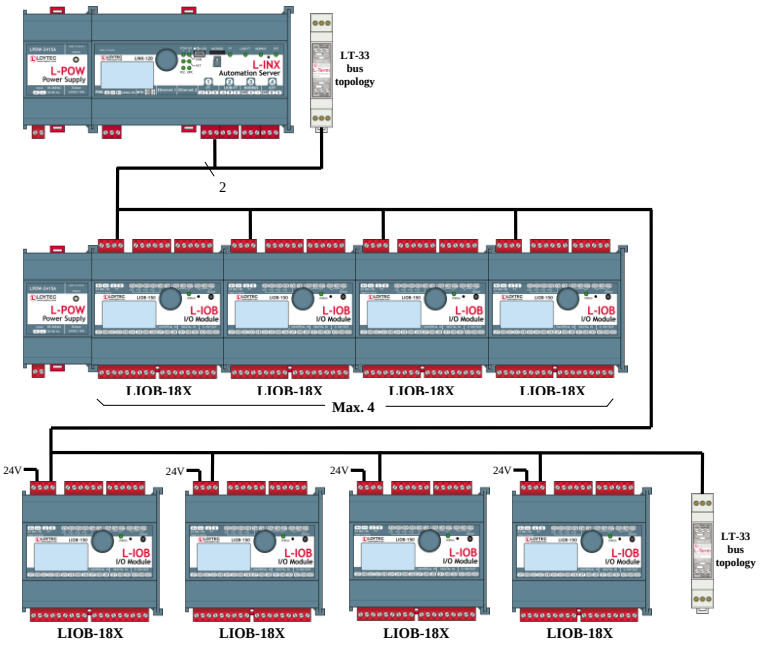


Figure 20: LIOB-18x in Bus Topology

4.4.3 CEA-852 Network Connection

Figure 21 shows the connection of LIOB-48x controllers over Ethernet/IP. The L-IOB devices can either be powered by L-POWs (middle part of Figure 21) or other 24 V power supplies (bottom part of Figure 21). The L-INX device at the top is an example of a CEA-852 configuration server, which is required in every CEA-852 network.

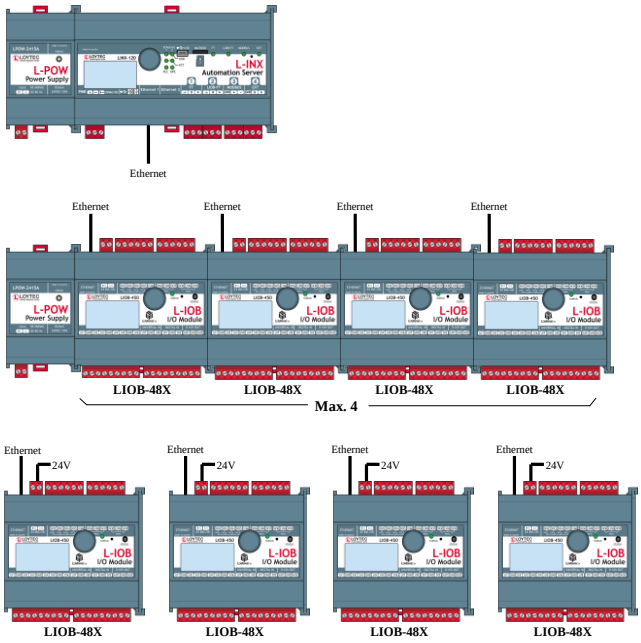


Figure 21: LIOB-48x Connection

4.4.4 BACnet/IP Network Connection

Figure 22 shows the connection of LIOB-58x controllers over Ethernet/IP. The L-IOB devices can either be powered by L-POWs (top part of Figure 22) or other 24 V power supplies (bottom part of Figure 22).

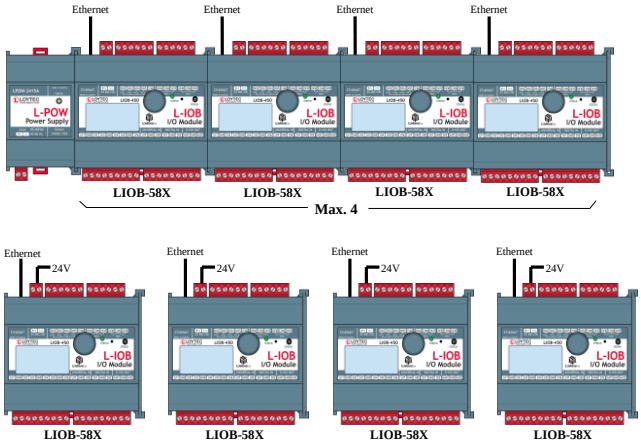


Figure 22: LIOB-58x Connection

4.4.5 Extension of LIOB-48x/58x controllers with LIOB-45x/55x device

The I/Os of LIOB-48x and LIOB-58x controllers can be extended with one LIOB-45x or LIOB-55x device (in L-INX mode) by using the LIOB-IP bus of the controller. Please refer to the LIOB-10x/x5x User Manual [4] for how to connect a LIOB-45x/55x device to a L-IOB host (in this case the LIOB-48x/58x controller).

4.5 LED signals

The L-IOB device is equipped with a three-color status LED which indicates the current state of the L-IOB device. Available LEDs and their location on the respective device

model can be found on the product's installation sheet. The installation sheet can be found in the product's box.

4.5.1 Status LED of LIOB-18x

The meaning of the LED signals for the LIOB-18x models is listed in Table 1.

Behavior	Description	Comment
OFF	Online	The L-IOB device is online.
Flickering GREEN	Traffic	The L-IOB device receives or transmits packets.
ORANGE	Manual Mode	At least one I/O is in manual mode.
RED	Error	An error has occurred (e.g. a sensor is disconnected or signals an error).
GREEN flashing at 0.5 Hz	Offline	The L-IOB device is offline.
RED flashing at 0.5 Hz and "LIOB Fallback" shown in LCD UI	Fallback Override	The primary firmware image is corrupt and the L-IOB has booted the fallback image. In this case, the firmware must be upgraded again.

Table 1: Status LED Patterns of LIOB-18x

4.5.2 Status LED of LIOB-48x

The meaning of the LED signals for the LIOB-48x models is listed in Table 2.

Behavior	Description	Comment
GREEN	Online	The L-IOB device is online.
Flickering GREEN	Traffic	The L-IOB device receives or transmits packets.
ORANGE	Manual Mode or no CS	At least one I/O is in manual mode or the configuration server cannot be contacted.
RED	Error	An error has occurred (e.g. a sensor is disconnected or the configuration server has rejected the device).
GREEN flashing at 0.5 Hz	Offline	The L-IOB device is offline.
ORANGE flashing at 0.5 Hz	Offline, no CS	The L-IOB device is offline and the configuration server cannot be contacted.
RED flashing at 0.5 Hz	No CEA-852 Config	The CEA-852 port is not configured. The device must be added to a CEA-852 IP channel.
RED flashing at 0.5 Hz and "LIOB Fallback" shown in LCD UI	Fallback Override	The primary firmware image is corrupt and the L-IOB has booted the fallback image. In this case, the firmware must be upgraded again.

Table 2: Status LED Patterns of LIOB-48x

4.5.3 Status LED of LIOB-58x

The meaning of the LED signals for the LIOB-58x models is listed in Table 3.

Behavior	Description	Comment
OFF	No Traffic	No packets are received or transmitted.
Flickering GREEN	Traffic	The L-IOB device receives or transmits packets.
ORANGE	Manual Mode	At least one I/O is in manual mode.
RED	Error	An error has occurred (e.g. a sensor is disconnected).
RED flashing at 0.5 Hz and "LIOB Fallback" shown in LCD UI	Fallback Override	The primary firmware image is corrupt and the L-IOB has booted the fallback image. In this case, the firmware must be upgraded again.

Table 3: Status LED Patterns of LIOB-58x

4.6 Status Button

The L-IOB device is equipped with a status button (see product installation sheet). When pressing the status button shortly during normal operation of the device, a service pin message (LIOB-18x/48x) or I-Am message (LIOB-58x) is sent out, the LCD is reset, and the LCD backlight is switched on.

The status button can also be used to switch the device back to factory default state. Press the button and power-cycle the device. Keep the button pressed until the Status LED illuminates orange permanently. Release the button within five seconds from that time on to reset the device to factory defaults.

5 Concepts

This chapter describes the basic concepts behind the installation, configuration, and data points of LIOB-18x/48x/58x devices. Observe that not all properties and data points will be available in all configuration instances. Some properties e.g. can only be seen or setup at configuration time, others only at run time.

5.1 Device Installation

For LIOB-58x devices, the first step after hardware installation and IP configuration is to configure the BACnet interface. For LIOB-48x devices, the first step after hardware installation and IP configuration is to add them to a CEA-852 channel. For detailed configuration steps refer to the LOYTEC Device User Manual [1]..

The I/Os of LIOB-48x and LIOB-58x controllers can be extended with one LIOB-45x or LIOB-55x device (in L-INX mode) by using the LIOB-IP bus of the controller. Please refer to the LIOB-10x/x5x User Manual [4] for how to connect a LIOB-45x/55x device to a L-IOB host (in this case the LIOB-48x/58x controller).

Both the LIOB-18x and LIOB-48x devices must be installed and commissioned like any other CEA-709 / LONMARK® node. Please consult the documentation of your CEA-709 network management tool on how to perform these tasks.

For LNS™ based network management tools, the LOYTEC L-INX Configurator Software acts as an LNS™ plug-in to configure the LIOB-18x/48x devices. It also installs the needed template for the LIOB-18x/48x models when it is registered as a plug-in in the network management software. Both off- and online installation of the L-IOB devices is supported.

For Non- LNS™ network management tools, a LOYTEC NIC such as the NIC-USB100 or NIC852 is needed for configuration of LIOB-18x/48x devices. The devices must be configured (using the CEA-709 connection method of the Configurator Software) before installing and commissioning them in the network management tool.

5.2 LONMARK® Device Mode (LIOB-18x/48x)

LIOB-18x/48x I/O Controllers in a certain configuration are LONMARK® certified. To setup this special configuration, the LONMARK® device mode must be setup in the LCD UI. From the main page choose **Device Settings »» Device Management »» Device Mode** menu and select the mode **LONMARK Device**. The LIOB-18x/48x device will reboot and from now on behave exactly like the corresponding LIOB-15x/45x I/O module in LONMARK® device mode. If e.g. a LIOB-180 device is switched to LONMARK® device mode, it will behave like a LIOB-150 I/O module (in LONMARK® device mode). To switch the device back, go to the device information and configuration page (toothed wheel icon)

in the L-IOB LCD UI and switch the device mode back to 'I/O Controller'. Please refer to the LIOB-10x/x5x User Manual [4] for more information about the LIOB I/O modules in LONMARK® device mode.

5.3 Data Point Configuration

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.

5.4 IEC61131 Variables

IEC61131 variables are used to exchange data with the IEC61131 program. These variables are represented in the data point configuration as register data points and can be connected to other data points, e.g. to CEA-709 NV points, via data point connections.

In contrast to CEA-709 variables, IEC61131 variables are always represented as single data point. In case of scalar values (representing CEA-709 scalar or enumeration types) one of the following basic data types might be used:

- **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.

Structured IEC61131 variables, representing for example structured NVs, or customer defined IEC61131 structures, are stored as user type:

- **User**: A *user* data point 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 entire data structure. User data points can only be connected to other user data points of the same data length.

6 IEC 61131

To design the IEC61131 program which should run on the device, the graphical programming tool logiCAD is required. The tool allows creating IEC61131 programs using various IEC61131 programming languages. It offers additional features downloading and debugging of the created program.

In addition to logiCAD, the L-INX Configurator is necessary to create an appropriate data point configuration in the automation server. The usage of logiCAD itself is beyond the scope of this manual. Please refer to the logiCAD online help in case of additional questions.

6.1 Overview

The PLC in the device is intended to perform IEC61131 programs operating on IEC61131 data points. The operating principle is to connect IEC61131 data points to data points derived from CEA-709, BACnet, or L-IOB IOs. Figure 23 depicts the usage of data points for IEC61131 programs at the example of using CEA-709 network variables.

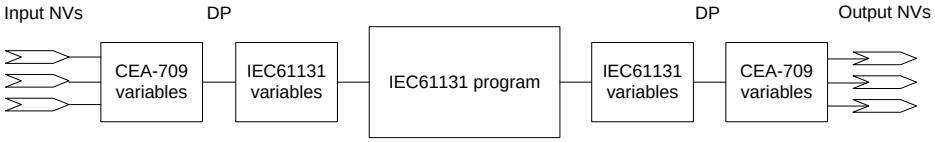


Figure 23: Usage of data points in IEC61131

Alternatively, data points can also be directly exposed as IEC61131 variables. The data points provide a PLC check box for this purpose. In this use case, not special IEC61131 data points are created and no connections are necessary.

6.2 Installing logiCAD

For developing IEC61131 programs with logiCAD the following components must be installed:

1. L-logiCAD setup package. This package installs the logiCAD software, which is needed to design PLC programs for the device.

2. L-INX Configurator. This software is required to configure the device to provide the necessary data points to the PLC and integrate the device into the network.

3. logiCAD license for using logiCAD on the PC. The license is available as a softlock version or as a hardlock version with a USB dongle. On virtual machines it is

mandatory to use the hardlock license. How to obtain and install the license is described in this Section.

The L-logiCAD installer installs the IEC61131 programming environment logiCAD and all related software packages. These packages include the template project for the device, the required software to build IEC61131 programs for the device, and required extensions to interface to CEA-709 networks. Follow the instructions of the installer to install logiCAD.

The language for the logiCAD user interfaces can be set to German or English using the administration folder of the logiCAD control center. The logiCAD control center can be started from the Windows start menu.

6.2.1 Softlock License

The license to run a copy of logiCAD on any PC is based on a softlock license. In case the original softlock license file was generated for an older L-logiCAD version, the software will request an update signature key for the new L-logiCAD installation. This signature key can be found in the file 'logicad_Readme.txt' that is located in the installation directory of the L-logiCAD software.

To obtain and install a new license, start logiCAD. The product activation dialog will appear as shown in Figure 24. In this dialog press the button **Download Softlock License.**

Important!	***When using Windows 7 or Windows 8, start logiCAD as administrator in order to allow the script read the computer number code. If there are x's in the code, the code was not read by the script.***

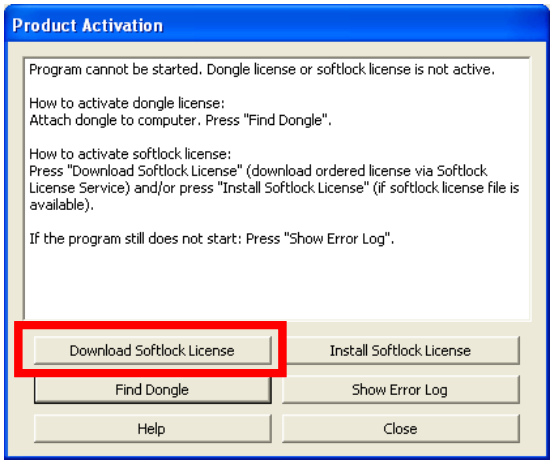


Figure 24: logiCAD Product Activation

Enter the license data provided with the product information on the Web site form as shown in Figure 25. Fill in the **SL-number** and **Licensing Code** from your L-LOGICAD registration form and click on **Download License File**. Save the license file on your computer or have it e-mailed to you.

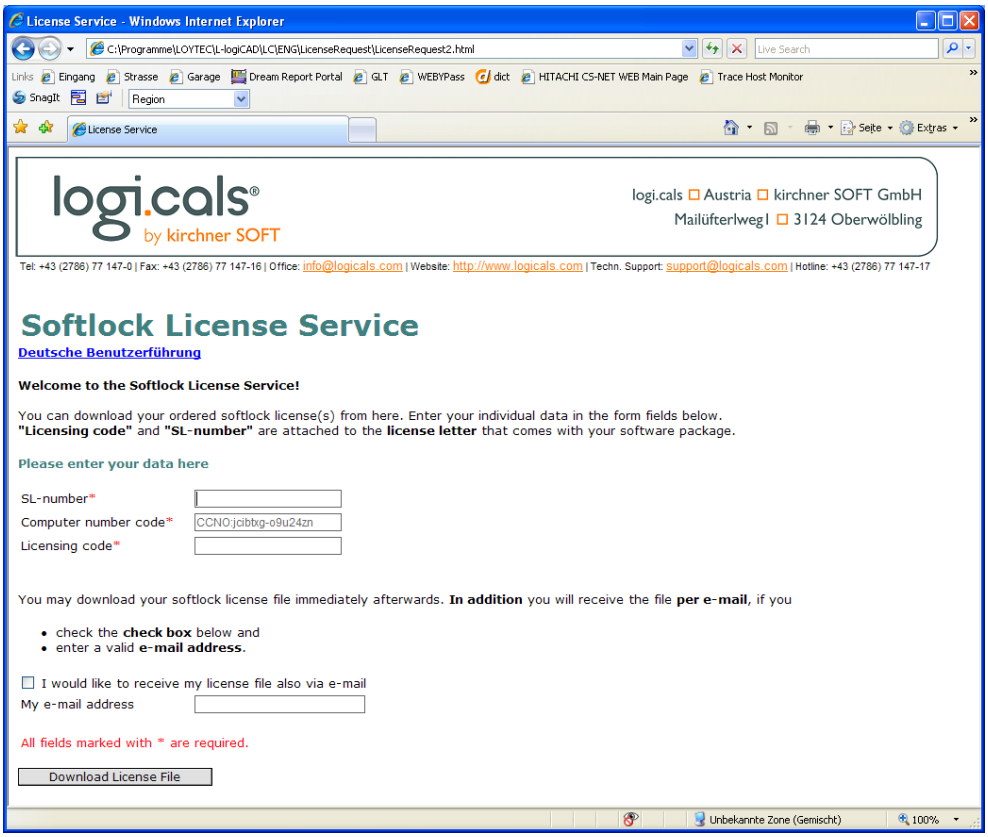


Figure 25: logiCAD softlock license Web form

Then install the license using the **Install Softlock License** button in the product activation dialog as indicated in Figure 26.

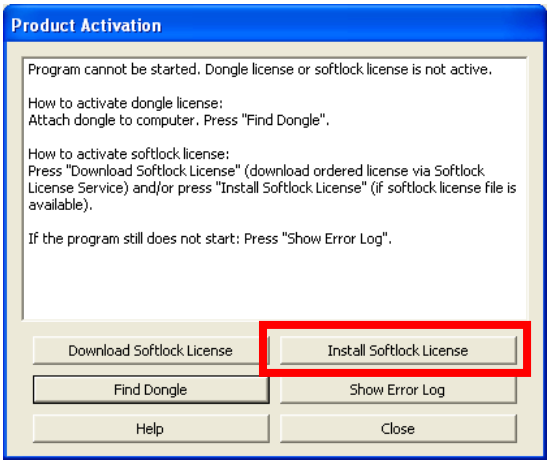


Figure 26: logiCAD softlock installation

A file requestor dialog opens. Locate the downloaded license file (logicad.lf) and click **Open**. Finally click **Close** in the product activation dialog and start logiCAD again. The softlock license is now activated.

6.2.2 Hardlock License

The hardlock license is needed to run a copy of logiCAD in a virtual machine on the PC. It needs to be purchased separately as L-LODICAD-USB and is distributed as a USB hardlock key of type ‘CodeMeter’. If no driver for this type of hardlock is already installed

on the PC, install the hardlock driver provided on the LOYTEC website. It can be found in Support → Download and the product selection for L-LOGICAD. After the driver installation has completed, plug in the USB key. It must be recognized as a mass storage device (the details on it show the location 'CodeMeter-Stick'). When starting logiCAD it recognizes the hardlock automatically.

If the driver installation failed, or the USB key was not detected or connected, the window shown in Figure 27 will be shown when logiCAD is started. In this case plug in the hardlock stick, wait for Windows to recognize it and then click **Find Dongle**.

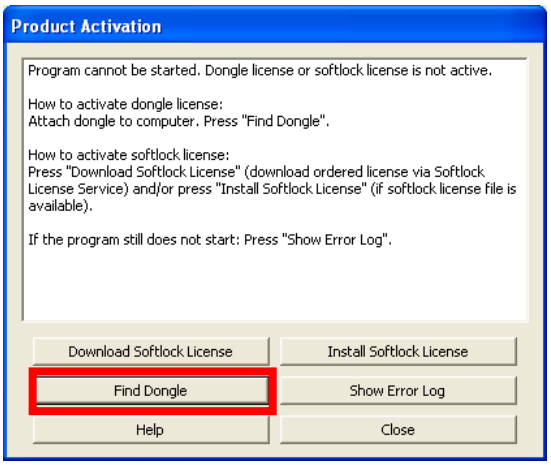


Figure 27: logiCAD Product Activation with Hardlock

Note:	*logiCAD checks frequently if the USB key is present. When unplugging the USB key, even if logiCAD was successfully started, all major features are automatically disabled. But there is no additional message for the user!*

6.3 IEC61131 Project Files

In the L-INX Configurator select the **LogiCAD Files** tab to attach an IEC61131 program and a logiCAD project to the project. The tab is shown in Figure 28.

If there is an IEC61131 program attached to the project, every time a new configuration is downloaded to the device, the L-INX Configurator asks, if the attached program should be downloaded as well. This way, no logiCAD is required to download a suitable IEC61131 program in a separate step. The project designed in the L-INX Configurator can hold all necessary information to set up a running device:

- IEC61131 data point configuration,
- Data point configuration in the device,
- Required connections,
- IEC61131 program.

The logiCAD project directory can also be attached to the configuration file, in order to include the logiCAD project sources from which the program was compiled.

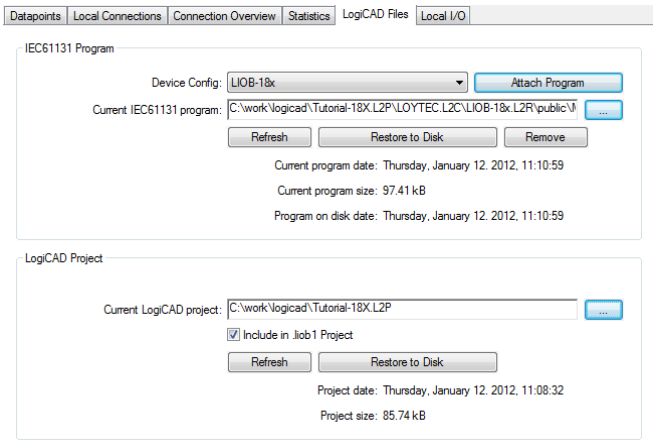


Figure 28: IEC61131 Project Files

To attach an IEC61131 program or a logiCAD project select the file/folder to attach. The selected data will be attached automatically the project. It can be restored to disk by pressing the **Restore to Disk** button. During the development process the attached data may change several times. To update the attached data, located on the before provided path, press the **Refresh** button.

Every time a logiCAD project is successfully compiled, the file MBRTCode.so, the compiled IEC61131 program, is copied to the *public* directory of the device resource for which the program was compiled. Select this file to attach it to the project of the Configurator. Note that the time and date of the file indicates the time of the last code generation. If logiCAD is not able to build a new program, the old file will not be deleted.

The project will be scanned for device resources and the available devices will be listed in the dropdown box called **Device Config**. Select the desired device and press the button **Attach Program** to automatically attach the correct MBRTCode.so file.

6.4 Working with logiCAD

For using the device with logiCAD, a predefined project template for the LIOB-18x/48x/58x must be used. Hence, when creating a new project e.g. for a LIOB-180, select the project template “Project for LIOB-18x”, see Figure 10. For additional information how to create, delete and manage projects please refer to the logiCAD online help.

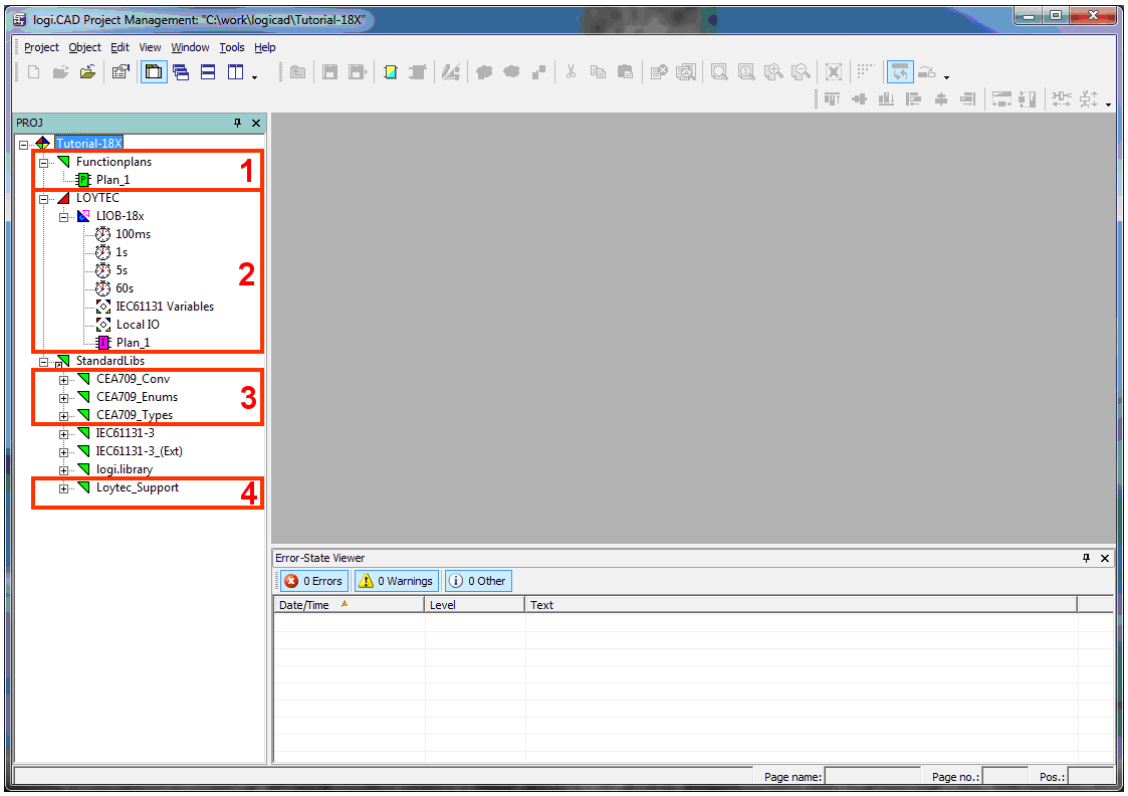


Figure 29: LOYTEC specific extensions

Figure 29 shows the standard project for the LIOB-18x/48x/58x including all LOYTEC-specific extensions for logiCAD. It shows the structure window showing the project structure on the left, an empty working area top right and the **Error-state viewer** on the bottom right side.

The structure window offers interfaces to the following features:

1. The folder **Functionplans** holds all the program types created within logiCAD. 'Plan_1' is the default plan to start with.

2. The folder **LOYTEC/LIOB-18x** represents the device. The folder **LOYTEC** represents a configuration containing one LIOB-18x/48x/58x resource; please refer to the logiCAD online help for details. To run a program, located in the folder **Functionplans**, a program instance of the corresponding function plan must be created. In the standard LIOB-18x/48x/58x template, a program instance of the 'Plan_1' is already defined. To be able to transfer IEC61131 variables from the device to the IEC61131 program a global variables object within the **LIOB-18x**, **LIOB-48x**, or **LIOB-58x** folder is required, see Section 6.4.1 for details.

3. LogiCAD operates on variable types standardized in the IEC61131 standard. Look for "Elementary and Generic Data Types" within the logiCAD online help to get information about the available data types. For those devices, which are intended to operate on structured NVs, appropriate type definitions for the NVs are required. These definitions are inside the folder **CEA-709_Types**. Additional NVs must be converted to data types that can be processed by logiCAD. Therefore *Technology Converters* are supplied with the project, which perform this conversion. See Section 6.4.3 for details.

4. For designing programs that support the force update functionality (see Section 6.6.1), or designing user-defined *Technology Mapper* (see Section 6.6.2), additional function blocks are required. These blocks are located within the **Loytec_Support** directory.

All LOYTEC-specific add-ons are provided using function blocks. Hence, in the following, all samples are based on designs using function blocks.

6.4.1 Managing Variables

On a function plan, three basic types of variables may be created using the tabs shown below the function plan sheet:

- **VAR:** Variables created on this tab will be visible only to the logic designed on this group of sheets. It will not be accessible to any other programs or to any function blocks which are used in this program. It is similar to a 'static' variable declaration in a C code function.

- **VAR GLOBAL:** Variables declared here will be accessible to the entire program, including any function blocks which are called by the program. Function blocks which need to reference this variable need to have a suitable declaration of an external variable (see next point). This declaration is similar to a 'static' declaration of a C variable outside a function, which will be visible to all functions inside the C code module, but not visible to other modules.

- **VAR EXTERNAL:** Variables declared in this list will be treated as open references to a global variable which exists somewhere in the scope of the device on which the program will be executed. This means that a global variable needs to be declared on the device resource, which will be available to all programs running on the device. If the physical address parameter of the variable is set to %I, %O, or %M, the variable will be handed down to the I/O driver of the device for processing. If a suitable IEC61131 variable exists in the data point configuration of the device, its value will be forwarded by the I/O driver between the PLC variable and the data point. If no physical address is set, the variable will only be visible to the PLC programs but not to the I/O driver, which may be used to exchange data between PLC tasks.

The basic data flow between the CEA709 network (or other technologies) and the PLC program is depicted in Figure 30.

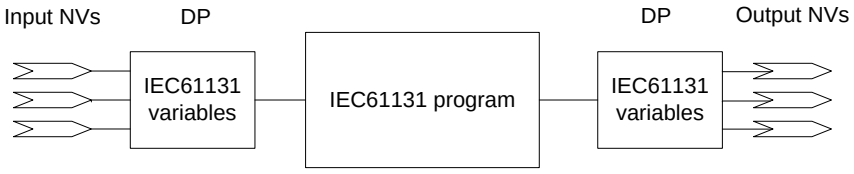


Figure 30: Connecting IEC61131 variables

The place where global variables are created on the device is shown in Figure 31.

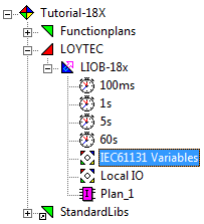


Figure 31: Global Variables Object

When starting a new project, no global variables object is available; it must be created before compiling the IEC61131 application. The global variables object is created automatically or manually by the LINX Configurator when exporting variables to logiCAD.

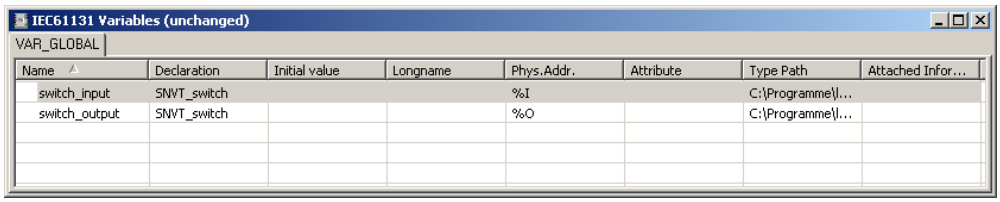


Figure 32: Variables defined in global variables object

Figure 32 shows the contents of a sample global variables object. As shown a global variable is defined by the fields **name**, **declaration** and **phys.addr.**:

- **Name:** The name of a global variable must be unique. The name is used from the IO driver to identify the global variable and from the L-INX Configurator to generate the corresponding IEC61131 data points.
- **Declaration:** Here the type of the global variable is defined.
- **Phys.Addr.:** The IO driver needs to know the data flow direction to be able to update variables. The direction is defined by adding %I for an input variable, %O for an output variable and %M for a marker (input and output). If the address is empty, the I/O driver will not handle the variable, but it may still be used by the tasks running on the device.

Important: ***Only ASCII characters can be used for naming the global variables.***

6.4.2 Build and Download the IEC61131 Program

IEC61131 programs, designed using logiCAD, must be cross compiled in order to run on the device. The prerequisite to compile an IEC61131 program are as follows:

- A program instance with associated program type
- A corresponding global variables object

Right click on the LIOB-18x/48x/58x resource, see Figure 31, and select Code Generation. Please refer to the logiCAD online help for the meaning of the options. Take care about the option breakpoint support, see section 6.4.5 for details.

After successfully finished code generation, the IEC61131 application can be downloaded to the device. Right click on the LIOB-18x resource and select download.

The IEC61131 program is downloaded to a LIOB-18x device via CEA-709:

- **CEA-709**: Select the network interface to use and fill out all other fields, see Figure 33. Alternatively, select the network interface and press Auto-detect via Service-Pin. Then press the service pin on the device. Note that this connection method requires an installed LOYTEC network interface.

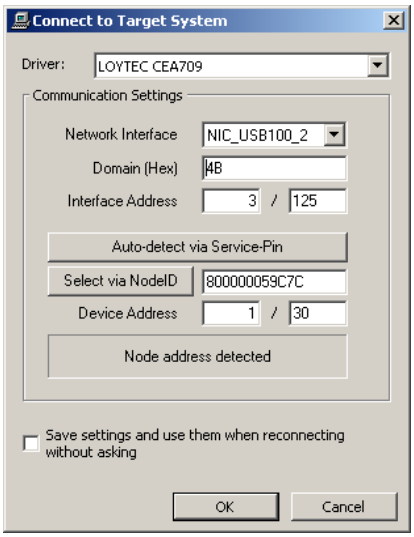


Figure 33: Connect via CEA-709

The IEC61131 program is downloaded to a LIOB-48x device via TCP/IP or CEA-852:

- **TCP/IP**: Enter the IP address of the LIOB-48x device. Do not change the default communication port (2048). This is the easiest and fastest way to connect to the device.
- **CEA-852 (CEA-709 over IP)**: If for some reason, downloading directly via TCP/IP is not possible (e.g. because the required communication port is not accessible over the network), the download can also be done over CEA-852. Select CEA-709 as described above and choose the NIC852 which is a member of the same CEA-852 channel as the LIOB-48x device.

Important:	***To be able to communicate with the device via CEA-709 or CEA-852, the device must be commissioned.***

The IEC61131 program is downloaded to a LIOB-58x device via TCP/IP:

- **TCP/IP**: Enter the IP address of the LIOB-58x device. Do not change the default communication port (2048).

6.4.3 Usage of NVs, Technology Converters (LIOB-18x/48x)

To use CEA-709 variables, the content of the NVs must be converted to IEC61131 compliant data types. Look for “Elementary and Generic Data Types” within the logiCAD online help to get information about the available data types. Technology Converters are used to perform the transformation from CEA-709 data types to IEC61131 data types. All Technology Converters are summarized in the subfolder CEA709_Conv located in the StandardLibs folder.

Depending on the type of the NV there are three different ways to use the NV within IEC61131 programs:

- Simple NVs that hold only one scalar value, e.g. SNVT_amp:

  Those kinds on NVs are represented as IEC61131 REAL values within logiCAD. There is no additional conversion necessary. Figure 45 shows an example program for scalar data types.

- Simple NVs based on an enumeration, e.g. SNVT_date_day:

The active identifier of the enumeration is represented as Boolean value. When using NVs based on enumerations, Enumeration Converters are used to identify the current state. There are two kinds of Enumeration Converters. First, the Enumeration Converters that convert the enumeration types to Boolean types, grouped in the folder **Convert from CEA709_Enums**. Second, the Enumeration Converters that converts several Boolean inputs to an enumeration type, grouped in the folder **Convert to CEA709_Enums**.

- Structured NVs that consists of a number of fields, e.g. SNVT_switch:

  On structured NVs two tasks must be performed by the Technology Converters. First the structure of the NV is mapped to IEC61131 conform data types. Second, if necessary scaling factors are applied. Similar to the Enumeration Converters, the Technology Converters are split up into two subfolders. The first one, which converts the NV into IEC61131 compliant data types, is located in the folder **From_CEA709_Types**. Technology Converters to set up NVs based on IEC61131 data types are grouped in the folder **To_CEA709_Types**.

Figure 34 shows the three possibilities how to use NVs within an IEC61131 program.

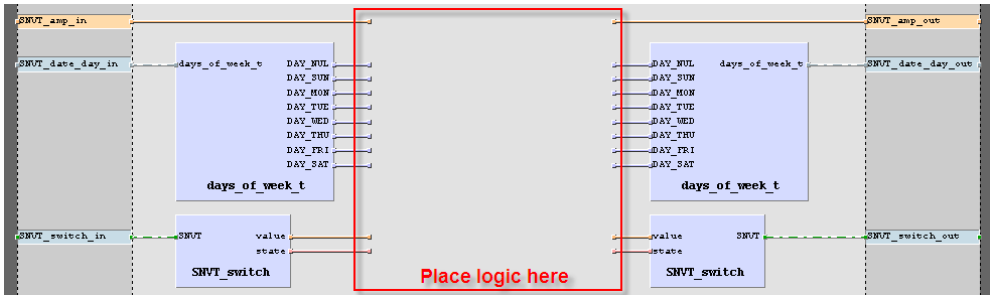


Figure 34: Usage of NVs

If a structured NV comprises enumerations types, these enumerations are not split up by the Technology Converter. To get the value of the enumeration, connect an Enumeration Converter to the corresponding output of the Technology Converter.

For every Technology Converter and Enumeration Converter an online help window, displaying the interface description is available. Select the Technology Converter and press F1 to get the interface description.

6.4.4 IEC61131 Program Cycle Time

IEC61131 programs are performed in a periodical manner. IEC61131 tasks are used do control the execution of an IEC61131 program. As shown in Figure 31 several default tasks are defined within the template project. Right click on the clock symbol and select properties to change the cycle time of the task.

As described in Section 6.4 a program instance is required to execute the IEC61131 program. The cycle time of the IEC61131 program is controlled by the task assigned to the program instance. In order to change the cycle time right click on the program instance and select properties. In the upcoming window the task assignment for the selected program type can be changed, see Figure 35.

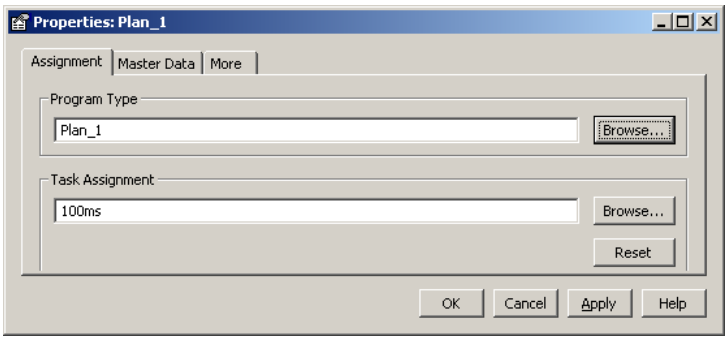


Figure 35: Task assignment

Please take care when defining names for the tasks. The names displayed in the project view are symbolic names, they do not correspond to the settings for the cycle time, even if the template project use the configured cycle time as task name.

6.4.5 CPU Overload

Several conditions affect the CPU utilization of the IEC61131 program. As a result it is not possible to predict the system load caused by the IEC61131 program. E.g. the following parameters are of particular importance when designing IEC61131 programs:

- Number of inputs and outputs handled by the I/O driver.
- Complexity of the logic in the running IEC61131 program.
- Number of simultaneously running program instances on one device.
- Cycle time of IEC61131 programs.
- logiCAD breakpoint support and force-able code enabled or disabled

The developer of the IEC61131 program is able to check the current system load on the LCD UI. In case of CPU overload, the IEC61131 program may not be able to finish its work within the defined cycle time. Adapt the program in order to reduce the total system load below 80%. Here are a few tips to keep the CPU load down:

- Increase the cycle time, so that the task may finish in time before the next cycle start is scheduled. The PLC kernel will always schedule the next run at an absolute time, no matter how long the previous run took, in order to compensate for irregular execution times and keep a steady cycle time if possible.
- Reduce the number of I/O variables, to reduce the load caused by exchanging data between the PLC program and the data points of the automation server.
- Reduce the number of independent tasks and try to place as much functionality as possible into one task. Every running task will call the I/O driver for new inputs and outputs independently, therefore two tasks running at a 1s cycle time each will cause twice the I/O load of one task running at 1s cycle time.
- Take special care about the complexity of function blocks which are used a lot. Bad performance of one such block may dramatically increase CPU load if it needs to be calculated several hundred times in one cycle.
- Try to disable breakpoint support and force-able code when generating code for the target, to get the most efficient PLC code out of your logic.

- For complex designs, it may be possible to add a state machine using SFC elements and enable/disable large parts of the logic based on the current state of operation.

- Whenever a function does not need to calculate new output values under certain conditions, use the built-in EN input of the block to disable execution and thus reduce the required CPU time, instead of adding your own 'Enable' input which causes the logic to 'behave' as if it would be disabled, while it is actually calculated every cycle. This is similar to power saving methods used in modern electronic devices. Parts which are not required are put into a low power mode instead of keeping them running in an unproductive state.

6.4.6 I/O Driver Settings

Before starting the IEC61131 program, the device performs a check on all defined global variables, which are marked %I, %O, or %M in the physical address parameter. For all variables, which cannot be loaded, because the corresponding data points cannot be located on the device, the I/O driver reports a warning in the system log and shows a listing on the Web interface.

Since IEC61131 program and the data point configuration are downloaded separately, it may be possible that the IEC61131 program does not match the currently active data point interface. In this case it could be dangerous to write values onto those potentially wrong data points. The fact that there are any variables that could not be loaded is taken as an indication for a configuration mismatch. The **I/O check** feature disables the I/O driver of the IEC61131 kernel automatically in this situation. This setting is enabled by default in a new configuration. In the rare case that it is intended to have missing data points, this feature can be disabled in the Configurator. In the system settings de-select the I/O check (see Figure 36). If the I/O driver is disabled, it can be temporarily enabled again on the Web interface until the next reboot.

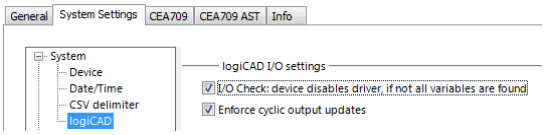


Figure 36: System settings for I/O check

The project setting **Enforce cyclic output update** enables a cyclic update of output data points, which will ensure that those output data points will contain the calculated value after each cycle. This is the default setting. It can be disabled, if the output data point shall be written only if the calculated value changes. In this mode the output data point can be modified over the Web UI for debugging reasons until the IEC61131 program calculates a new value. This mode can also be used to implement an event-style output operation.

6.4.7 PLC Conflicts

PLC output variables are cyclically updating the respective data points, which are configured as PLC out. If those data points are also written to by other objects (e.g. output of a math object, receiver in a connection) this will not have the desired effect. Also favorites that are PLC out and linked to a PLC out data point will result in two different PLC output variables writing to one and the same data point.

The **PLC Conflicts** tab provides information for detecting such write conflicts on PLC out data points. The tab shows a list of PLC write conflicts and the writing objects, which are in conflict with a PLC out data point. Each reported conflict line can be selected. By clicking the **Go to data point** button the Configurator navigates to the conflicting object.

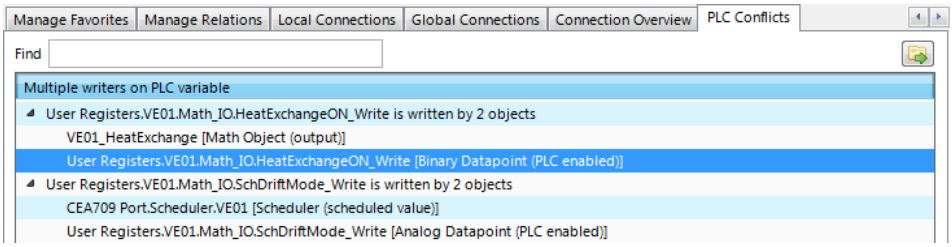


Figure 37: PLC conflicts tab.

An example is shown in Figure 37. The user register 'HeatExchangeON_Write' is written to by the PLC, because it is PLC out. But the register is also an output in the math object 'VE01_HeatExchange'. To resolve the conflict, either remove the register from the math object or remove the PLC out check box from the user register. For doing the latter, select the conflict line reporting the register and select **Go to data point**. Then uncheck PLC out on the data point.

6.5 Workflows

6.5.1 Starting with Data Points

This workflow is based on defining data points used for the IEC61131 program in the L-INX Configurator and then export them to logiCAD. Figure 38 presents the basic steps of this workflow. To follow the steps refer to the quick-start description of Section 3.5.

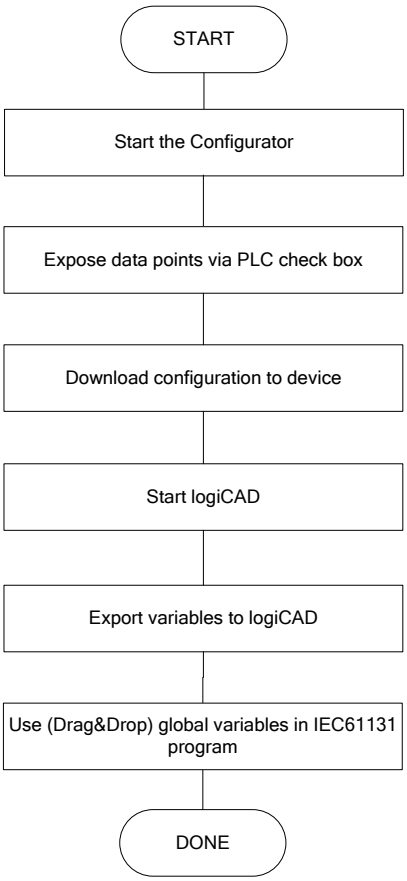


Figure 38: Start with network based information

It is assumed that there are already network data points (CEA-709, I/O, etc.) available. Clicking the Export variables to logiCAD button in the Configurator creates the IEC61131 variables in logiCAD. The following rules are applied in this process:

- The name of the global variables object is derived from the technology base folder (e.g. CEA709, User Registers, etc.). If there is no matching global variables object one is created. If there is already a suitable global variables object, the existing variables are saved.

- There is already a suitable global variables object: If there is an old and new variable with identical name, the type of the variable is checked. In case of a type mismatch of the old and new variable, the old one is discarded and the new one is imported. Additional a warning is printed on the **Error-state viewer**.

- The name of the global variables object represents the folder name in the L-INX Configurator.

After exporting the data points as global variables to logiCAD, they can be used in the Functionplan 'Plan_1'. To open the respective global variables folder double-click on it. To use the imported variables simply drag and drop the required global variable on to the Functionplan, see Figure 39. The external variables (see Section 6.5.2) are automatically created when adding the global variable to the Functionplan.

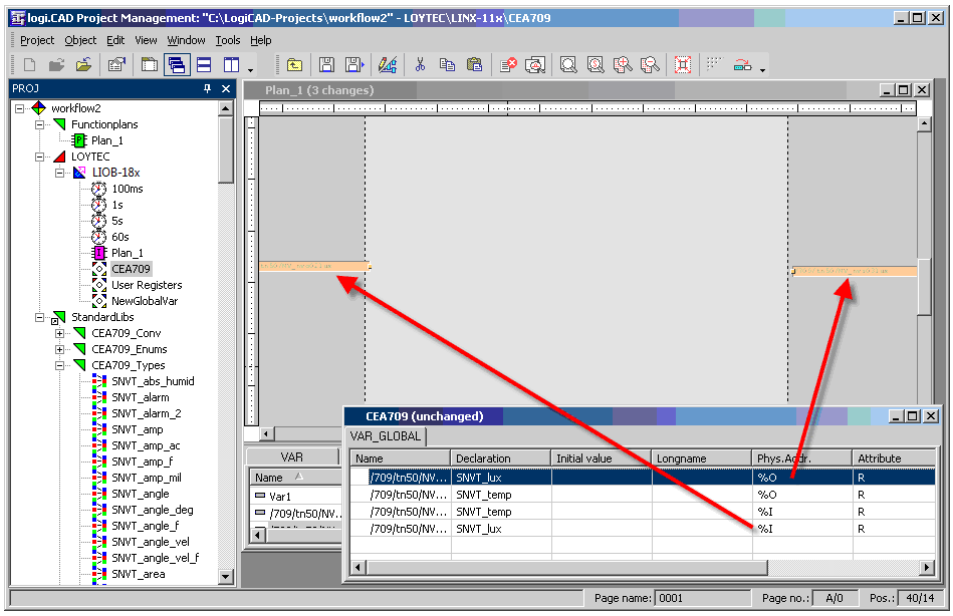


Figure 39: Adding global variables to Functionplans

After adding function blocks that perform the expected work, the IEC61131 program is ready to compile and download (see Section 6.4.2).

6.5.2 Starting with logiCAD

This section introduces a workflow how to develop a new IEC61131 program from scratch using logiCAD. Figure 40 shows the necessary steps to perform. For starting a new logiCAD project refer to the quick-start of Section 3.5.

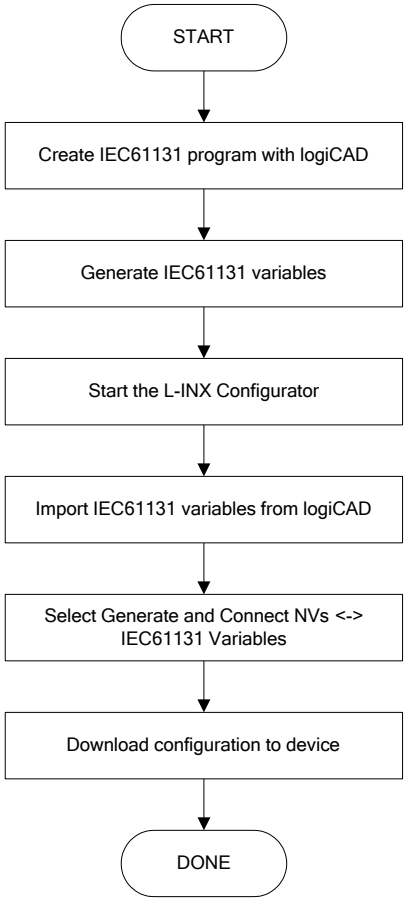


Figure 40: Starting with logiCAD

After creating a new project for a LIOB-18x/48x/58x resource and opening the Functionplan Plan_1 an empty input area is shown, similar to Figure 41.

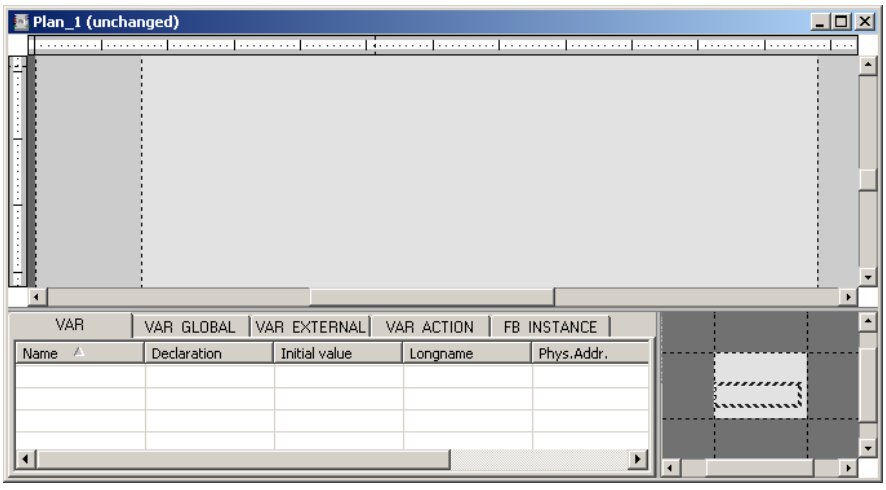


Figure 41: Start new function plan

The dark gray areas on the left and right side are intended to place the input and output variables, the light gray area is used to place the functional blocks.

As described above, global variables are used to interface IEC61131 data points on the device. As assumed for the current workflow, the IEC61131 data points are created on information based on the global variables exported from logiCAD. Hence, for designing the

program, external variables are used during the design phase. To create a new variable select the **VAR_EXTERNAL** tab, right click in the declaration area and select **New**, as shown in Figure 42.

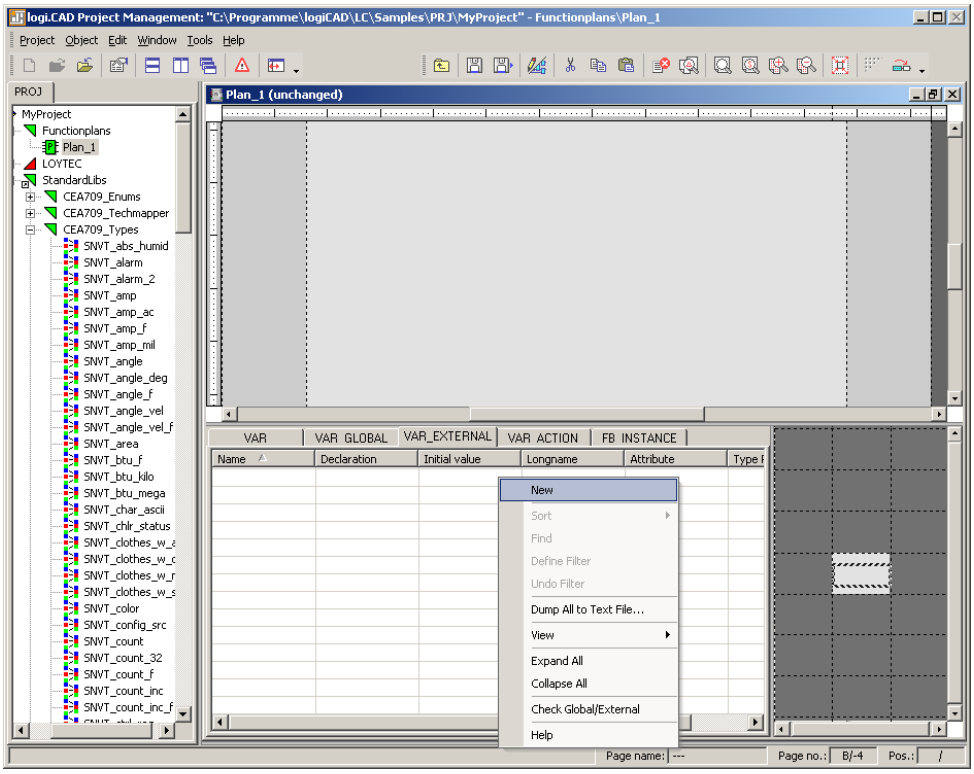


Figure 42: New external variable

In the upcoming dialog the name and the type declaration of the variable must be specified. The type declaration can be done directly by prompting the type into the declaration field, by selecting the type from the pull-down menu or by drag-and-drop of a specific type from the project tree, see Figure 43. Finally, the new variable is added by pressing the **Add** button.

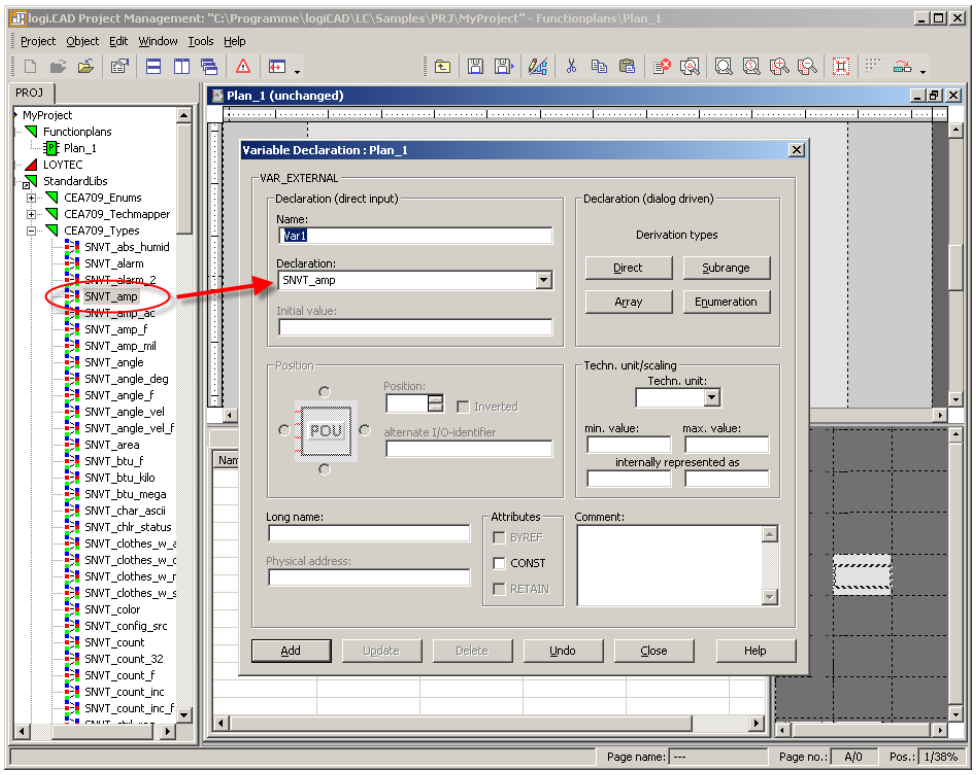


Figure 43: External variable type declaration

The created variable is added to the declaration area and placed to the drawing area by drag-and-drop. At this time the direction of the external variable is not defined, it can be used as input as well as output variable.

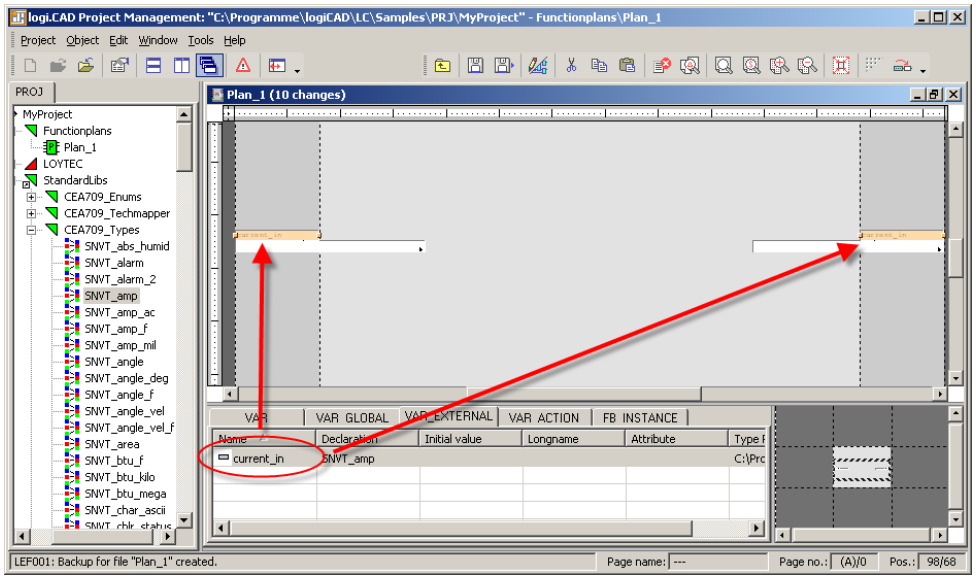


Figure 44: Drag-and-drop external variable to drawing area

Please take care to use an external variable only as input OR output. After adding the external variables to the drawing area, add function blocks to perform the desired actions, see Figure 45 for a sample configuration.

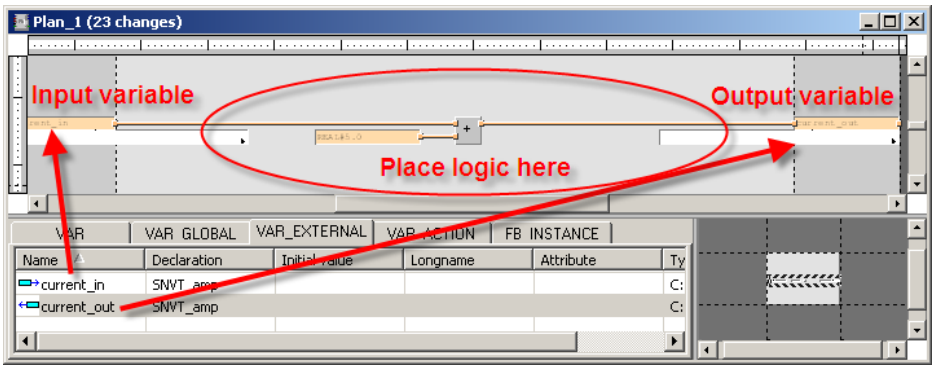


Figure 45: Use external variables

The Functionplan 'Plan_1' represents now a simple program. It adds a defined value to the value of the input variable and sends the result to the output variable.

Add all expected functionality to the 'Plan_1' or use different Functionplans to split up the expected functionality into smaller pieces. But take care about the name and type declaration of external variables when using more than one Functionplan. All external variables with the same name refer to the same global variable.

After adding all functionality, global variables matching the requirements of the defined external variables, must be generated. A tool automatically performs the process of generating the global variables object and the required global variables.

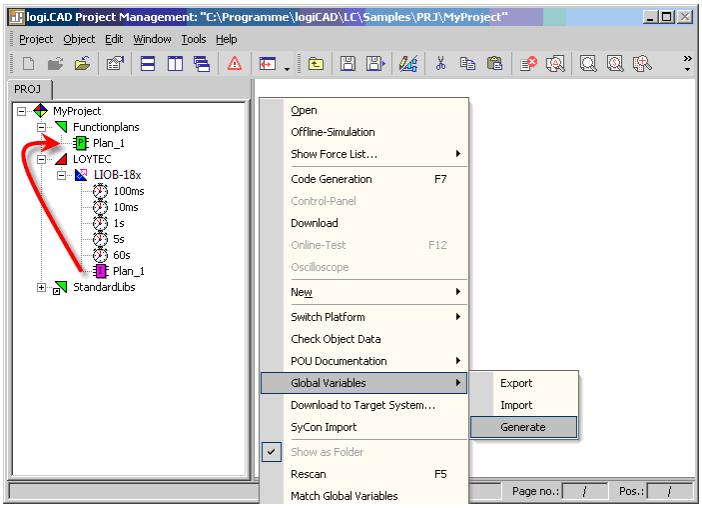


Figure 46: Auto-create global variables

To start the creation of global variables, based on external variables, **Save** all changes, then right click the LIOB-18x/48x/58x device and select **Global Variables → Generate**. Then the selected resource is parsed and every program instance found is checked for external variables. In Figure 46 the type instance Plan_1 refers to the Functionplan 'Plan_1', as defined in Figure 45. If there are more Functionplans than the predefined 'Plan_1' appropriate program instances for these plans must be added to the resource before creating the global variables. Anyway, only Functionplans referred from program instances are executed.

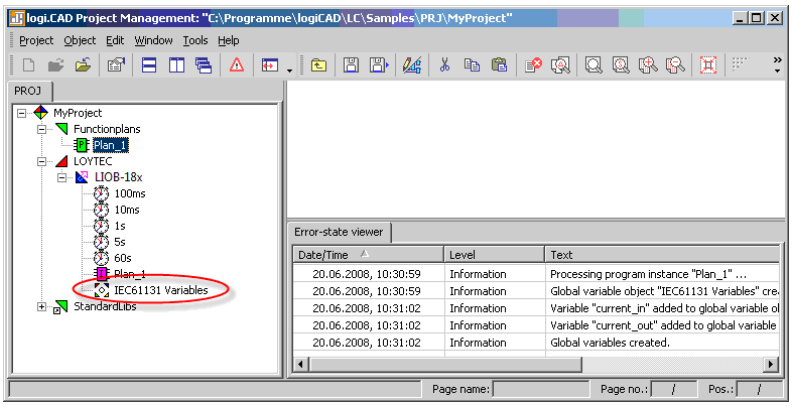


Figure 47: Created global variables object

Now there is a global variables object called IEC61131 Variables available, containing all global representations of the external variables defined before. The **Error-state viewer** reports all processed program instances and added variables, see Figure 47 for details.

The global variables are crated based on the following rules:

- The direction of the variable is determined based on the graphical representation, shown in Figure 48.

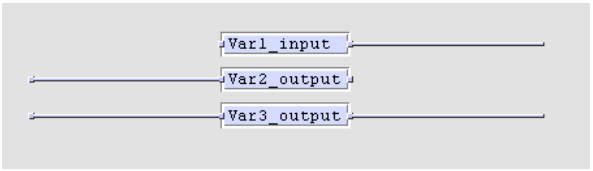


Figure 48: Direction of global variables

  Every external variable connected on the right terminal results in a global input variable. External variables connected on the left terminal or on both sides results in a global output variable. Variables connected on both sides can either be used for the force update feature (see Section 6.6.1) or as marker (see Section 6.6.4). As the tool can not distinguish between these two possibilities, per default a global output variable is created.

- If there is already a global variables object, only new variables are added. In case of external variables using the same name as an already existing global variable, the new definition is used and a warning is printed in the **Error-state viewer**.

- In case of two Functionplans, each referring to a global variable with the same name but a different type, the creation process is stopped and an error is printed in the **Error-state viewer**.

Now the IEC61131 program is ready to compile and download, refer to section 6.4.2 for details.

Based on the above created global variables corresponding IEC61131 data points are created on the device. For doing so, start the L-INX Configurator and click the speed button **Import variables from logiCAD**. Data points are created in the **IEC61131 Variables** folder.

The Configurator reports the results of the import. For the import process the following rules are applied:

- New variables are added
- Variables with same name and type are ignored
- If there are variables with the same name but different type or direction, the variable to import is ignored and a warning is added to the import log.

The name of the folder to import the new variables corresponds to the name of the global variables object. Figure 49 shows the result of the import process.

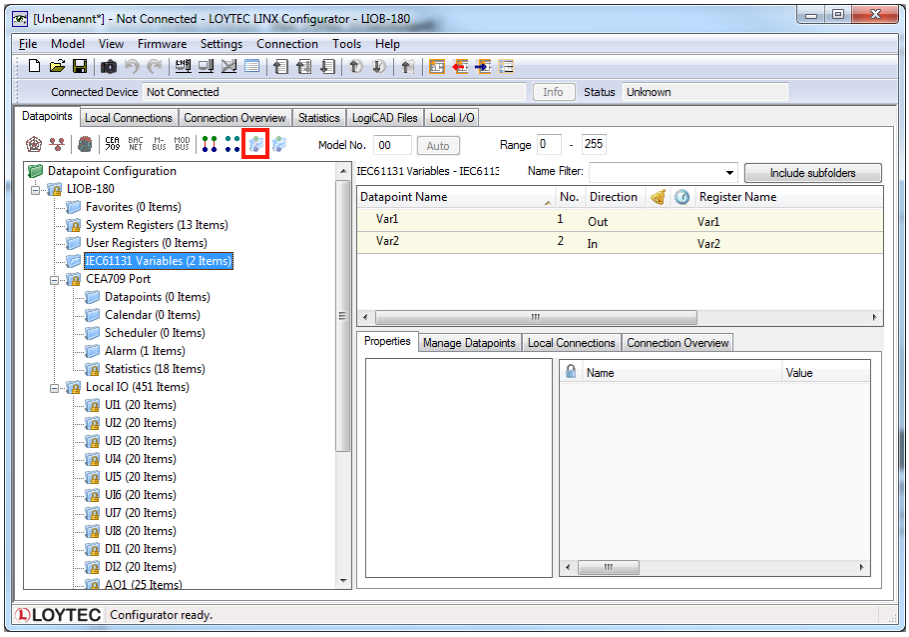


Figure 49: Connect IEC61131 variables

To create the appropriate network data points for all imported IEC61131 variables select the **IEC61131 Variables** folder and press the button **Generate and Connect Network <-> IEC61131 variable from folder**. Check the log output for errors and finally download the configuration to the device.

After rebooting the device, the IEC61131 program is up and running. Check the PLC LED for potential overload.

6.5.3 Pre-compiled IEC61131 Program

In opposite to the last two chapters it is assumed that there are already some components finished, hence starting up from scratch is not suitable. Second, there is the possibility to have an already defined IEC61131 program or an already fixed network interface.

Starting with an already precompiled IEC61131 program results in a similar workflow as presented in Section 6.5.1. The difference is that all logiCAD related tasks are missing. As the IEC61131 program is compiled, the name of the IEC61131 data points is already fixed. The definitions for the data points must be available either in form of a CSV file to import to the L-INX Configurator or as part of a Configurator project. If the network interface for the device was not already defined, the L-INX Configurator can be used to generate and connect the needed data points.

Additionally, there is the possibility that also the network data points are already fixed or that a given, user-defined interface is necessary. Then the developer has to connect the IEC61131 data points to the corresponding network data points by himself.

Finally, after downloading the configuration and rebooting the device, the IEC61131 program can be downloaded to the device via the LINX Configurator. After a final reboot the device loads and executes the IEC61131 program.

6.6 Additional Features

6.6.1 Force Update Functionality

Per default the IEC61131 program only sends updates on changed output values. Every program cycle the input values are fetched, the IEC61131 program is executed and the calculated values are sent to the output driver. If the old values are identically with the new one no updates are sent to the IEC61131 data points. As a result no update is sent to the network.

For some applications, e.g. for a scene controller, it is necessary to send an update on request. E.g. every time the input value is updated, the output value is forwarded to the network, regardless if the value of the output value was changed or not.

For implementing this feature, special vendor blocks are available. First it is necessary to check if there was an update on a selected input within the last execution cycle of the running IEC61131 program. That functionality is offered by the function block Update Notify located in the StandardLibs→Loytec_Support folder. Second, an output must be forced to send an update even if the value was not changed. The function block Force Update is used for that functionality, it is located in the StandardLibs→Loytec_Support.

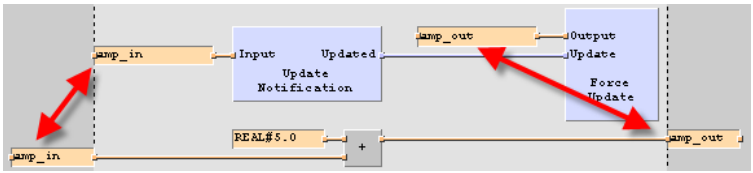


Figure 50: Force update

Figure 50 depicts how to use the force update functionality. Besides the part of the IEC61131 program that defines the calculations to perform (lower half of Figure 50), additional logic for the force update functionality is required. The global input variable that is monitored for changes is connected to the Update Notification function block. As a result, the Boolean output 'Updated' is set to TRUE for one program cycle, if the value of the connected variable has been updated since the last cycle start. To force the I/O driver send out an update, the global output variable, which shall be updated, is connected to 'Output' of the Force Update block. Hence, every time the 'Update' input of the Force Update block is TRUE, the connected global output variable sends out an update at the end of the program cycle.

Important: ***Every global variable connected to the update notification or force update block must be connected via the right-hand side terminal!***

6.6.2 Using UNVT variables (LIOB-18x/48x)

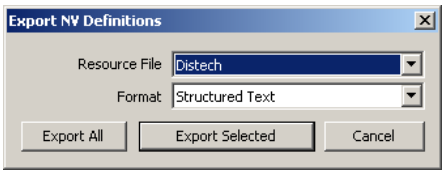
Similar to the predefined CEA709 data types and the technology converter functions, user-defined network variable data types can be used. The L-INX Configurator supports the developer to generate the type definitions needed for UNVTs and enumerations based on LONMARK resources files.

To generate type definitions for UNVTs in logiCAD

1. Start the L-INX Configurator.

2. Select the menu **Tools → Export NV Resource File…**.

3. Select the resource file to be exported. Select structured text as format and export the file to a location of your choice by clicking **Export Selected**.



4. To import the created type definition file into logiCAD, add a new library to the project. Right-click on the created library and select **Export/Import → Start ST-Import**.

5. Select the file to be imported and check the **Error-state viewer** for the results of the import process.

Using the newly created data types, suitable technology converter function blocks can be created. For each UNVT type, create a normal function block to convert an input of the UNVT type to a number of standard IEC61131 data types and vice versa. You may look at the technology converter blocks for SNVT types which are provided by LOYTEC, to get ideas how to implement your own converter functions.

6.6.3 Create Your Own Data Type

For special applications, custom IEC61131 compliant data types may be created by the user, which do not correspond to any CEA709 network type but should still be available as a data point on the device. Further, such data points can also be made persistent, which makes variables of such custom data types persistent.

Most of the IEC61131 data types may be used as global variables on the device and the L-INX Configurator will be able to create a suitable register data point for the IEC61131 data type. Supported data types include custom enumeration types (a multi-state register with the required states will be created automatically), strings (a string data point with a maximum length of 128 characters will be built), and simple arrays.

While the Configurator will be able to automatically determine the required data point size for simple arrays (like ARRAY [1..16] OF INT) and create a suitable register data point during the IEC61131 variable import, the data size of custom structures cannot be determined automatically at the moment, so the L-INX Configurator does not know how to create a suitable user data point. As a workaround, the type name must contain the desired size of the data point in bytes, for example: MyStructuredType(UT16) will tell the L-INX Configurator to create an IEC61131 register of type ‘user’ with a length of 16 bytes, to hold the data for the IEC61131 structure defined in the logiCAD program.

6.6.4 Using Persistent Data Points and Markers

Persistent data points are data points on the device that hold their value even after power loss. There is no difference in handling global variables connected to persistent data points or to non-persistent data points. Global variables connected to persistent data points are marked with retain attribute in logiCAD and with the persistent flag in the L-INX Configurator.

Global input variables marked as persistent, supply the IEC61131 program every time with the last received, valid data, even after a power failure. To set an input variable as persistent in logiCAD, open the global variables object containing the appropriate variable, double click the variable, set the **retain** check box and press the **Update** button. Now export the global variables and import them to the L-INX Configurator. After downloading the new

IEC61131 point configuration to the device, the data point is set to persistent. Setting the data point persistent in the L-INX Configurator, export the data points and import them to logiCAD is also possible.

Global variables declared as marker can be used as input and output variable in IEC61131 programs. LogiCAD is not able to distinguish markers from global output variables used in combination with the force update feature, see Figure 50. As a result it is not possible to create a global variables object holding markers automatically. The procedure as described in Section 6.5.1 is not able to decide when to create an output variable or a marker. Markers must be created manually by adding them to the global variables object and setting the physical address to %M.

6.6.5 Using Retain Variables

Retain variables can be used to maintain certain states or parameters of a program over a reboot in those variables, which cannot be kept in persistent data points. Typically, input variables (%I) will be marked as retain variables. For doing so check the **RETAIN** attribute in the variable declaration when creating or editing a variable as shown in Figure 51.

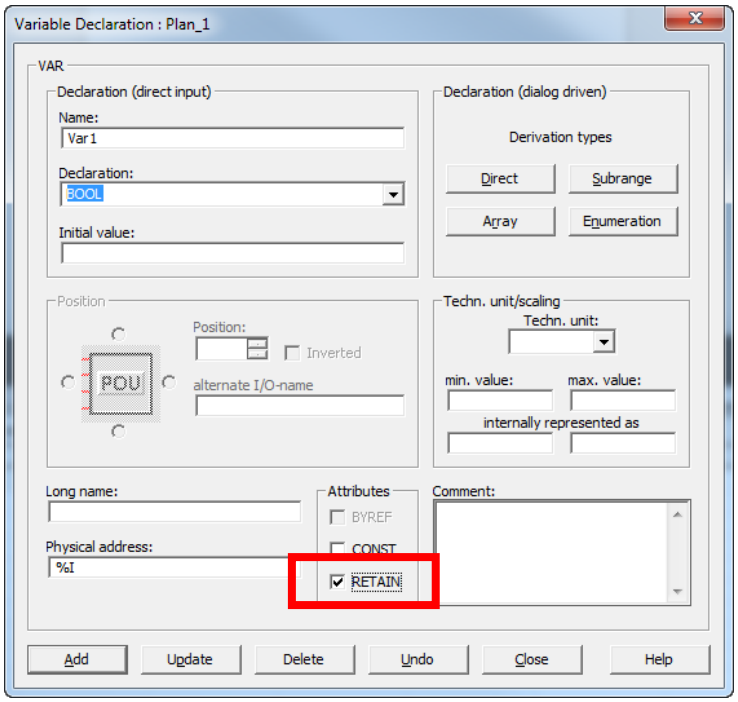


Figure 51: Declaring a retain variable.

The retain attribute should be used in global variables and local variables on function plan only. Global variables on the device resource are managed by the I/O driver and must not use the retain attribute. Instead data point persistency (see Section 6.6.4) must be used for those variables. Furthermore, it is not recommended to mark local variables in function blocks as retain variables.

The contents of retain variables are written to Flash storage every 5 minutes and when shutting down the device. This provides good compromise between write granularity for normal operation and ensuring Flash life-time. If the contents shall be written explicitly at a defined time, use the **RetainCtl** function block located in the **logi.library/Data** folder. Connect the store input (S) and set this input TRUE for one cycle to trigger writing retain data. Make sure to set the input back to FALSE after this cycle to avoid multiple consecutive writes.

6.6.6 System Registers, System Time

System registers, such as the System time or the CPU load, can be used within IEC61131 programs. Therefore, for each system input variable, a global input variable of type UDINT may be created within the IEC61131 program. Then, connections to the appropriate system registers are created manually with the L-INX Configurator.

To use the system time within the IEC61131 program, connect the AtoDT converter (located in the StandardLibs->IEC61131-3_ (EXT) folder) to the global input variable that receives the system time.

6.6.7 Code Protection

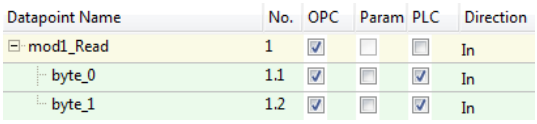
There are 4 data points used for code protection. These data points, in combination with an adapted IEC61131 program, can be used to protect your 61131-Program Intellectual Property. Please contact LOYTEC sales for further information.

6.6.8 Using Structured Data Point Members

Some network technologies provide structured data points. Their types are defined by the network technology and are available in the Configurator. Those structure types can either be used in logiCAD by implementing a technology converter (see Section 6.4.3) or by exposing the sub-data points of the respective structures.

To Use Sub-Data Points in logiCAD

1. In the Configurator, expand the structured data point and expose the desired sub-data points of its structure members to the PLC.



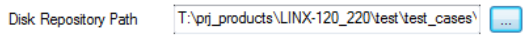
2. Export the variables to logiCAD. The exposed structure members appear for example as '/UR/mod1_Read.byte_0' in the 'User Registers' global variables block.

3. Export the types for the respective type category in the menu **Tools → Manage Structured Types…** . This opens the structured types management dialog.

4. Choose the type **Category**, e.g. 'Modbus'.

5. Click the speed button Save to disk  .

6. In the dialog **Export Type to Disk** choose a **Disk Repository Path** and select the extension '.ST' for structured text.

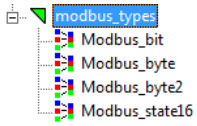


7. Copy all listed types to the disk repository by clicking  . Alternatively select specific types and click  . Then click **Save** and exit the dialog.

8. In logiCAD create a new library folder, e.g. 'modbus_types'.

9. Right-click on that new library and select **Export/Import → Start ST Import**. In the file dialog choose the previously exported structured text file.

10. The type definitions are now available in the new library.



6.6.9 BACnet Server Objects (LIOB-58x)

BACnet server objects provide additional functionality compared to other network technologies. The most important tasks in an IEC61131 program are:

- Reading sensor data from BACnet input objects,
- writing to commandable BACnet objects with a priority,
- revoking values from commandable BACnet objects,
- handling objects when out of service.

The default data flow direction for a commandable BACnet object (e.g. AO) is input to the PLC (%I). This means its value is commanded over the BACnet network and the logic processes the resulting value. The default for non-commandable BACnet objects (e.g. AI) is output (%O). This means the logic writes the value, which is read out of the object over the BACnet network. With this type of variables the BACnet objects are treated as variables of the IEC61131 program.

When exposing I/Os to BACnet server objects, the BACnet objects represent the I/O values and the IEC61131 program behaves like a user from the network on those objects. Typically sensor values are connected to BACnet input objects (e.g. AI). To read the sensor value the PLC needs an input variable. Since the default data point direction is output, a separate access data point for the PLC must be created. For doing so, use the **Add/Remove BACnet properties** option from the context menu of the data point list and create a read data point for the Present_Value. Figure 52 shows an example, where the PLC uses the input '/BAC/AI2.Present_Value_Read' to get the sensor values from AI2.

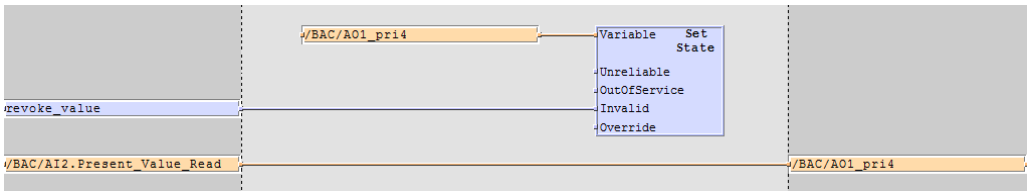


Figure 52: PLC with BACnet objects

BACnet output objects (e.g. AO) are connected to actuator I/Os. On those BACnet objects the PLC behaves like a user from the network and needs to write and revoke values at certain priority slots. For writing with a certain priority to a commandable object, a priority write data point needs to be created (see LINX Configurator User Manual [2], Section BACnet Configuration, Write and Read with Priority). This data point is configured with a BACnet write priority in the Configurator and used as an output in the PLC. The example in Figure 52 writes with priority '4' to AO1 over '/BAC/AO1_pri4'.

To revoke values in commandable server objects, the **SetValueState** function block must be used. It is located in the **Loytec_Support/Service** folder. The I/O variable of the controlled BACnet object is connected to the **Variable** input of the function block. A Boolean signal needs to be connected to the **Invalid** input. When this input changes to TRUE, the value is revoked at the respective priority slot. When it remains TRUE in the next cycle the service

function block is idle. Figure 52 shows an example, how to revoke a value in ‘AO1’ at priority slot ‘4’.

With the out-of-service feature BACnet server objects can be decoupled from the senor and actuator I/O equipment. When taken out of service, the I/O variables of the PLC are then decoupled from the network. The input variable Present_Value_Read of an AI no longer tracks the sensor value but reflects the value set in the AI from a BACnet OWS. This holds true for all non-commandable BACnet objects. Writing to the priority output variable of an AO only updates the priority array but no longer drives the actuator. This holds true for all commandable BACnet objects.

The out-of-service state is reflected in the data point status and can be accessed with the **ValueState** function block. This function block reads the value state and is used in a similar fashion as the **SetValueState** function block.

7 Firmware Update

The L-IOB firmware supports remote upgrade over the CEA-709 interface (LIOB-18x), the CEA-852 interface (LIOB-48x), and over IP (LIOB-48x/58x). To guarantee that the L-IOB is not destroyed due to a failed firmware update, the L-IOB firmware consists of two images:

1. fallback image,
2. primary image.

The 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-IOB device boots up with the fallback image, the status LED is flashing red.

7.1 Firmware Update via the Configurator

The primary image can be updated using the Configurator Software. See the LINX Configurator User Manual [2] on how to install the Software.

To Update the Firmware using the Configurator

1. Start the Configurator from the Windows Start menu: **Start → Programs → LOYTEC LINX Configurator → LOYTEC LINX Configurator**.
2. Select the menu: **Connection → Connect to device** or **Connection → Connect via LNS**. See the LINX Configurator User Manual [2] for setting up the connection.
3. In the connection dialog click on **Connect**.
4. Optionally, check for updates by selecting the menu **Help → Check for updates** . This function checks for new firmware and Configurator versions.
5. Select the menu: **Firmware → Update**.
6. This opens the Firmware Update dialog as shown in Figure 53. Click on the button … and select the firmware image.

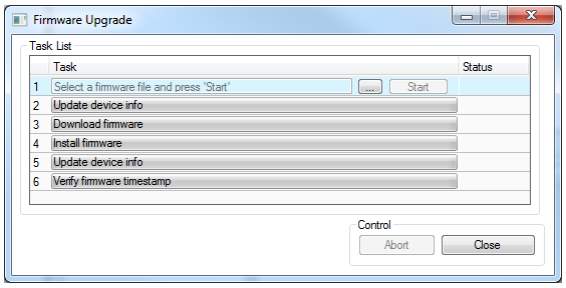


Figure 53: Firmware Update dialog of the Configurator

7. Click on **Start** and observe the download progress.
8. When the download is complete, a dialog appears. Click **OK**.
9. In the Firmware Update dialog, click **Close**.
10. The device's firmware has now been successfully upgraded.

7.2 Firmware Update via the Web Interface (LIOB-48x/58x)

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 the LOYTEC Device User Manual [1].

8 Troubleshooting

8.1 Technical Support

LOYTEC offers free telephone and e-mail support for the L-IOB 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

e-mail : ***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

e-mail: ***support@loytec-americas.com***
Web: ***http://www.loytec-americas.com***
tel: ***+1 (512) 402 5319***
fax: ***+1 (262) 408 5238***

8.2 Remote Packet Capture (LIOB-48x/58x)

8.2.1 Configure Remote Packet Capture

Remote packet capture is able to capture packets on the Ethernet port. To enable the remote packet capture feature, go to the **Config / Port Config** Web page and enable Remote packet capture as shown in Figure 54.

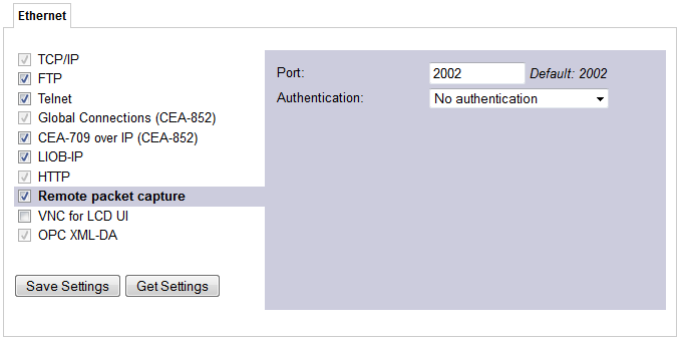


Figure 54: Remote packet capture port configuration.

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.

8.2.2 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 55.

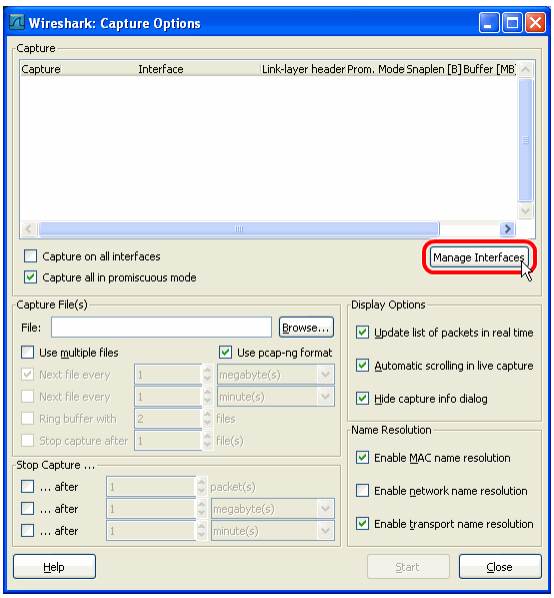


Figure 55: Wireshark Capture Options Dialog.

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 56.

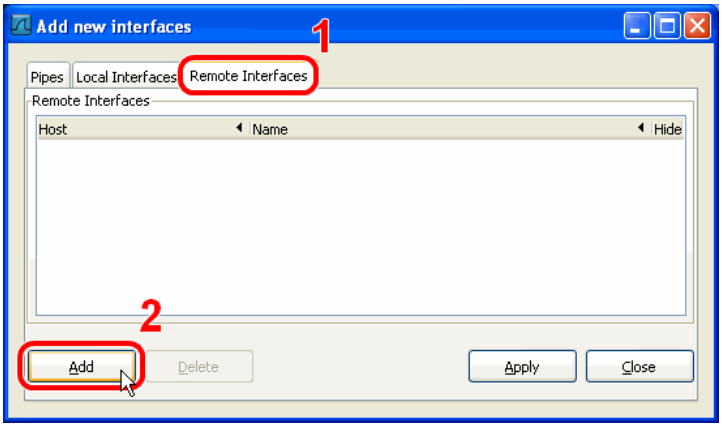


Figure 56: Wireshark Add New Interfaces Dialog.

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 57.

5. Note that only the users **admin** and **operator** are allowed to connect.

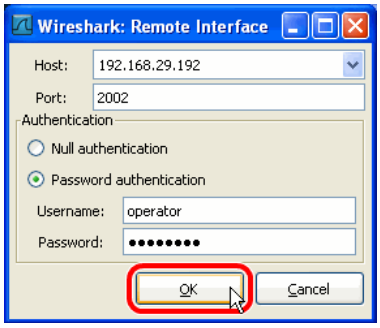


Figure 57: Wireshark Remote Interface Dialog.

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 58.

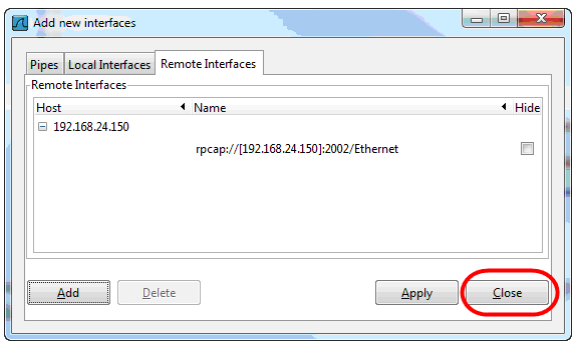


Figure 58: Added new interface to Wireshark.

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 ‘Ethernet’.

2. Click the **Start** button as shown in Figure 59.

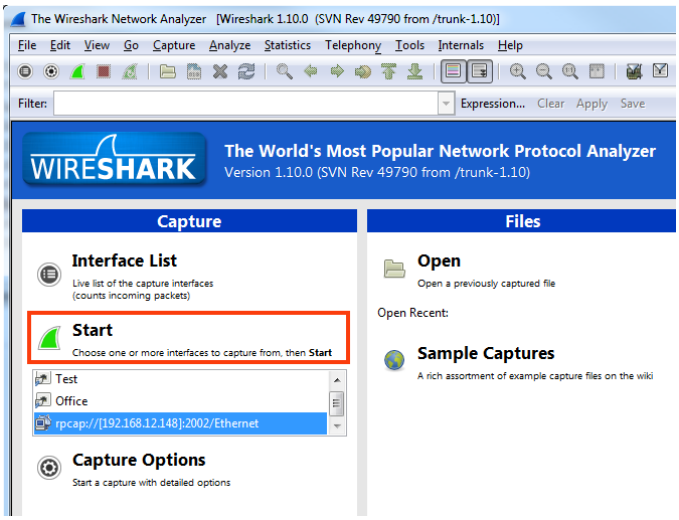


Figure 59: Start Remote Capture in Wireshark.

3. Wireshark will attempt to establish a connection to the device and, if successful, start displaying packets. An example capture is shown in Figure 60.

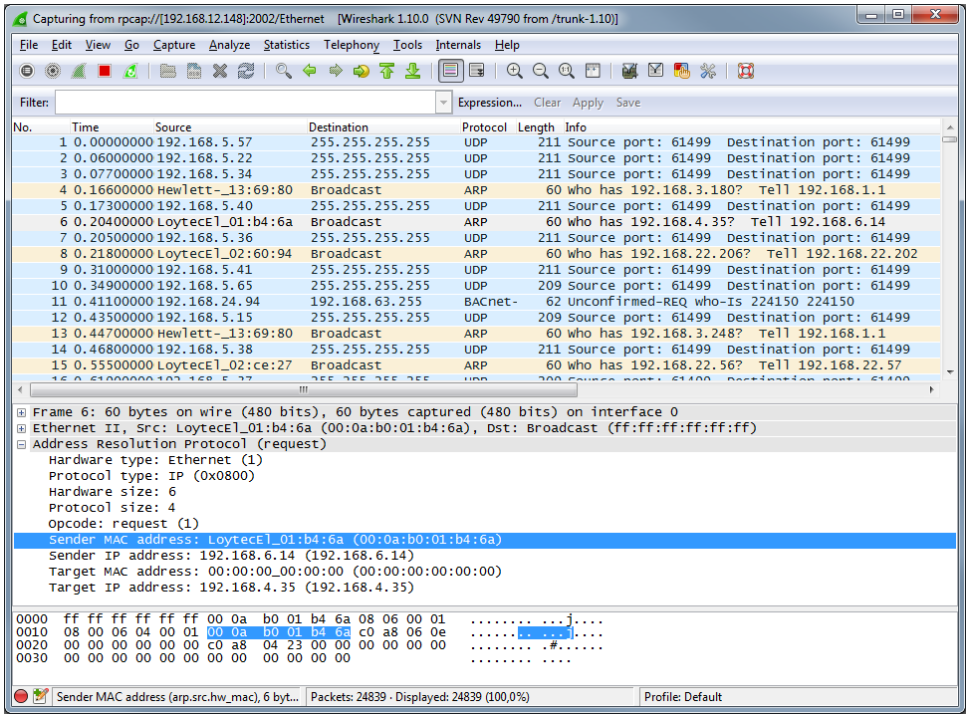


Figure 60: Example remote capture in progress.

9 Application Notes

9.1 External Power Supply (not using LPOW-2415A)

When using a non-LOYTEC power supply (see Figure 61), the following rules must be complied with:

- Consistent polarity must be maintained when connecting LOYTEC I/O controllers and modules to the transformer. That is, the '– ~' terminal of each I/O controller and each I/O module must be connected to the same terminal on the secondary side of the transformer.
- The I/O controllers and modules are half-wave rectified. Connecting two half-wave rectified devices to the same transformer without maintaining polarity will cause short circuit.
- The GND terminals of the I/O controller or module are internally wired to the '– ~' terminal. Therefore, if powering I/O controllers and modules with the same transformer, it is again essential to maintain polarity. Failure to do so will result in short circuit and/or damaged device.
- If the transformer output must be grounded, connect the '– ~' terminal to earth ground.

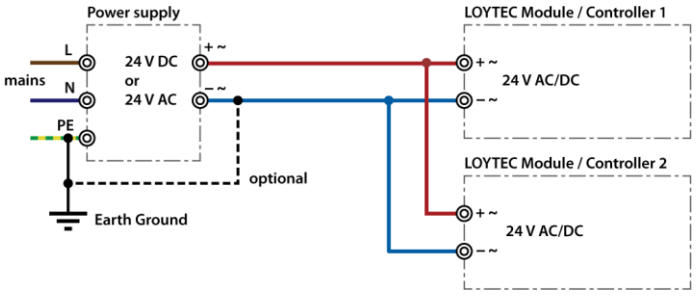


Figure 61: External power supply

9.2 Physical Connection of Inputs

9.2.1 Connection of Switches

On- or off-switches can either be connected to the DIs (Digital Inputs) or to the UIs (Universal Inputs) in digital interpretation.

9.2.1.1 Switch connected to a DI

A switch can be directly connected to a digital input as shown in Figure 62.

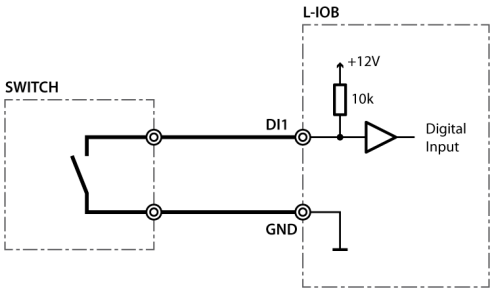


Figure 62: Switch connected to DI

The digital inputs (DI) recognize the following digital signals according to the connected resistance (switch):

Resistance of Switch	**Status**
< 6.8 kΩ	Closed Switch
> 10 kΩ	Open Switch

9.2.1.2 Switch connected to a UI

A switch can be directly connected to a universal input with signal type resistance as shown in Figure 63.

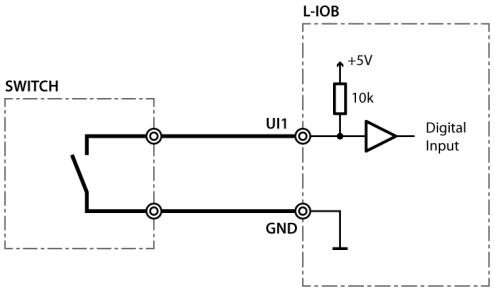


Figure 63: Switch connected to UI

UIs recognize the following digital signals according to the input resistance (switch):

Resistance Switch	**Status**
< 1.9 kΩ	Closed Switch
> 6.7 kΩ	Open Switch

9.2.2 Connection of S0 Pulse Devices (Meters)

S0 pulse meters must be connected to digital inputs (DI) as shown in Figure 64.

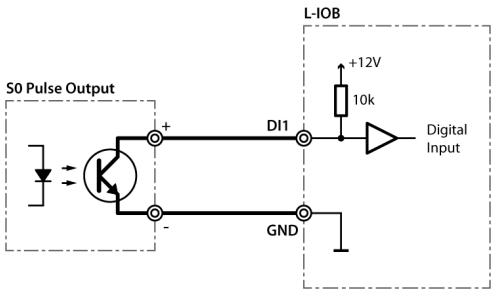


Figure 64: S0 pulse meter connected to DI

9.2.3 Connection of Voltage Sources to Universal Inputs

The Universal Input (UI) provides voltage measurement both if used as an analog or digital input. The signal type must be configured to 'Voltage 0-10V' or 'Voltage 2-10V' in both cases.

9.2.3.1 Voltage Source connected to UI with Analog Interpretation

Figure 65 shows the connection of a voltage source to a universal input in analog mode.

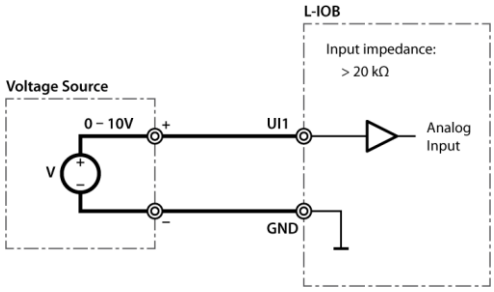


Figure 65: Voltage source on UI in analog mode

9.2.3.2 Voltage Source connected to UI with Digital Interpretation

Figure 66 shows the connection of a voltage source to a universal input in digital mode. In this case, the voltage source acts as a switch with the depicted low and high levels.

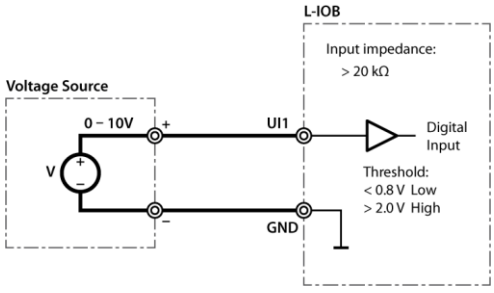


Figure 66: Voltage source on UI in digital mode

9.2.4 Connection of 4-20mA Transmitters to Universal Inputs

9.2.4.1 4-20mA Transmitter connected to UI with Internal Shunt

Some universal inputs have an internal shunt which can be activated (in pairs with another UI) in the Configurator software (signal type 'Current 4-20mA int. Shunt'). Which UIs are equipped with shunts is documented in Section 11.3 and the following Sections. Figure 67 shows the connection of a 4-20mA transmitter to a universal input with internal shunt.

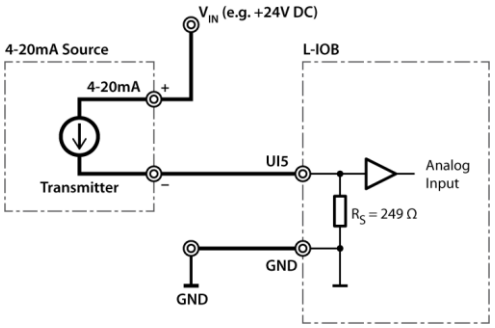


Figure 67: 4-20mA transmitter with internal shunt on UI

9.2.4.2 4-20mA Transmitter connected to UI with External Shunt

On universal inputs, which do not have an internal shunt, an external shunt must be used as shown in Figure 68. The signal type must be set to 'Current 4-20mA' in the Configurator software.

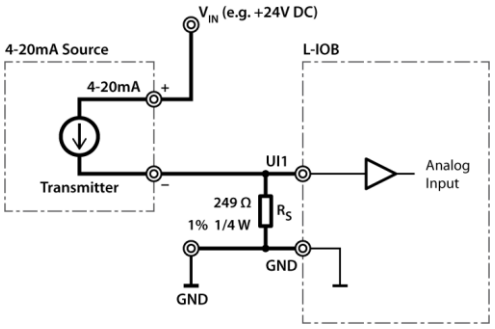


Figure 68: 4-20mA transmitter with external shunt on UI

9.2.5 Connection of Resistive Sensors

Figure 69 shows the connection of resistive sensors to the universal inputs with a temperature sensor as an example. Sensors in the resistance range of 1 kΩ to 100 kΩ can be measured. The signal type must be set to 'Resistance' in the Configurator software.

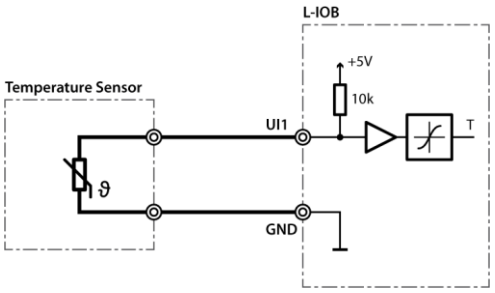


Figure 69: Temperature measurement on UI

9.2.6 Connection of STId Card Readers

Figure 70 shows the connection of an STId card reader to three L-IOB inputs (UIs or DIs). Observe that the clock signal must be connected to an interrupt-capable input of the L-IOB device. More information on STId card readers can be found in the LINX Configurator User Manual [2].

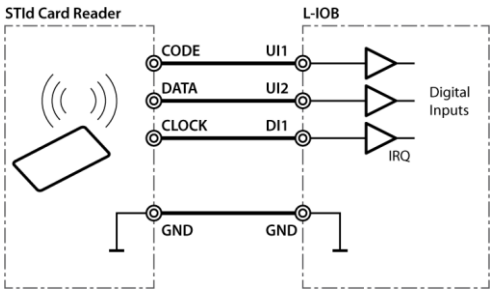


Figure 70: STId card reader

9.3 Physical Connection of Outputs

9.3.1 6A Relays with one External Fuse

The total current of all used 6A relays must be restricted to 6A, if more than two relays share a common (COM) terminal. The wiring shown in Figure 71 can be used for all L-IOB models with common terminals.

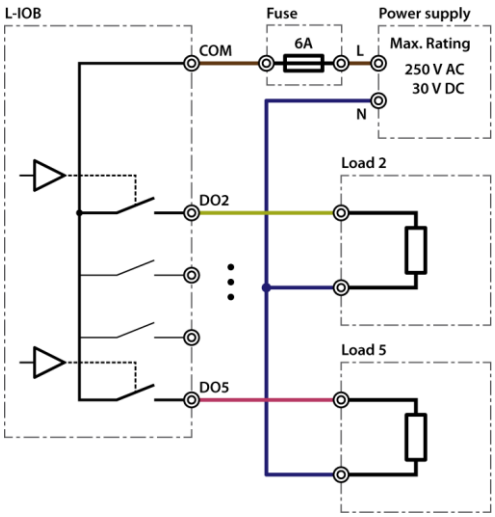


Figure 71: 6A relays with one external fuse

9.3.2 6A Relays on LIOB-xx2 using Separate Fuses

Figure 72 shows the wiring of the 6A relays for the LIOB-182/482/582 models using separate fuses. In this case, two relays share one common terminal (COM).

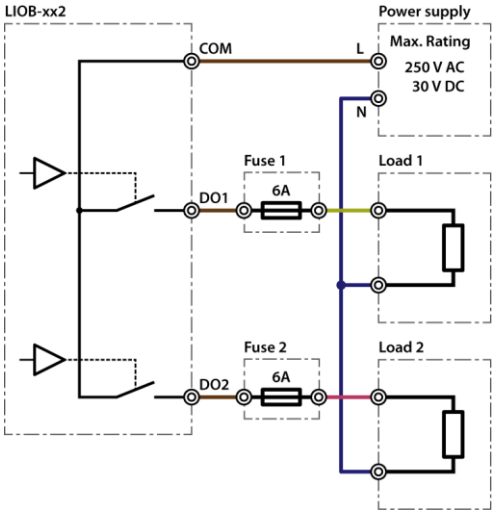


Figure 72: LIOB-182/482/582 6A relays

9.3.3 16A and 6A Relays on LIOB-xx3

The 16A and 6A relays on the LIOB-183/483/583 models all have two separate terminals per relay. There are no common (COM) terminals. This means that a 16A (or 6A) fuse must be wired to one of the two terminals of each relay, as shown in Figure 73.

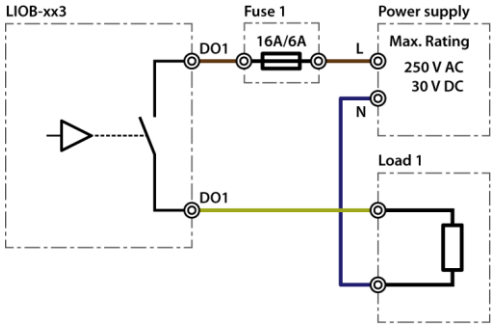


Figure 73: LIOB-183/483/583 16A/6A relays

9.3.4 External Relays and Inductive Loads

When controlling an external relay or inductive load using a L-IOB relay, either an integrated suppressor circuit must be used for the inductor, or a free-wheeling diode, a varistor, RC circuit, etc. must be installed to suppress voltage peaks and sparking due to switching off inductive circuits. It is recommended to use diodes that are part of the 1N400x family and to place them close to the relay, as shown in Figure 74. Figure 75 shows the connection of a 230V relay with a varistor.

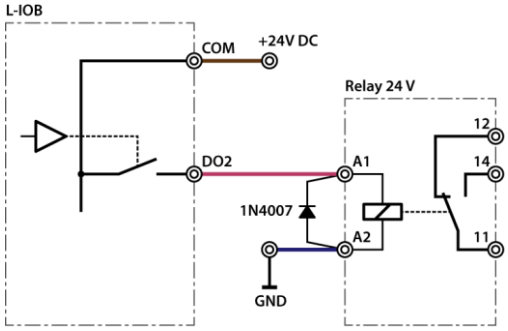


Figure 74: Suppressor circuit with free-wheeling diode

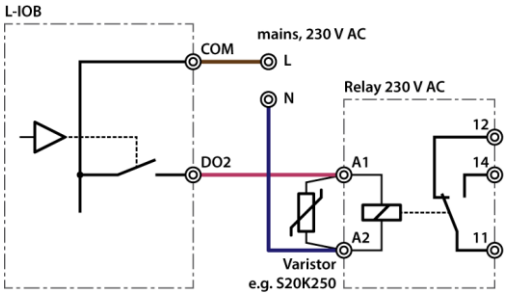


Figure 75: Suppressor circuit with varistor

9.3.5 1A Triacs

Figure 76 shows the wiring of the 1A Triac Outputs.

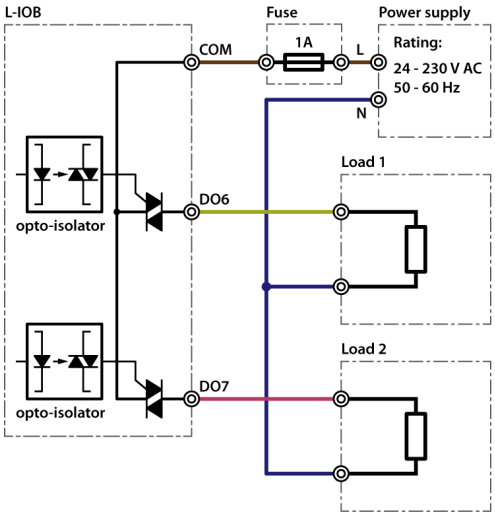


Figure 76: 1A Triacs

9.3.6 Analog Outputs

Figure 77 shows the wiring of the analog outputs (AO). Observe that the analog outputs are labeled ‘0-10V OUT’ but are in fact capable of delivering over 11V.

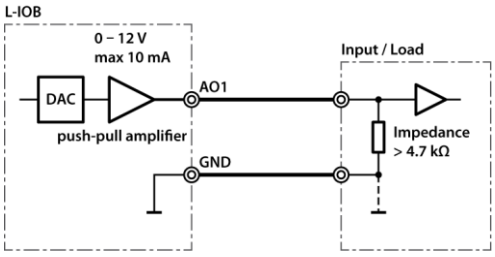


Figure 77: Analog outputs

The input impedance of the connected load must be greater than or equal to 4.7 kΩ for linear output.

9.4 Redundant Ethernet (LIOB-48x/58x)

9.4.1 Ethernet Cabling Options

Some L-IOB models 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-IOB device, 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-IOB device 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.

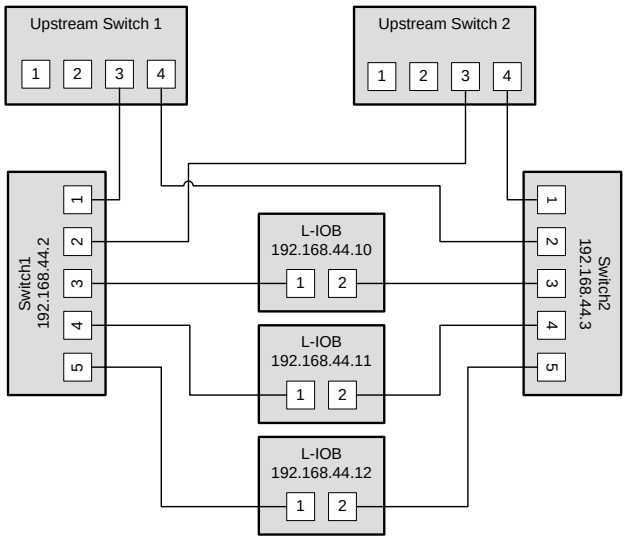


Figure 78: Fully redundant Ethernet topology

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 78, the L-IOB devices 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 79, the L-IOB 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.

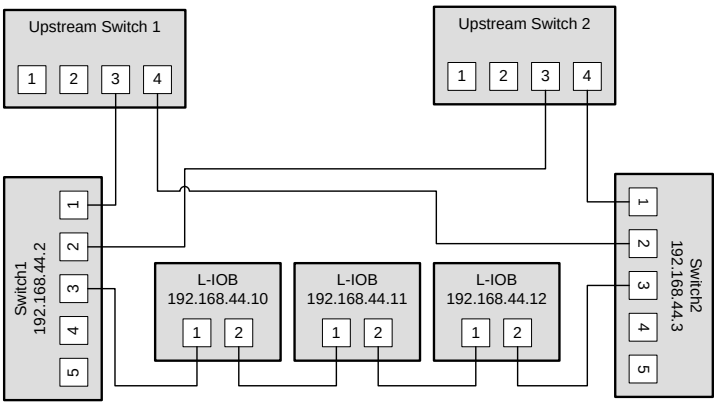


Figure 79: Ring Ethernet topology

9.4.2 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-IOB 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 80.

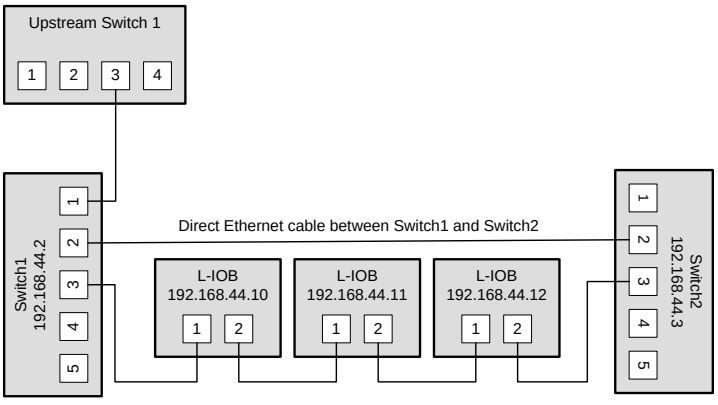


Figure 80: Single upstream connection.

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-IOB 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 78.

9.4.3 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.

9.4.4 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.

9.4.5 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-IOB 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-IOB device directly connected to Switch1.

  - The connection between Switch2 and the L-IOB device directly connected to Switch2.

  - A connection in the middle of the chain, which is not connected directly to either Switch1 or Switch2.

9.4.6 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 78.

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 Security Hardening Guide

This guide contains security-relevant information for operating the LIOB-48x/58x on IT networks. The information refers to the firmware version and the instructions found in the previous chapters of this User Manual.

10.1 Installation Instructions

Install the device over the Web interface:

- Set up the basic device functions and protocol settings as described in Chapter 3.
- Disable the FTP, and Telnet servers in the IP port configuration as described in the LOYTEC Device User Manual.

10.2 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 7.

10.3 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. 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 (LIOB-58x): 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 10.1:

- 21 tcp: This port is opened by the FTP server. The port can be changed and disabled.
- 22 tcp: This port is opened by the SSH 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.
- 5900 tcp: This port is opened by the VNC server, if it is enabled. This port is disabled by default. The port can be changed.
- 2048 tcp: This port is opened by the logiCAD online test. It cannot be changed. The service can be disabled but the port will remain open.
- 16028/16029 udp: These ports are opened for LIOB-IP on the device. These ports cannot be changed. They can be disabled.

10.4 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 (LIOB-58x): 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 10.1:

- 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.
- VNC: The VNC server can be used for remote access to the LCD display on devices that have it. The service is disabled by default.
- logiCAD online test: This service is used by the L-logiCAD programming tool for online debugging of IEC61131 programs. It is enabled by default. The service can be disabled.
- LIOB-IP: This service is used by the L-IOB host function to operate LIOB-IP I/O modules. This service is enabled by default on all L-INX devices. The service can be disabled.

10.5 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 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 node.
- 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.

11 Specifications

11.1 I/O Specification

11.1.1 UI - Universal Input

UIs are universal inputs for four different input types. They have an input voltage range of 0 to 10V, and can withstand up to 30V. The UIs correspond to class 1 with a relative accuracy of +/-1% (of measured value) between 1V and 10V, and an absolute accuracy of +/-10mV between 0V and 1V. The ADC resolution is 16 bits. Galvanically isolated sensors resp. switches must be connected. Universal inputs can be configured as:

- **Binary Input (Digital Input):** input impedance > 20kΩ, sampling period 10ms.
    - In voltage mode, the threshold values are < 0.8V for low level and > 2V for high level.
    - In resistance mode, the threshold values are < 1.9kΩ for low level and > 6.7kΩ for high level.

  Between the threshold values, the resulting level of the UI is not defined.
- **Voltage Metering 0-10V**: input impedance > 20kΩ, sampling period < 1s.
- **Current Loop 4-20mA**: input impedance 249Ω, sampling period < 1s. An internal shunt of 249Ω is available for some universal inputs. Otherwise, an external resistor of 249Ω must be used as a shunt.
- **Resistance Measurement**: input impedance 10kΩ, sampling period < 1s. Resistors in the range of 1kΩ to 100kΩ can be measured.

The average sampling period *p* of analog inputs depends on the number of active (non-disabled) universal inputs *n* that are configured in analog mode. The formula for *p* is:

$$p = n * 125ms$$

This means if e.g. only two UIs are configured as analog inputs, a new sample is taken every 250ms (on average) for each of the two inputs. The UIs configured as digital inputs are unaffected (sampling period always 10ms) by this formula.

11.1.2 DI - Digital Input, Counter Input (S0-Pulse)

DIs are fast binary inputs, which can also be used as counter inputs (S0). They follow the S0 specification for electric meters and have a sampling period of 10ms. They change state at a

load of 195Ω between the DI terminal and GND. Galvanically isolated sensors resp. switches must be connected.

11.1.3 AO - Analog Output

AOs are analog outputs with a signal range of 0 to 10V (up to 12V), a resolution of 10 bits, and a maximum output current of 10mA (short circuit proof). The accuracy over the whole range is +/-100mV.

11.1.4 DO - Digital Output

The following digital outputs are available:

- Relay 6A Output: Switching capacity 6A, 250VAC resp. 30VDC.
- Relay 16A Output: Switching capacity 16A, 250VAC resp. 30VDC.
- TRIAC Output: Switching capacity 1A, 24 to 230VAC.

When connecting an air gap switch to a L-IOB relay, a quenching circuit like a varistor or RC element must be used.

11.1.5 PRESS - Pressure Sensor

These inputs represent differential pressure sensors which measure pressures from 0 - 500 Pascal. They are equipped with two 3/16" (4.8 mm) hose connectors.

11.2 Internal Translation Tables

The L-IOB devices offer fixed internal translation tables for easy configuration of some temperature sensors. The xin/xout values of these tables are listed in Table 4.

xout: Temp. [°C]	xin: Resistance [Ω]			
	PT1000	**NTC10K**	**NTC1K8**	**Ni1000**
-30	882.2	176680	24500	842
-20	921.6	96970	14000	893
-10	960.9	55300	8400	946
0	1000.0	32650	5200	1000
10	1039.0	19900	3330	1056
20	1077.9	12490	2200	1112
25	1097.4	10000	1800	1141
30	1116.7	8060	1480	1171
40	1155.4	5320	1040	1230
50	1194.0	3600	740	1291
60	1232.4	2490	540	1353
70	1270.0	1750	402	1417
80	1308.9	1260	306	1483
90	1347.0	920	240	1549
100	1385.0	680	187	1618
120	1460.6	390	118	1760

Table 4: Internal translation table values

11.3 Specification for LIOB-18x Models

Dimensions [mm]	107 x 100 x 75 (L x W x H)				
Operating Temperature (ambient)	0°C to +50°C				
Storage Temperature	-10°C to +85°C				
Humidity (non condensing) operating / storage	10 to 90 % RH @ 50°C				
Environmental Protection	IP 40 (enclosure); IP 20 (screw terminals)				
Power Supply	24 VDC / 24 VAC ±10 %				
Installation	DIN rail mounting (EN 50 022)				
Interface	1 x CEA-709/FT				
Types	LIOB-180	LIOB-181	LIOB-182	LIOB-183	LIOB-184
Power Consumption [W]	1.7 / 2.6 (all Relays on)	1.7	1.7 / 2.7	1.7 / 2.5	1.7 / 2.5
Universal Input (UI)	8	8	6	6	7 + 1 x Pressure Sensor
Digital Input (DI)	2	12	-	-	-
Analog Output (AO)	2	-	6	6	4
Digital Output (DO)	8 (4 x Relay, 4 x Triac)	-	8 (8 x Relay)	5 (4 x Relay 16 A, 1 x Relay 6 A)	7 (5 x Relay, 2 x Triac)
Digital output specification	Relay: 6 A Triac: 1 A @ 24-230 VAC	-	Relay: 6 A	Relay: 16 A and 6 A	Relay: 6 A Triac: 1 A @ 24-230 VAC
Internal shunt available for current measurement	UI5 & UI6, UI7 & UI8	UI5 & UI6, UI7 & UI8	UI3 & UI4, UI5 & UI6	UI3 & UI4, UI5 & UI6	UI5 & UI6, UI7
Supported STId card readers	3	2	-	-	1

11.4 Specification for LIOB-48x Models

Dimensions [mm]	107 x 100 x 75 (L x W x H)				
Operating Temperature (ambient)	0°C to +50°C				
Storage Temperature	-10°C to +85°C				
Humidity (non condensing) operating / storage	10 to 90 % RH @ 50°C				
Environmental Protection	IP 40 (enclosure); IP 20 (screw terminals)				
Power Supply	24 VDC / 24 VAC ±10 %				
Installation	DIN rail mounting (EN 50 022)				
Interface	1 x IP-852				
Types	LIOB-480	LIOB-481	LIOB-482	LIOB-483	LIOB-484
Power Consumption [W]	1.7 / 2.6 (all Relays on)	1.7	1.7 / 2.7	1.7 / 2.5	1.7 / 2.7
Universal Input (UI)	8	8	6	6	7
Digital Input (DI)	2	12	-	-	-
Analog Output (AO)	2	-	6	6	4
Digital Output (DO)	8 (4 x Relay, 4 x Triac)	-	8 (8 x Relay)	5 (4 x Relay 16 A, 1 x Relay 6 A)	7 (5 x Relay, 2 x Triac)
Digital output specification	Relay: 6 A Triac: 1 A @ 24-230 VAC	-	Relay: 6 A	Relay: 16 A and 6 A	Relay: 6 A Triac: 1 A @ 24-230 VAC
Internal shunt available for current measurement	UI5 & UI6, UI7 & UI8	UI5 & UI6, UI7 & UI8	UI3 & UI4, UI5 & UI6	UI3 & UI4, UI5 & UI6	UI5 & UI6, UI7
Supported STId card readers	3	2	-	-	1

11.5 Specification for LIOB-58x Models

Dimensions [mm]	107 x 100 x 75 (L x W x H)				
Operating Temperature (ambient)	0°C to +50°C				
Storage Temperature	-10°C to +85°C				
Humidity (non condensing) operating / storage	10 to 90 % RH @ 50°C				
Environmental Protection	IP 40 (enclosure); IP 20 (screw terminals)				
Power Supply	24 VDC / 24 VAC ±10 %				
Installation	DIN rail mounting (EN 50 022)				
Interface	1 x BACnet/IP				
Types	LIOB-580	LIOB-581	LIOB-582	LIOB-583	LIOB-584
Power Consumption [W]	1.7 / 2.6 (all Relays on)	1.7	1.7 / 2.7	1.7 / 2.5	1.7 / 2.7
Universal Input (UI)	8	8	6	6	7
Digital Input (DI)	2	12	-	-	-
Analog Output (AO)	2	-	6	6	4
Digital Output (DO)	8 (4 x Relay, 4 x Triac)	-	8 (8 x Relay)	5 (4 x Relay 16 A, 1 x Relay 6 A)	7 (5 x Relay, 2 x Triac)
Digital output specification	Relay: 6 A Triac: 1 A @ 24-230 VAC	-	Relay: 6 A	Relay: 16 A and 6 A	Relay: 6 A Triac: 1 A @ 24-230 VAC
Internal shunt available for current measurement	UI5 & UI6, UI7 & UI8	UI5 & UI6, UI7 & UI8	UI3 & UI4, UI5 & UI6	UI3 & UI4, UI5 & UI6	UI5 & UI6, UI7
Supported STId card readers	3	2	-	-	1

11.6 Resource Limits

Table 5 specifies the resource limits of the different L-IOB models.

Model / Limits	18x	48x	58x
Total number of data points	2000	2000	2000
OPC Tags	-	200	200
User Registers	1000	1000	1000
IEC61131 Variables	1000	1000	1000
CEA-709 NVs	200	200	-
CEA-709 Alias NVs	200	200	-
CEA-709 Address table entries/non-ECS	256/15	256/15	-
BACnet Server Objects	-	-	200
BACnet Client Mappings	-	-	200
Calendar Patterns	25	25	25
Scheduler objects	10 (max. AST configuration size 384KB, 64 data points per scheduler)		10
Alarm Servers	1	1	32
Trend Logs	-	50	50
Total trended data points	-	100	100
Total aggregated size	-	6MB	6MB
E-mail templates	-	20	20
Math objects	-	20	20
Alarm Logs	-	5	5
Connections (local)	200	200	200
Connections (global)	-	100	100
L-WEB Clients (concurrent)	-	8	8
Extension with external L-IOB Modules	-	1 x LIOB-45x	1 x LIOB-55x

Table 5: Resource limits of different L-IOB models

12 References

[1] LOYTEC Device User Manual, LOYTEC electronics GmbH, Document № 88086501, March 2016.

[2] LINX Configurator User Manual, LOYTEC electronics GmbH, Document № 88086701, March 2016.

[3] NIC User Manual, LOYTEC electronics GmbH, Document № 88067217, May 2013.

[4] LIOB-10x/x5x User Manual, LOYTEC electronics GmbH, Document № 88078519, April 2014.

13 Revision History

Date	Version	Author	Description
2016-03-23	6.0	STS	Re-organized User Manual structure, branched out common parts into LOYTEC Device User Manual and LINX Configurator User Manual.