Electric Drives and Controls | Hydraulics | Linear Motion and Assembly Technologies | Pneumatics | Service

Rexroth Bosch Group

Rexroth IndraLogic L20 03VRS System Description

R911312329
Edition 02

Operating And Programming Guide

Title
Rexroth IndraLogic L20 03VRS
System Description

Type of Documentation
Operating and Programming Guide

Document Typecode
DOK-CONTRL-IC*L20*****-AW02-EN-P

Internal File Reference
Document Number, 120-0401-B325-02/EN

Purpose of Documentation
This documentation describes provides an overview of the system components belonging to the IndraLogic L20 control and describes their projecting and programming.

Record of Revisions

Description	Release Date	Notes
120-0401-B325-01/EN	08/05	First Edition
120-0401-B325-02/EN	11/06	Second Edition

Validity
The specified data is for product description purposes only and may not be deemed to be guaranteed unless expressly confirmed in the contract. All rights are reserved with respect to the content of this documentation and the availability of the product.

Published by
Bosch Rexroth AG
Bgm.-Dr.-Nebel-Str. 2 • D-97816 Lohr a. Main
Telephone +49 (0)93 52/40-0 • Tx 68 94 21 • Fax +49 (0)93 52/40-48 85
http://www.boschrexroth.com/
Dept. BRC/EPY (NH)

Note
This document has been printed on chlorine-free bleached paper.

Contents

1 System Overview

1.1 Components

Control

The control IndraLogic L20 consists of:

- Hardware: IndraControl L20 incl. a connector set
- Firmware: IndraLogic L20 (on memory card)

The modular and scalable hardware platform IndraControl L20 can be used in combination with the IndraLogic L20 firmware for PLC applications.

Software

The software "IndraWorks Logic" serves to commission and project the IndraLogic L20. It consists of the following components:

- **IndraWorks:** Project planning, configuration
- **IndraLogic:** PLC programming
- **IndraWorks HMI:** Visualization and user interface as well as diagnostic functions (ProVi)
- **IndraWorks WinStudio:** Engineering tool to create user screens for IndraWorks HMI
- **IndraLogic L20 TSP:** Target files (Target Support Package) to edit the IndraLogic L20 with IndraWorks and IndraLogic
- **Target Manager:** Management of TSP files, e. g., while updating control functions.

All components are automatically installed.

1.2 Related Documentation

No.	Title	Identification
/1/	Rexroth IndraControl L20; Project Planning Manual	DOK-CONTRL-IC*L20*****-PR..-EN-P
/2/	PLC Programming with Rexroth IndraLogic; Operating and Programming Guide	DOK-CONTRL-IL**PRO*V02-AW..-EN-P
/3/	Rexroth Inline Profibus DP; Application Manual	DOK-CONTRL-R-IL*PBSSYS-AW..-EN-P
/4/	Rexroth Inline Profibus DP Terminal and Module Supply; Functional Description	DOK-CONTRL-R-IL*PB*-BK-FK..-EN-P
/5/	Rexroth IndraWorks Engineering; Operating and Programming Guide	DOK-IWORKS-ENGINEE*V..-AW..-EN-P
/6/	Rexroth WinStudio; Overall View	DOK-CONTRL-WIS*PC**V06-KB..-EN-P
/7/	Rexroth PLCopen Function Blocks For Field Bus Drives; Application Manual	DOK-CONTRL-PLCOPENFB*D-AW..-EN-P

Fig. 1-1: Related documentation

2 Important Directions for Use

2.1 Appropriate Use

Introduction

Rexroth products represent state-of-the-art developments and manufacturing. They are tested prior to delivery to ensure operating safety and reliability.

The products may only be used in the manner that is defined as appropriate. If they are used in an inappropriate manner, then situations can develop that may lead to property damage or injury to personnel.

Note: Bosch Rexroth, as manufacturer, is not liable for any damages resulting from inappropriate use. In such cases, the guarantee and the right to payment of damages resulting from inappropriate use are forfeited. The user alone carries all responsibility of the risks.

Before using Rexroth products, make sure that all the pre-requisites for appropriate use of the products are satisfied:

- Personnel that in any way, shape or form uses our products must first read and understand the relevant safety instructions and be familiar with appropriate use.
- If the product takes the form of hardware, then they must remain in their original state, in other words, no structural changes are permitted. It is not permitted to decompile software products or alter source codes.
- Do not mount damaged or faulty products or use them in operation.
- Make sure that the products have been installed in the manner described in the relevant documentation.

Areas of Use and Application

The IndraLogic L20 of Rexroth is a compact small control including a standardized I/O system on the basis of terminal technology and is designed for logic applications.

Note: The IndraLogic L20 may only be used with the accessories and parts specified in this documentation (see "Related Documentation"). If a component has not been specifically named, then it may not be either mounted or connected. The same applies to cables and lines.

Operation is only permitted in the specified configurations and combinations of components using the software and firmware as specified in the relevant function descriptions.

In case of non-observance the warranty claim expires automatically.

Typical applications of the IndraLogic L20 are:

- Handling and assembly systems,
- Packaging and foodstuff machine,
- Printing and paper processing machines
- Machine tools.

The IndraLogic L20 may only be operated under the assembly, installation and ambient conditions (temperature, system of protection, humidity, EMC requirements, etc.) as described in document /1/ and in the position specified.

In residential areas as well as in business and commercial areas Class A devices may be used with the following note:

Note: This is a Class A device. In a residential area, this device may cause radio interferences. In such a case, the user may be required to introduce suitable countermeasures at his own cost.

2.2 Inappropriate Use

Using the IndraLogic L20 outside of the above-referenced areas of application or under operating conditions other than described in the document and the technical data specified is defined as “inappropriate use".

The IndraLogic L20 may not be used, if

- they are subject to operating conditions, that do not meet the above specified ambient conditions. This includes, for example, operation under water, in the case of extreme temperature fluctuations or extremely high maximum temperatures, or if
- Bosch Rexroth has not specifically released them for that intended purpose. Please note the specifications outlined in the general Safety Guidelines!

3 Safety Instructions for Electric Drives and Controls

3.1 Introduction

Read these instructions before the initial startup of the equipment in order to eliminate the risk of bodily harm or material damage. Follow these safety instructions at all times.

Do not attempt to install or start up this equipment without first reading all documentation provided with the product. Read and understand these safety instructions and all user documentation of the equipment prior to working with the equipment at any time. If you do not have the user documentation for your equipment, contact your local Bosch Rexroth representative to send this documentation immediately to the person or persons responsible for the safe operation of this equipment.

If the equipment is resold, rented or transferred or passed on to others, then these safety instructions must be delivered with the equipment.

WARNING

Improper use of this equipment, failure to follow the safety instructions in this document or tampering with the product, including disabling of safety devices, may result in material damage, bodily harm, electric shock or even death!

3.2 Explanations

The safety instructions describe the following degrees of hazard seriousness in compliance with ANSI Z535. The degree of hazard seriousness informs about the consequences resulting from non-compliance with the safety instructions.

Warning symbol with signal word	Degree of hazard seriousness according to ANSI
 **DANGER**	Death or severe bodily harm will occur.
 **WARNING**	Death or severe bodily harm may occur.
 **CAUTION**	Bodily harm or material damage may occur.

Fig. 3-1: Hazard classification (according to ANSI Z535)

3.3 Hazards by Improper Use

DANGER

High voltage and high discharge current! Danger to life or severe bodily harm by electric shock!

DANGER

Dangerous movements! Danger to life, severe bodily harm or material damage by unintentional motor movements!

WARNING

High electrical voltage due to wrong connections! Danger to life or bodily harm by electric shock!

WARNING

Health hazard for persons with heart pacemakers, metal implants and hearing aids in proximity to electrical equipment!

CAUTION

Surface of machine housing could be extremely hot! Danger of injury! Danger of burns!

CAUTION

Risk of injury due to improper handling! Bodily harm caused by crushing, shearing, cutting and mechanical shock or incorrect handling of pressurized systems!

CAUTION

Risk of injury due to incorrect handling of batteries!

3.4 General Information

- Bosch Rexroth AG is not liable for damages resulting from failure to observe the warnings provided in this documentation.
- Read the operating, maintenance and safety instructions in your language before starting up the machine. If you find that you cannot completely understand the documentation for your product, please ask your supplier to clarify.
- Proper and correct transport, storage, assembly and installation as well as care in operation and maintenance are prerequisites for optimal and safe operation of this equipment.
- Only persons who are trained and qualified for the use and operation of the equipment may work on this equipment or within its proximity.
  - The persons are qualified if they have sufficient knowledge of the assembly, installation and operation of the equipment as well as an understanding of all warnings and precautionary measures noted in these instructions.
  - Furthermore, they must be trained, instructed and qualified to switch electrical circuits and equipment on and off in accordance with technical safety regulations, to ground them and to mark them according to the requirements of safe work practices. They must have adequate safety equipment and be trained in first aid.
- Only use spare parts and accessories approved by the manufacturer.
- Follow all safety regulations and requirements for the specific application as practiced in the country of use.
- The equipment is designed for installation in industrial machinery.
- The ambient conditions given in the product documentation must be observed.
- Use only safety features and applications that are clearly and explicitly approved in the Project Planning Manual.
  For example, the following areas of use are not permitted: construction cranes, elevators used for people or freight, devices and vehicles to transport people, medical applications, refinery plants, transport of hazardous goods, nuclear applications, applications sensitive to high frequency, mining, food processing, control of protection equipment (also in a machine).
- The information given in the documentation of the product with regard to the use of the delivered components contains only examples of applications and suggestions.
  The machine and installation manufacturer must
  - make sure that the delivered components are suited for his individual application and check the information given in this documentation with regard to the use of the components,
  - make sure that his application complies with the applicable safety regulations and standards and carry out the required measures, modifications and complements.
- Startup of the delivered components is only permitted once it is sure that the machine or installation in which they are installed complies with the national regulations, safety specifications and standards of the application.
- Technical data, connections and operational conditions are specified in the product documentation and must be followed at all times.

- Operation is only permitted if the national EMC regulations for the application are met.
  The instructions for installation in accordance with EMC requirements can be found in the documentation "EMC in Drive and Control Systems".
  The machine or installation manufacturer is responsible for compliance with the limiting values as prescribed in the national regulations.

3.5 Protection Against Contact with Electrical Parts

Note: This section refers to equipment and drive components with voltages above 50 Volts.

Touching live parts with voltages of 50 Volts and more with bare hands or conductive tools or touching ungrounded housings can be dangerous and cause electric shock. In order to operate electrical equipment, certain parts must unavoidably have dangerous voltages applied to them.

DANGER

High electrical voltage! Danger to life, severe bodily harm by electric shock!

⇒ Only those trained and qualified to work with or on electrical equipment are permitted to operate, maintain or repair this equipment.

⇒ Follow general construction and safety regulations when working on high voltage installations.

⇒ Before switching on power the ground wire must be permanently connected to all electrical units according to the connection diagram.

⇒ Do not operate electrical equipment at any time, even for brief measurements or tests, if the ground wire is not permanently connected to the points of the components provided for this purpose.

⇒ Before working with electrical parts with voltage higher than 50 V, the equipment must be disconnected from the mains voltage or power supply. Make sure the equipment cannot be switched on again unintended.

⇒ The following should be observed with electrical drive and filter components:

⇒ Wait five (5) minutes after switching off power to allow capacitors to discharge before beginning to work. Measure the voltage on the capacitors before beginning to work to make sure that the equipment is safe to touch.

⇒ Never touch the electrical connection points of a component while power is turned on.

⇒ Install the covers and guards provided with the equipment properly before switching the equipment on. Prevent contact with live parts at any time.

⇒ A residual-current-operated protective device (RCD) must not be used on electric drives! Indirect contact must be prevented by other means, for example, by an overcurrent protective device.

⇒ Electrical components with exposed live parts and uncovered high voltage terminals must be installed in a protective housing, for example, in a control cabinet.

To be observed with electrical drive and filter components:

DANGER

High electrical voltage on the housing!
High leakage current! Danger to life, danger of injury by electric shock!

⇒ Connect the electrical equipment, the housings of all electrical units and motors permanently with the safety conductor at the ground points before power is switched on. Look at the connection diagram. This is even necessary for brief tests.

⇒ Connect the safety conductor of the electrical equipment always permanently and firmly to the supply mains. Leakage current exceeds 3.5 mA in normal operation.

⇒ Use a copper conductor with at least 10 mm² cross section over its entire course for this safety conductor connection!

⇒ Prior to startups, even for brief tests, always connect the protective conductor or connect with ground wire. Otherwise, high voltages can occur on the housing that lead to electric shock.

3.6 Protection Against Electric Shock by Protective Low Voltage (PELV)

All connections and terminals with voltages between 0 and 50 Volts on Rexroth products are protective low voltages designed in accordance with international standards on electrical safety.

WARNING

High electrical voltage due to wrong connections! Danger to life, bodily harm by electric shock!

⇒ Only connect equipment, electrical components and cables of the protective low voltage type (PELV = Protective Extra Low Voltage) to all terminals and clamps with voltages of 0 to 50 Volts.

⇒ Only electrical circuits may be connected which are safely isolated against high voltage circuits. Safe isolation is achieved, for example, with an isolating transformer, an opto-electronic coupler or when battery-operated.

3.7 Protection Against Dangerous Movements

Dangerous movements can be caused by faulty control of the connected motors. Some common examples are:

- improper or wrong wiring of cable connections
- incorrect operation of the equipment components
- wrong input of parameters before operation
- malfunction of sensors, encoders and monitoring devices
- defective components
- software or firmware errors

Dangerous movements can occur immediately after equipment is switched on or even after an unspecified time of trouble-free operation.

The monitoring in the drive components will normally be sufficient to avoid faulty operation in the connected drives. Regarding personal safety, especially the danger of bodily injury and material damage, this alone cannot be relied upon to ensure complete safety. Until the integrated monitoring functions become effective, it must be assumed in any case that faulty drive movements will occur. The extent of faulty drive movements depends upon the type of control and the state of operation.

DANGER

Dangerous movements! Danger to life, risk of injury, severe bodily harm or material damage!

⇒ Ensure personal safety by means of qualified and tested higher-level monitoring devices or measures integrated in the installation. Unintended machine motion is possible if monitoring devices are disabled, bypassed or not activated.

⇒ Pay attention to unintended machine motion or other malfunction in any mode of operation.

⇒ Keep free and clear of the machine's range of motion and moving parts. Possible measures to prevent people from accidentally entering the machine's range of motion:

- use safety fences
- use safety guards
- use protective coverings
- install light curtains or light barriers

⇒ Fences and coverings must be strong enough to resist maximum possible momentum, especially if there is a possibility of loose parts flying off.

⇒ Mount the emergency stop switch in the immediate reach of the operator. Verify that the emergency stop works before startup. Don't operate the machine if the emergency stop is not working.

⇒ Isolate the drive power connection by means of an emergency stop circuit or use a starting lockout to prevent unintentional start.

⇒ Make sure that the drives are brought to a safe standstill before accessing or entering the danger zone. Safe standstill can be achieved by switching off the power supply contactor or by safe mechanical locking of moving parts.

⇒ Secure vertical axes against falling or dropping after switching off the motor power by, for example:
- mechanically securing the vertical axes
- adding an external braking/ arrester/ clamping mechanism
- ensuring sufficient equilibration of the vertical axes

The standard equipment motor brake or an external brake controlled directly by the drive controller are not sufficient to guarantee personal safety!

⇒ Disconnect electrical power to the equipment using a master switch and secure the switch against reconnection for:
- maintenance and repair work
- cleaning of equipment
- long periods of discontinued equipment use

⇒ Prevent the operation of high-frequency, remote control and radio equipment near electronics circuits and supply leads. If the use of such equipment cannot be avoided, verify the system and the installation for possible malfunctions in all possible positions of normal use before initial startup. If necessary, perform a special electromagnetic compatibility (EMC) test on the installation.

3.8 Protection Against Magnetic and Electromagnetic Fields During Operation and Mounting

Magnetic and electromagnetic fields generated near current-carrying conductors and permanent magnets in motors represent a serious health hazard to persons with heart pacemakers, metal implants and hearing aids.

WARNING

Health hazard for persons with heart pacemakers, metal implants and hearing aids in proximity to electrical equipment!

⇒ Persons with heart pacemakers, hearing aids and metal implants are not permitted to enter the following areas:
- Areas in which electrical equipment and parts are mounted, being operated or started up.
- Areas in which parts of motors with permanent magnets are being stored, operated, repaired or mounted.

⇒ If it is necessary for a person with a heart pacemaker to enter such an area, then a doctor must be consulted prior to doing so. Heart pacemakers that are already implanted or will be implanted in the future, have a considerable variation in their electrical noise immunity. Therefore there are no rules with general validity.

⇒ Persons with hearing aids, metal implants or metal pieces must consult a doctor before they enter the areas described above. Otherwise, health hazards will occur.

3.9 Protection Against Contact with Hot Parts

CAUTION

Housing surfaces could be extremely hot! Danger of injury! Danger of burns!

⇒ Do not touch housing surfaces near sources of heat! Danger of burns!

⇒ After switching the equipment off, wait at least ten (10) minutes to allow it to cool down before touching it.

⇒ Do not touch hot parts of the equipment, such as housings with integrated heat sinks and resistors. Danger of burns!

3.10 Protection During Handling and Mounting

Under certain conditions, incorrect handling and mounting of parts and components may cause injuries.

CAUTION

Risk of injury by incorrect handling! Bodily harm caused by crushing, shearing, cutting and mechanical shock!

⇒ Observe general installation and safety instructions with regard to handling and mounting.

⇒ Use appropriate mounting and transport equipment.

⇒ Take precautions to avoid pinching and crushing.

⇒ Use only appropriate tools. If specified by the product documentation, special tools must be used.

⇒ Use lifting devices and tools correctly and safely.

⇒ For safe protection wear appropriate protective clothing, e.g. safety glasses, safety shoes and safety gloves.

⇒ Never stand under suspended loads.

⇒ Clean up liquids from the floor immediately to prevent slipping.

3.11 Battery Safety

Batteries contain reactive chemicals in a solid housing. Inappropriate handling may result in injuries or material damage.

CAUTION

Risk of injury by incorrect handling!

⇒ Do not attempt to reactivate discharged batteries by heating or other methods (danger of explosion and cauterization).

⇒ Never charge non-chargeable batteries (danger of leakage and explosion).

⇒ Never throw batteries into a fire.

⇒ Do not dismantle batteries.

⇒ Do not damage electrical components installed in the equipment.

Note: Be aware of environmental protection and disposal! The batteries contained in the product should be considered as hazardous material for land, air and sea transport in the sense of the legal requirements (danger of explosion). Dispose batteries separately from other waste. Observe the legal requirements in the country of installation.

3.12 Protection Against Pressurized Systems

Certain motors and drive controllers, corresponding to the information in the respective Project Planning Manual, must be provided with pressurized media, such as compressed air, hydraulic oil, cooling fluid and cooling lubricant supplied by external systems. Incorrect handling of the supply and connections of pressurized systems can lead to injuries or accidents. In these cases, improper handling of external supply systems, supply lines or connections can cause injuries or material damage.

CAUTION

Danger of injury by incorrect handling of pressurized systems !

⇒ Do not attempt to disassemble, to open or to cut a pressurized system (danger of explosion).

⇒ Observe the operation instructions of the respective manufacturer.

⇒ Before disassembling pressurized systems, release pressure and drain off the fluid or gas.

⇒ Use suitable protective clothing (for example safety glasses, safety shoes and safety gloves)

⇒ Remove any fluid that has leaked out onto the floor immediately.

Note: Environmental protection and disposal! The media used in the operation of the pressurized system equipment may not be environmentally compatible. Media that are damaging the environment must be disposed separately from normal waste. Observe the legal requirements in the country of installation.

Notes

4 Projecting and Programming

4.1 Requirements and Overview of the Proceeding

Requirements

- The IndraWorks Logic package is installed.
- The target information (Target Support Package) of the IndraLogic L20 are available in IndraWorks. All relevant data are automatically generated during the installation of IndraWorks.
- IndraWorks Engineering is started.
- An IndraWorks project exists and is indicated in the project explorer.

 For detailed information on the user interface and the handling of IndraWorks refer to the IndraWorks documentation /5/ or online help.

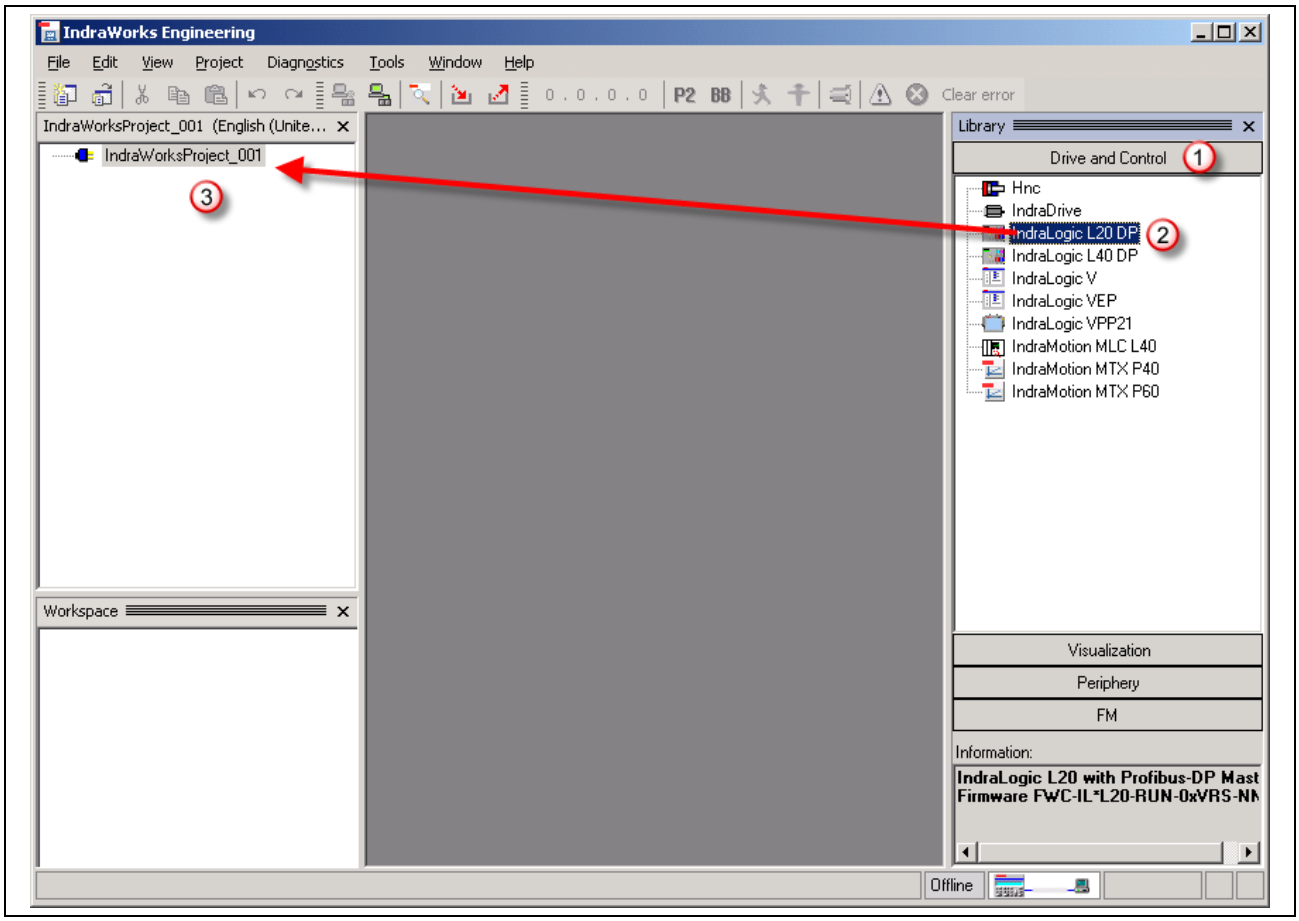
Required Steps

1. Creating a new IndraLogic L20 device with IndraWorks
2. Defining basic settings with IndraWorks
3. Defining field bus and I/O configuration with IndraWorks
4. PLC programming with IndraLogic
5. Loading configuration and PLC program data into the hardware of the IndraLogic L20 and activating them with IndraWorks

4.2 Creating New IndraLogic L20 Device

Drag the device "IndraLogic L20 DP" from the "Drive and Control" library into the desired project in the project explorer. Alternatively, you can also use the **Copy** and **Insert** function of the particular context menu (right mouse button).

For further information refer to the documentation or online help of IndraWorks /5/.



(1) "Drive and Control" library
(2) "IndraLogic L20 DP" device
(3) Project explorer

Fig. 4-1: Inserting the IndraLogic L20 DP into the project explorer

Then, a wizard starts automatically to define the necessary basic settings.

4.3 Defining Basic Settings (Wizard-Guided)

Device Settings

Enter the general data of the inserted device.

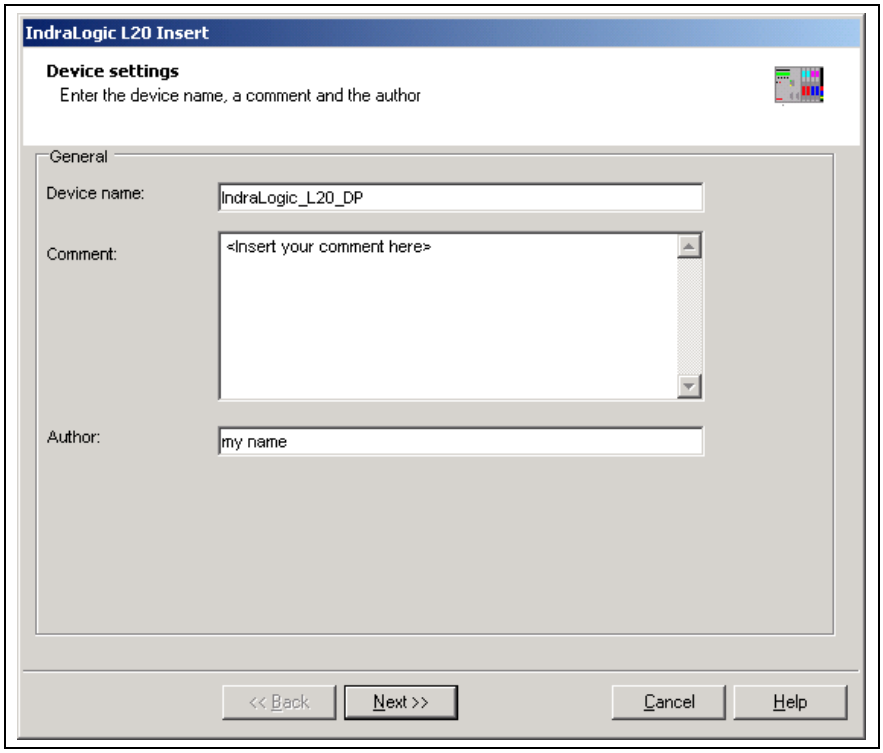


Fig. 4-2: Device settings

Device Name Please enter here any device name. The field contains by default the name of the library. With this name the project appears later in the project explorer.

Comment Please enter here any comment to describe the device in detail.

Author Please enter here the name of the project author.

Note: These inputs can also be changed later, for this see section "Modifying Basic Settings" on page 4-10.

Confirm the settings with "Next >>".

Device and Function Module Settings

Select the firmware used for the device. Depending on the selected firmware additional settings concerning the existing field bus interfaces (Profibus, Ethernet/IP) can be made.

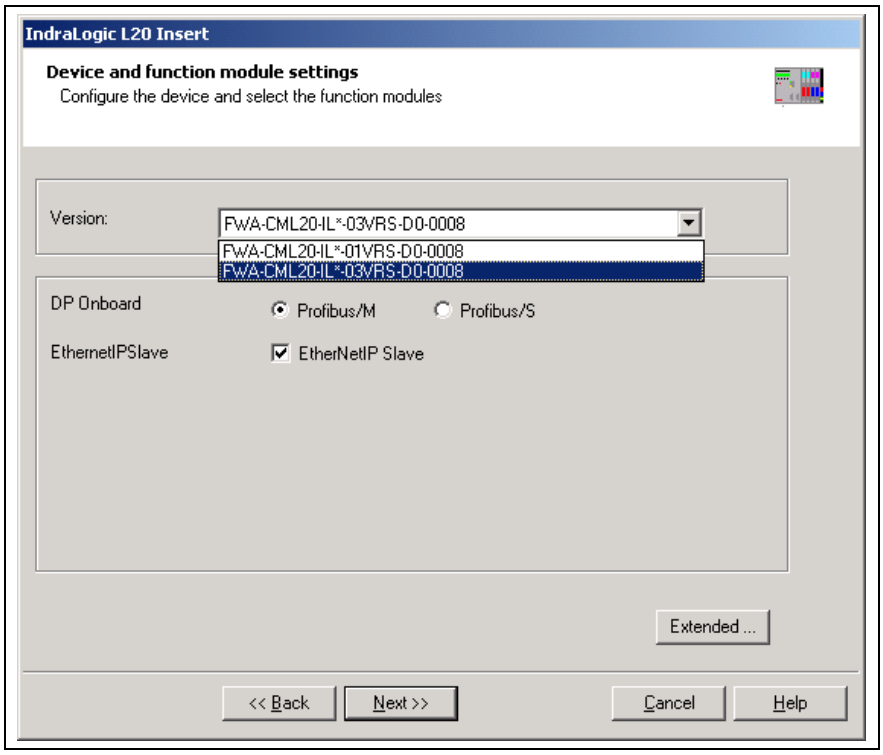


Fig. 4-3: Enter the firmware version and the settings for the field bus interfaces

Note: After inserting the new IndraLogic L20 device into the project explorer "Version" and "DP Onboard" settings can no longer be modified!

Version Select here the **firmware version** of the target device.
This also affects further settings provided in this dialog.

DP Onboard "**Profibus/M**": Via the onboard Profibus DP interface (X7P) subordinated Profibus DP slaves are to be coupled to the IndraLogic L20. From the view of the slave connected there the IndraLogic L20 is its master.

"**Profibus/S**": Via the onboard Profibus DP interface (X7P) the IndraLogic L20 is to be coupled to a superordinated Profibus DP master as subordinated Profibus DP slave.

Ethernet/IP Slave Activate this option to use the IndraLogic L20 via the onboard Ethernet interface (X7E) as follows:

- As **CIP data server** (CIP: Communication Industrial Protocol): Allows the simultaneous access for a maximum of 6 CIP data clients to enabled variables of the IndraLogic L20 via acyclic DataTable services ("Unconnected Explicit Messaging" and "Class 3 Connected Messaging").
- As **CIP data client**: Allows the access to enabled variables via the "RIL_EtherNetIP" library of a CIP data server via acyclic DataTable services.
- As **Ethernet/IP slave** (target): Allows the cyclic exchange of I/O data blocks with a superordinated control (master/originator) via an "Exclusive Owner Connection", (Transport Class 1) and a further "Listen Only Connection", (Transport Class 1).

Type of Addressing

Click on the **Extended...** button to set the desired I/O addressing type (byte or word).

Note: Basic settings of the I/O addressing type:

- lower than firmware version 03 VRS: word
- higher than firmware version 03 VRS: bytes

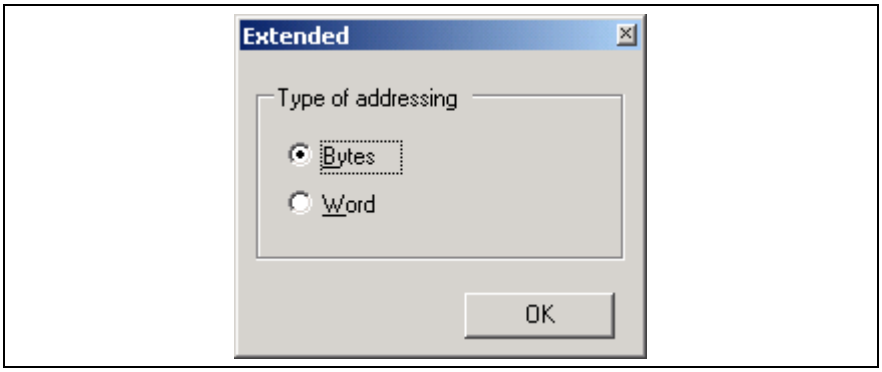


Fig. 4-4: Setting the I/O addressing type:

Select the setting which corresponds to the type of addressing in the PLC programs to be used. Word addressing requires that, for example, output byte 10 (QB10) is addressed by "QW5". By contrast, byte addressing requires that output byte 10 (QB10) is addressed by "QW10".

Note: After inserting the new IndraLogic L20 device into the project explorer the type of addressing can no longer be modified!

Confirm the settings with "Next >>".

The dialog to define the communication settings opens.

Communication Settings

Define here the settings for the communication between IndraWorks and the IndraLogic L20. When opening this dialog the IndraLogic gateway server used to enable the communication starts automatically. If the gateway server is already provided with communication settings, it is possible to select a setting via the "Channels" dropdown list.

Note: If there is no entry yet, a dialog to create a communication channel appears automatically. If there's no suitable entry, you can create a new or modify an already existing communication channel. Click on the **Communication parameters...** button.

The further proceeding is described in section "Communication Parameters" on page 4-7.

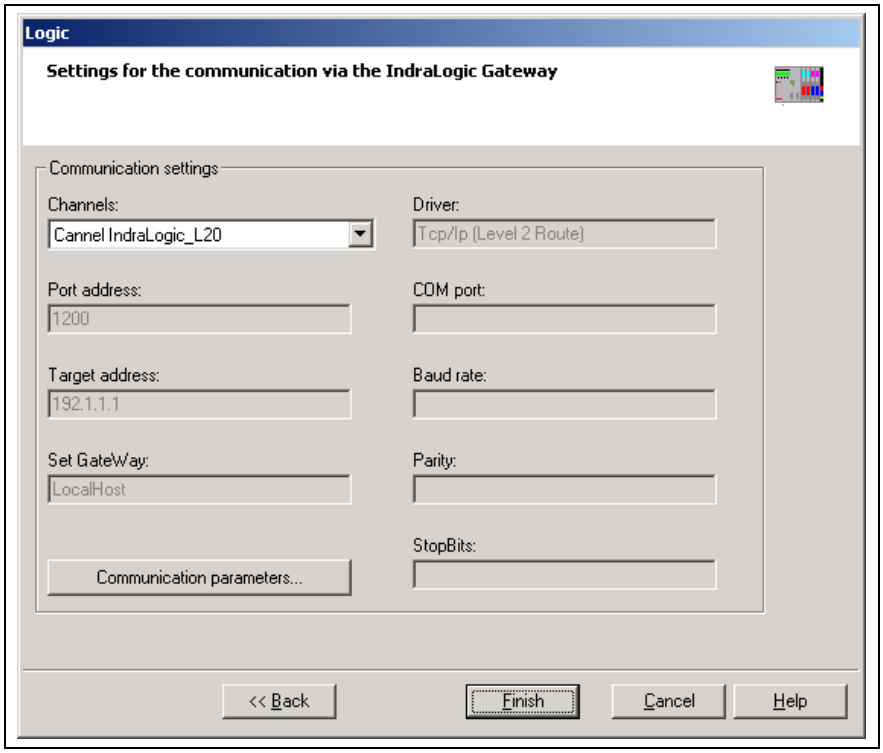


Abb. 4-5: Communication settings of the IndraLogic L20

The boxes highlighted in gray indicate the parameters of the entry selected under "Channels".

The communication settings must correspond to the actual settings of the IndraLogic L20, see also section "Default Menu" on page 7-5.

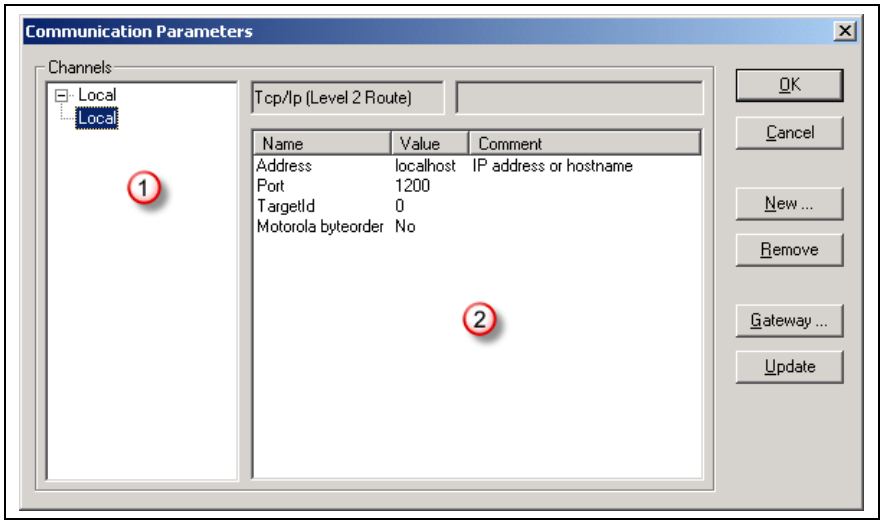
You will find further information about the topic "Communication" in the IndraLogic help, or in the IndraLogic manual /2/.

Finish the settings by pressing the "Finish" button. With this action, the wizard-guided basic settings are terminated. The new IndraLogic L20 device appears in the project explorer. You can change the basic settings subsequently, see section "Modifying Basic Settings" on page 4-10.

For more detailed information on the device please refer to section "Components of the IndraLogic L20 Device" on page 4-9.

Communication Parameters

Create here new communication channels (communication instances), or change or delete already existing channels.



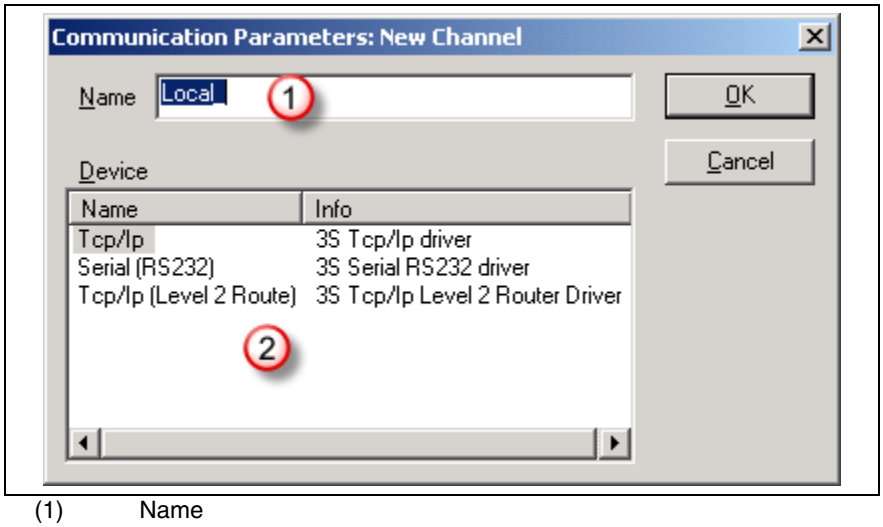
(1) Communication channels
(2) Parameters of the selected communication channel

Fig. 4-6: Communication parameters

Create a new communication channel:

Click on the **New...** button to create a new communication channel.

The following dialog appears:



(1) Name
(2) Device

Fig. 4-7: Create a new communication channel

Name Please enter here a name for the new communication channel.

Device Select the desired communication driver to communicate between IndraWorks and the IndraLogic L20. For this, click in the "Name" column on the corresponding line and confirm the settings with the **OK** button.

The following communication drivers are available:

- **Tcp/Ip**: communication via the Ethernet interface (X7E) via TCP/IP protocol
- **Serial (RS232)**: communication via the RS232 interface (X3C)

- **Tcp/Ip (Level 2 Route)**: communication via the Ethernet interface (X7E) via TCP/IP protocol. Difference to **Tcp/Ip**: extended package size. If the communication is realized via the Ethernet interface, you should use the **Tcp/Ip (Level 2 Route)** driver for performance reasons.

The new communication channel appears in area 1 in Fig. 4-8.

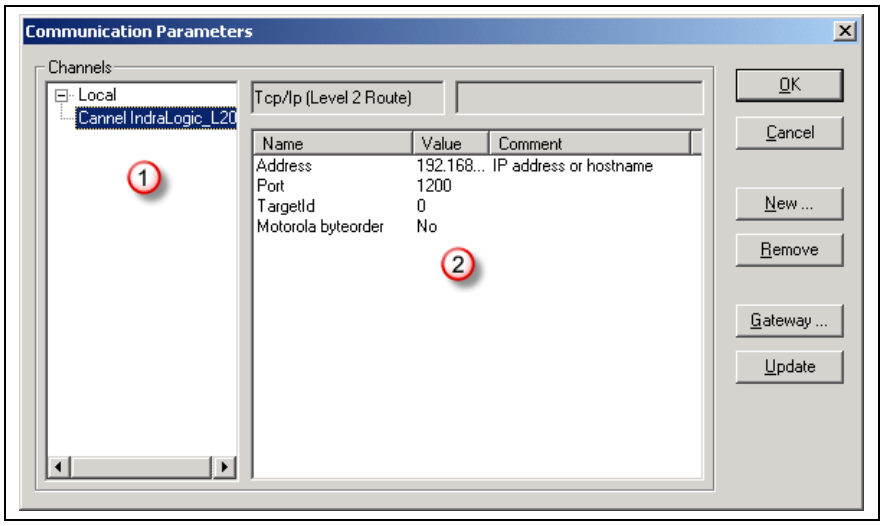


Fig. 4-8: Adapt the parameters of a communication channel

Modify a communication channel:

Select the new communication channel with the left mouse button and adapt its parameters. For this, double-click on the corresponding line in area 2 (see Fig. 4-8) and change the entry in the "Value" column according to your requirements. Confirm the modifications with the <Enter> key.

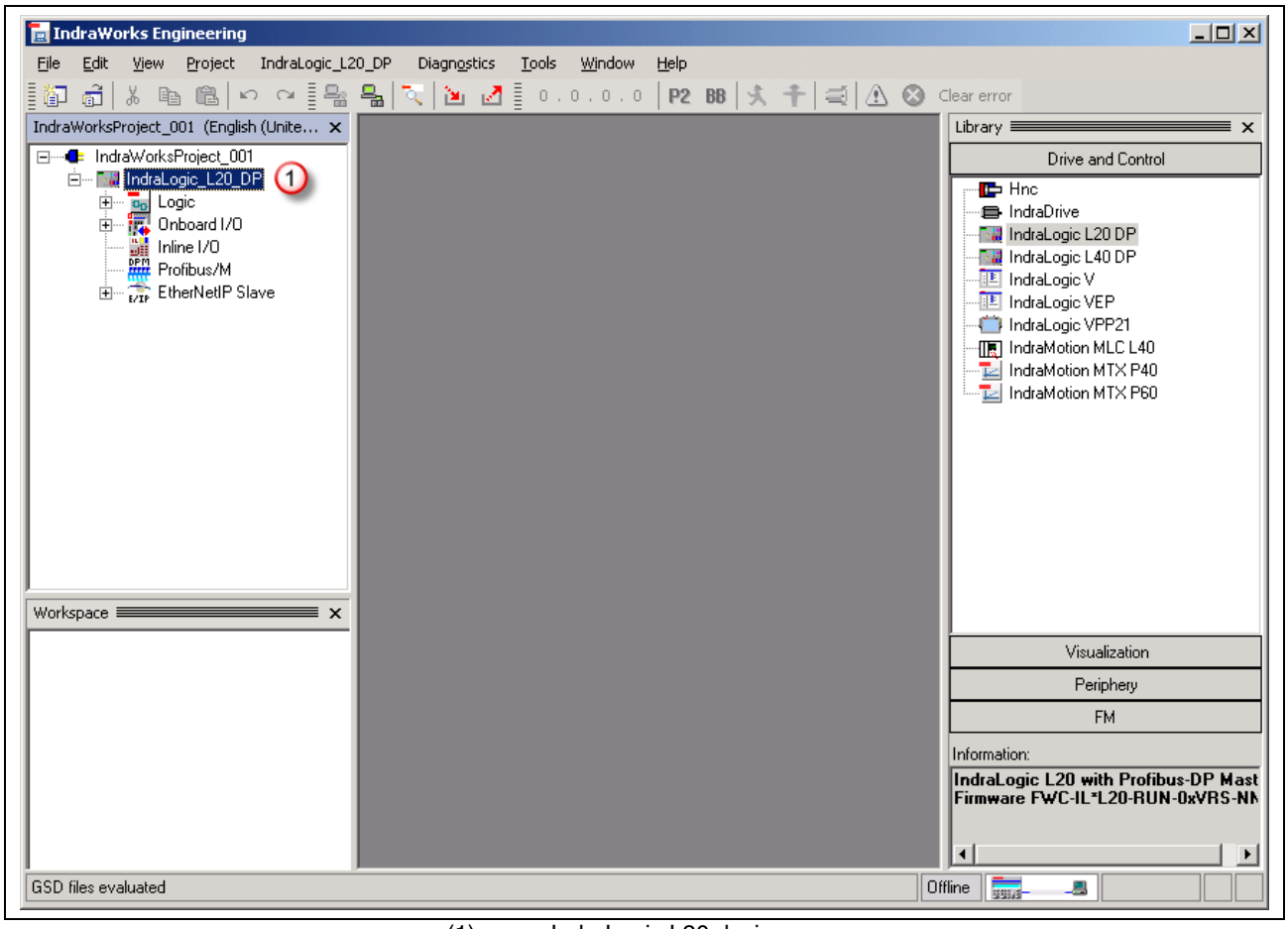
Delete a communication channel:

Select the desired communication channel and click on the **Delete** button.

You will find further information on the individual parameters in the IndraLogic help, or in the IndraLogic manual /2/.

Components of the IndraLogic L20 Device

To display all subordinated objects of the new IndraLogic L20 device, click on the plus symbol of the object tree in the project explorer.



(1) IndraLogic L20 device

Fig. 4-9: IndraLogic L20 device with subordinated objects

The object tree of the IndraLogic L20 can contain the following objects:

- "**Logic**": part of the IndraLogic L20 device responsible for processing by programming of all I/O signals. This includes e. g. PLC modules (POUs), the PLC task management and global variables. A double-click on the "Logic" object starts IndraLogic with the "IndraLogic L20" target system.
- "**Onboard I/O**": local I/O area of the IndraLogic L20 (Onboard I/O). By default, this area is equipped with 8 high-speed inputs and 8 high-speed outputs. The configuration is described in chapter "Configuring Onboard I/O" as of page 4-12.
- "**Inline I/O**": local Inline I/O area of the IndraLogic L20. It contains the I/Os of Rexroth Inline modules that can be coupled to the IndraLogic L20. The configuration is described in chapter "Projecting Inline Modules" as of page 4-16.
- "**Profibus/M**": only available, if the onboard Profibus DP interface (X7P) has been configured as "Profibus/M" (Profibus master), see Fig. 4-3. Important Profibus operating data can be parameterized and Profibus slaves can be integrated with this object; for this, see chapter "Configuring IndraLogic L20 as Profibus DP Master" as of page 4-19.

- "**Profibus/S**": only available, if the onboard Profibus DP interface (X7P) has been configured as "Profibus/S" (Profibus slave), see Fig. 4-3. With this object the Profibus slave settings of the IndraLogic L20 can be configured, see chapter "Configuring IndraLogic L20 as Profibus DP Slave" as of page 4-30.
- "**EthernetIP Slave**": only available, if the IndraLogic L20 has been configured as Ethernet/IP slave, see Fig. 4-3. With this object the Profibus slave settings of the IndraLogic L20 can be configured, see chapter "Configuring IndraLogic L20 as Ethernet/IP Slave" as of page 4-35.

Inserting an "EthernetIP Slave" Object Subsequently

Note: To insert an "EthernetIP Slave" object subsequently is only possible, if there's no "EthernetIP Slave" in the object tree of the IndraLogic L20!

Drag the "Ethernet IP Slave" object from the "FM" (function modules) library into the object tree of the IndraLogic L20. Alternatively, you can also use the **Copy** and **Insert** function of the particular context menu (right mouse button).

Modifying Basic Settings

Some of the settings generated with the help of the wizard can also be modified subsequently.

Modify Device Settings

Use the "Properties" menu item in the context menu of the IndraLogic L20 device.

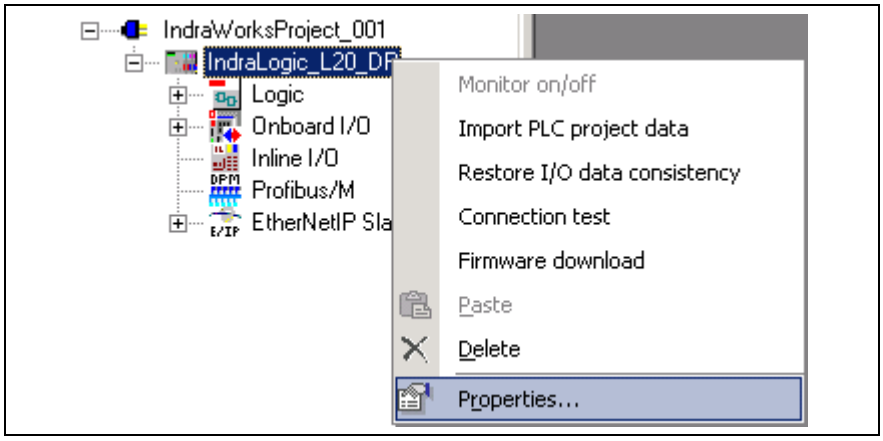


Fig. 4-10: Context menu for the IndraLogic L20

The dialog to modify the device settings appears, see Fig. 4-2.

Modifying Communication Settings

Use the "Properties" menu item in the context menu of the "Logic" object.

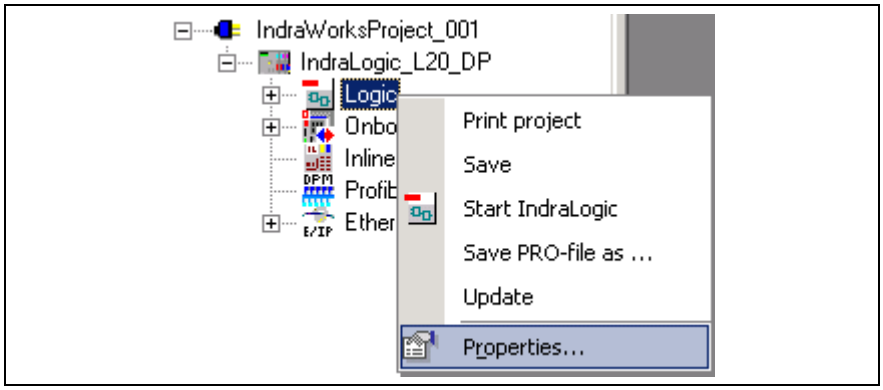


Fig. 4-11: Context menu for the "Logic" object

The following dialog opens:

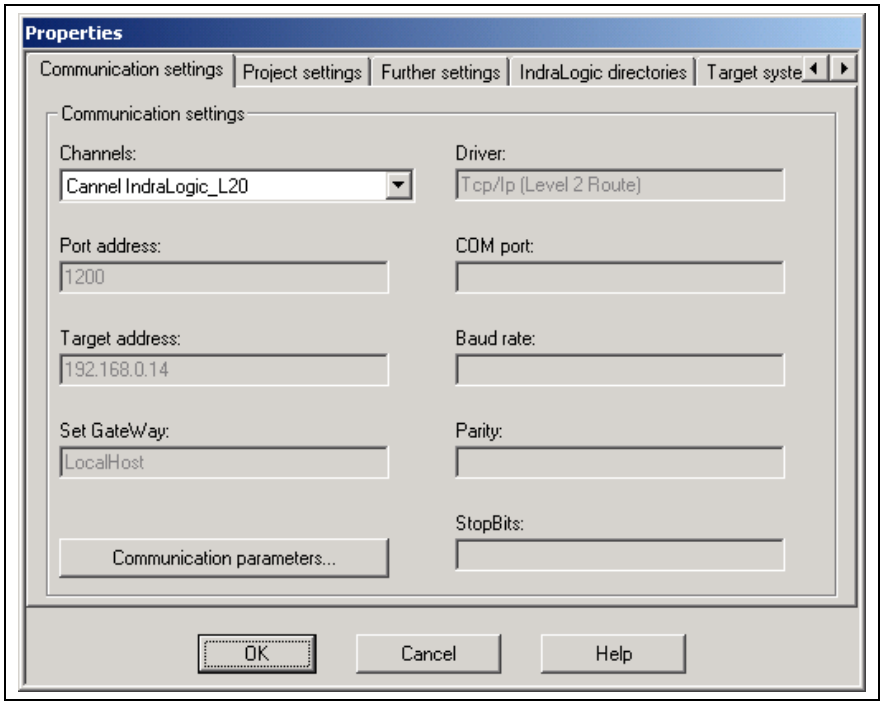


Fig. 4-12: "Properties" dialog of the "Logic" object

For this, observe section "Communication Settings" on page 4-6.

Besides the communication settings the tabs provide also further settings. For further information refer to the documentation or online help of IndraWorks /5/.

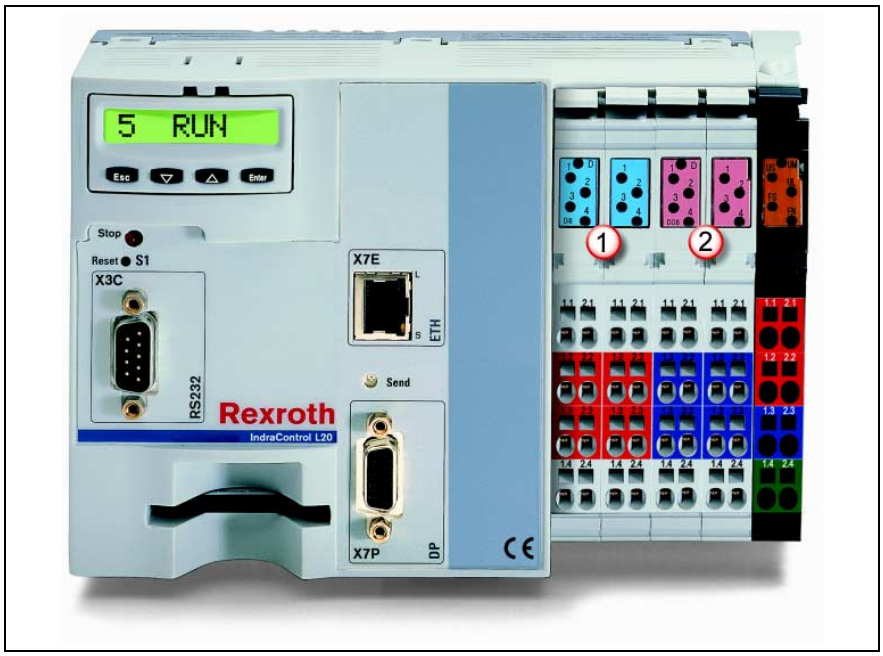
4.4 Configuring Onboard I/O

Overview

By default, the IndraLogic L20 is provided with eight high-speed, interrupt-compliant digital inputs and eight high-speed, interrupt-compliant digital outputs.

Concerning the interrupt capability please consider section "Task Configuration" as of page 4-45.

An I/O extension can be realized by using Inline modules (see chapter "Projecting Inline Modules" as of page 4-16) and/or by coupling subordinated slaves (e. g. Profibus DP) to the IndraLogic L20.



(1) Inputs
(2) Outputs

Fig. 4-13: Onboard I/O of the IndraLogic L20

The digital inputs and outputs available on the slots 1 to 4 (from the left to the right) are assigned to light-emitting diodes and bit addresses according to the following table:

		Inputs								Outputs							
	Slot	1				2				3				4			
	Status LED	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Byte-bit view	Bytes	IX0.0 – 0.7 (default)								QX0.0 – 0.7 (default)							
	Bit	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
Module	Terminal point (signal)	1.1	2.1	1.4	2.4	1.1	2.1	1.4	2.4	1.1	2.1	1.4	2.4	1.1	2.1	1.4	2.4
	Terminal point (24 V)	1.2	2.2	1.3	2.3	1.2	2.2	1.3	2.3	-	-	-	-	-	-	-	-
	Terminal point (last ground)	-	-	-	-	-	-	-	-	1.2	2.2	1.3	2.3	1.2	2.2	1.3	2.3

Fig. 4-14: Default address assignment of inputs and outputs

To configure the addresses double-click on "Onboard I/O" in the project explorer.

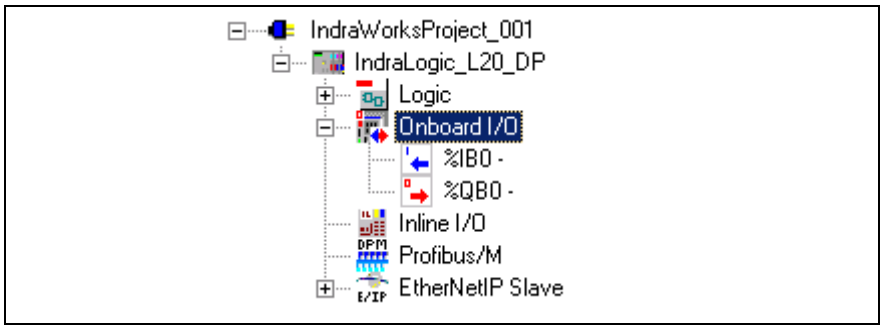


Fig. 4-15: "Onboard I/O" object

A window opens in the workspace:

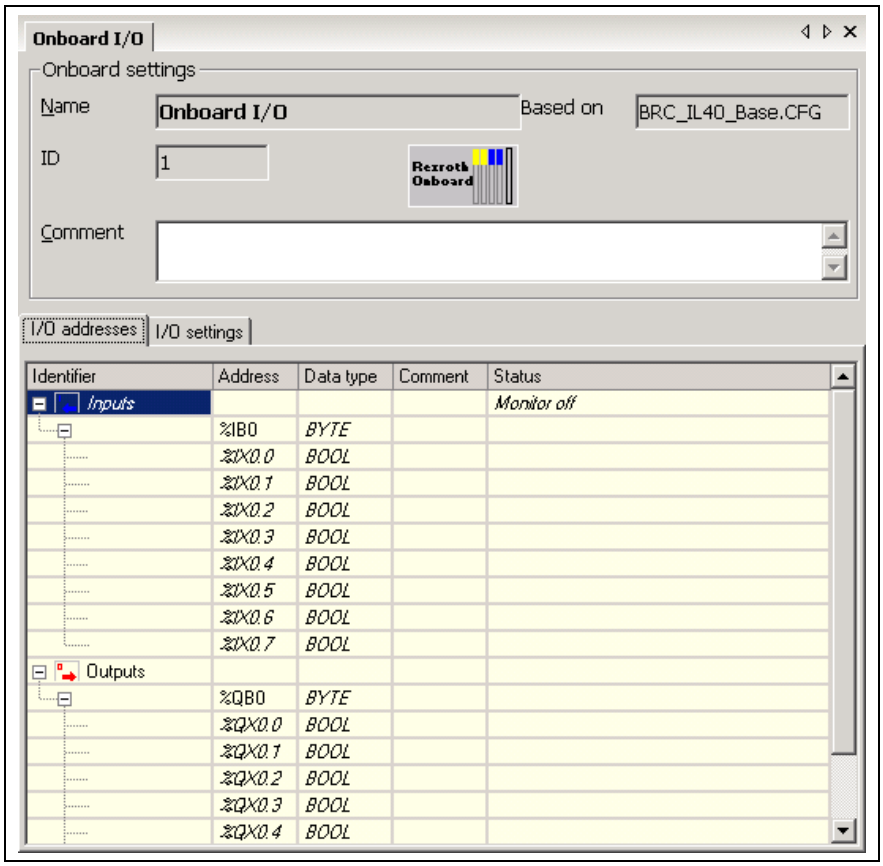


Fig. 4-16: "Onboard I/O" window

"Onboard Settings"

Name: Internal name, specified by the installed target device.

Based on: Name of the used device description file. The file is a fixed part of the installed target system.

ID: Internal name, specified by the installed target system.

Comment: Please enter here any comment to describe the onboard I/O in detail.

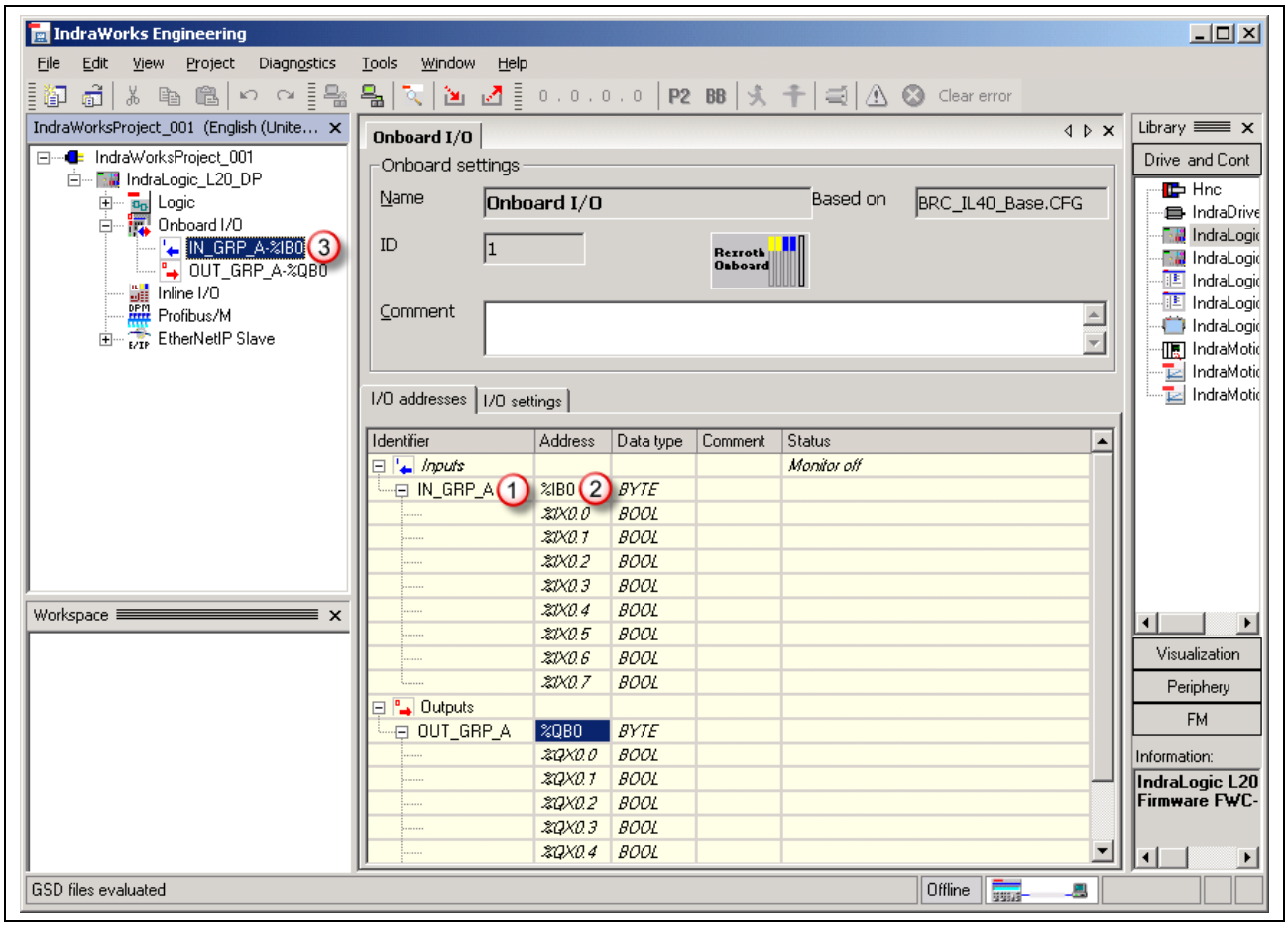
"I/O Addresses" Tab

Assign here the I/O areas of the onboard I/O to the physical addresses of the control (I/O addresses of the PLC).

Identifier: This column shows the two input and output modules. You can change between the byte and bit view with the plus and minus symbol.

For each absolute address the assignment of a symbolic address is possible (double-click on the particular field). After entering the symbolic address, it is created automatically as global variable in the PLC project.

The symbolic address of a node also appears in the project explorer:



(1) Symbolic address
(2) Absolute address
(3) Resulting entry in the project explorer

Fig. 4-17: Symbolic and absolute addresses of I/O objects

Address: I/O address. Enter the desired I/O address as byte address (e. g. %IB10). Entries in italics serve only for display and cannot be edited.

Note: An automatic readdressing is possible in the "I/O settings" tab.

Data type: Byte addresses are identified with "BYTE", bit addresses with "BOOL".

Comment: Please enter here any comment about an address.

Status: Physical status of the input/output. The status is only indicated in diagnostic mode during the communication between IndraWorks and IndraLogic L20.

"I/O Settings" Tab

Start here the automatic assignment of I/O addresses.

From output / **From input**: Current or desired start addresses of the outputs and inputs.

"**Apply**": Renumbers automatically all outputs/inputs of the onboard I/O in ascending order starting with the indicated start addresses (see "From output / From input"). Please note, that existing address gaps are closed during this process!

If the automatic numbering causes collisions with already assigned address areas, IndraWorks indicates the collision cause and determines automatically the next free address area.

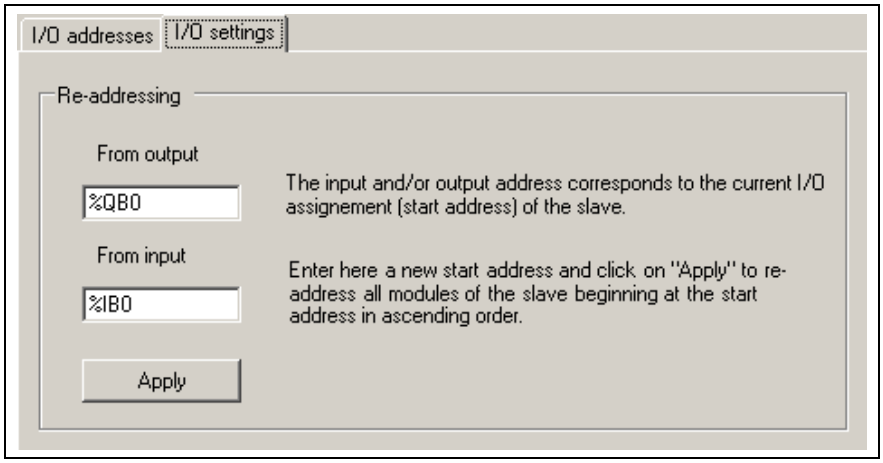


Fig. 4-18: Readdressing the onboard I/O

4.5 Projecting Inline Modules

Overview

The locally available I/O units can be extended by the Rexroth Inline I/O system just by simply mounting the components at the right side of the IndraLogic L20.

Main features of the Inline module system of the IndraLogic L20:

- Extending the local I/O field to up to 32 bytes (total of all inputs and outputs)
- Coupling up to 63 Inline modules to the IndraLogic L20 in any address-independent order. The actual usable number of Inline modules also depends on the current consumption of the single modules, see documentation /1/.

Inserting Inline Modules

All Inline modules provided for the IndraLogic L20 are in the "Periphery" library under "Inline", "Rexroth Inline". Drag the required Inline modules from the library into the corresponding "Inline I/O" object. In the project explorer new Inline modules can also be inserted between already existing Inline modules.

Alternatively, you can also use the **Add module** function in the context menu of the "Inline I/O" object. The new module is inserted as last module under "Inline I/O".

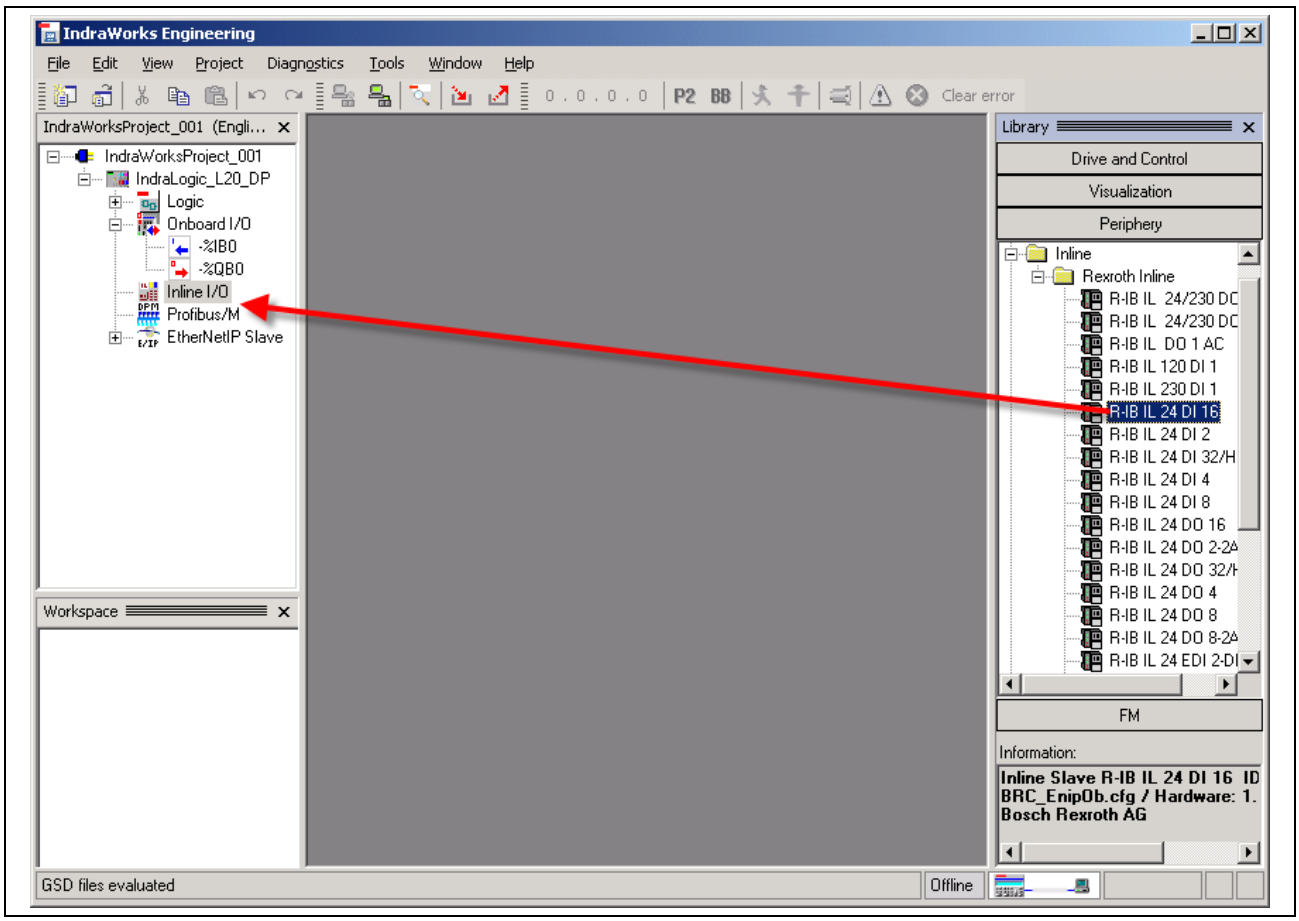


Fig. 4-19: Insert Inline module (example)

Configuring Inline Modules

Double-click in the project explorer on the desired Inline module.

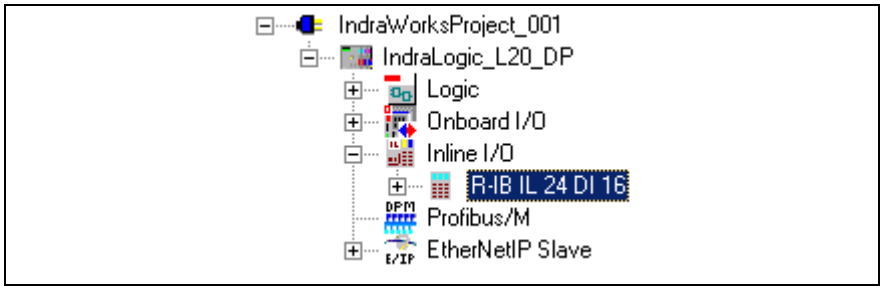


Fig. 4-20: Inline module object (example)

A window opens in the workspace:

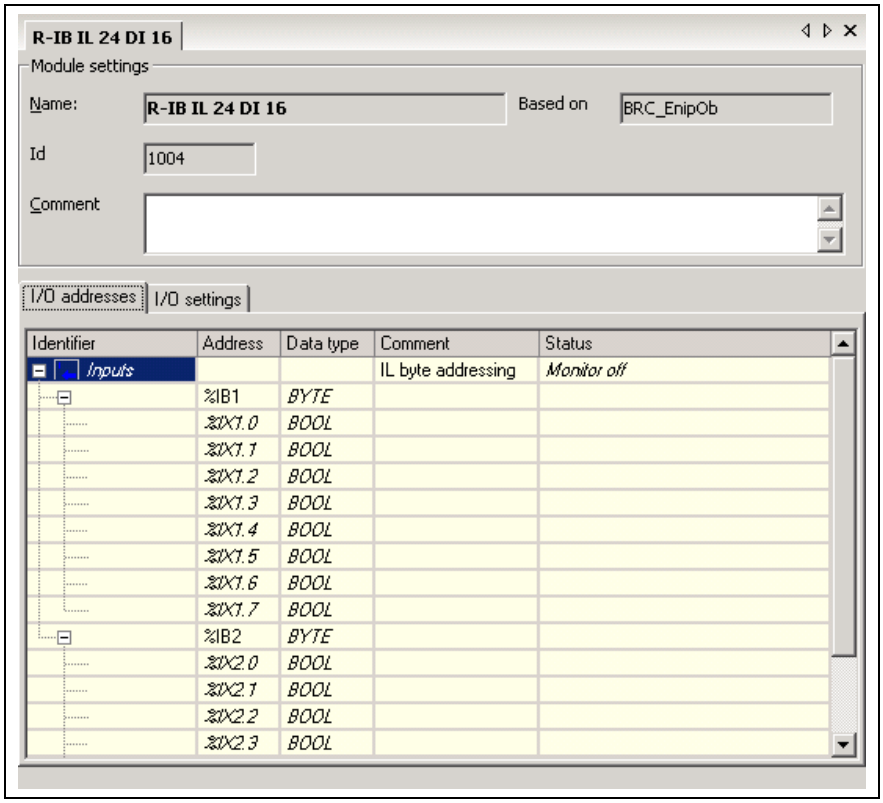


Fig. 4-21: Window of an Inline module (example)

"Module Settings"

Name: Internal name, specified by the installed target device.

Based on: Path and name of the used device description file. The file is a fixed part of the installed target system.

ID: Internal name, specified by the installed target system.

Comment: Please enter here any comment to describe the Inline module in a detailed manner.

"I/O Addresses" Tab

Assign here the I/O areas of the Inline modules to the physical addresses of the control (I/O addresses of the PLC).

Identifier: This column shows the input and output structure. You can change between the byte and bit view with the plus and minus symbol.

For each absolute address the assignment of a symbolic address is possible (double-click on the particular field). After entering the symbolic address, it is created automatically as global variable in the PLC project.

The symbolic address of a node also appears in the project explorer. You will find an example for onboard I/O in Fig. 4-17.

Address: I/O address. Enter the desired I/O address as byte address (e. g. %IB10). Entries in italics serve only for display and cannot be edited.

Note: An automatic readdressing is possible in the "I/O settings" tab.

Data type: Byte addresses are identified with "BYTE", bit addresses with "BOOL".

Comment: Please enter here any comment about an address.

Status: Physical status of the input/output. The status is only indicated in diagnostic mode during the communication between IndraWorks and IndraLogic L20.

"I/O Settings" Tab

Start here the automatic assignment of I/O addresses.

From output / **From input**: Current or desired start addresses of the outputs or inputs. Corresponding to the used Inline module functionality (module with inputs, module with outputs) only relevant input boxes are indicated. If you parameterize, for example, an Inline module with only inputs, the input box "From output" is missing.

"**Apply**": Renumbers automatically all outputs/inputs of the Inline module in ascending order starting with the indicated start addresses (see "From output / From input"). Please note, that existing address gaps are closed during this process!

If the automatic numbering causes collisions with already assigned address areas, IndraWorks indicates the collision cause and determines automatically the next free address area.

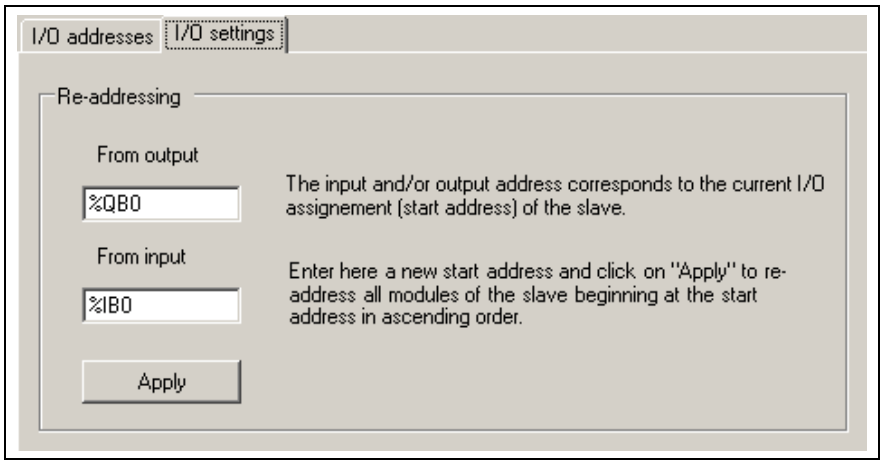


Fig. 4-22: Readdressing an Inline module

4.6 Configuring IndraLogic L20 as Profibus DP Master

Overview

The IndraLogic L20 is equipped with a Profibus DP interface (X7P) with bus master functionality according to DIN EN 50170, part 2. For this reason, the connection of Profibus DP slaves and the access on their I/O fields is not possible.

To use the IndraLogic L20 as Profibus DP master please execute the following steps:

1. Define the IndraLogic L20 as Profibus DP-Master, see section "Device and Function Module Settings" on page 4-4. Only in this case the required subordinated "Profibus/M" object is available in the project explorer.
2. Modifying master-specific settings.
3. Inserting Profibus DP slave in the "Profibus/M" object in the project explorer.
4. Configuring inserted Profibus DP slaves.

Note: The IndraLogic L20 can also be operated via the onboard Profibus DP interface (X7P) as subordinated Profibus DP slave. For this, observe chapter "Configuring IndraLogic L20 as Profibus DP Slave" as of page 4-30.

Modifying Master-Specific Settings

Double-click in the project explorer on the relevant "Profibus/M" object.

A window opens in the workspace:

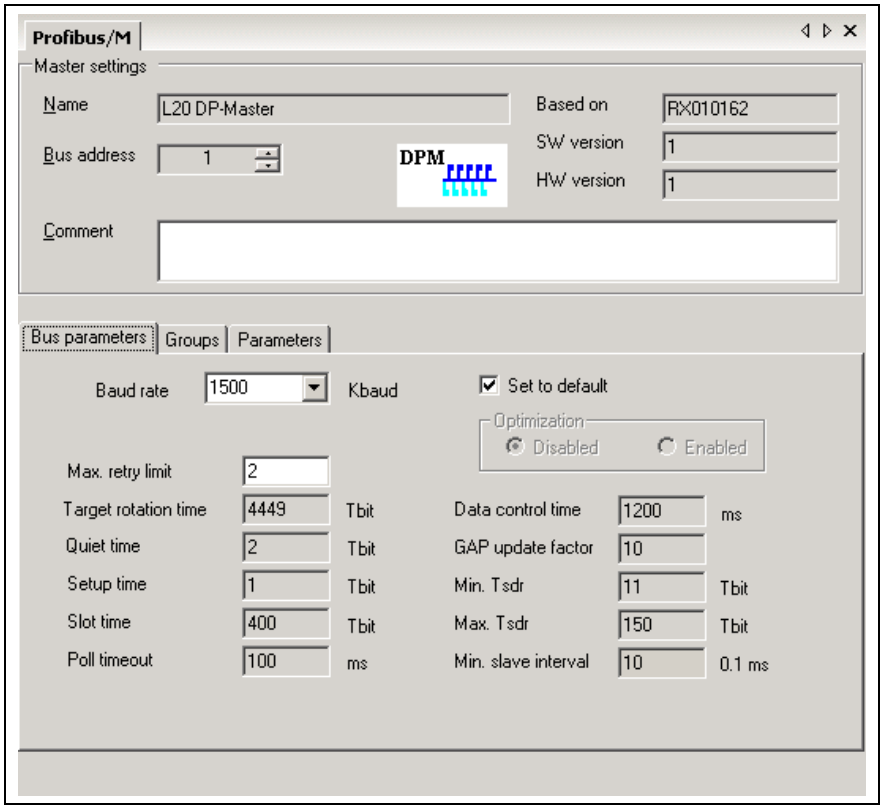


Fig. 4-23: "Profibus/M" window

Master Settings

Name: Internal device name of the bus master

Bus address: Bus address of the bus master (FDL address: Fieldbus Data Link). Address "1" is automatically entered. If necessary, enter here another address. Address 0 is reserved for engineering devices and cannot be used.

Note: Always use address values for the master being as low as possible. High address values deteriorate the bus performance!

Comment: Please enter here any comment to describe the bus master in a detailed manner.

Based on: Ident number according to the GSD file.

SW version: Software version according to the GSD file.

HW version: Hardware version according to the GSD file.

"Bus Parameter" Tab

Contains the bus parameters required for the operation of the Profibus DP. If the "**Set to default**" option is activated, only the boxes "Baud rate" and "Max. retry limit" can be edited. All other parameters are adapted to the currently defined baud rate and suitable for the majority of applications.

To change the values, deactivate the "**Set to default**" option and set the "**Optimization**" option to "**Enabled**". To disable the input boxes, you can reset the "**Optimization**" to "**Disabled**". As soon as the "**Set to default**" option is activated, the values entered before are overwritten by the adapted values.

CAUTION

Modifications of the bus parameters might cause an unpredictable system behavior!

⇒ Thus, the bus parameters may only be modified by instructed Profibus DP specialists, who are aware of the effects of this modifications!

Baud rate (transmission rate): Select here the data transmission rate of the entire bus system. All connected slaves have to support the specified value. The maximum baud rate might not exceed the highest possible baud rate of the "worst" slave.

Max. Retry Limit: Maximum number of repetitions of a call telegram by the initiator, if the responder (receiver) does not send any answer. Possible settings: 1 to 15.

Target Rotation Time (Ttr): Command token rotation time for the multi-master operation. Possible settings: 256 to 6647 Tbits (bit time units).

Quiet Time (Tqui): Modulator quiet time or repeater rotation time. During this time it is waited for "Quietness at the bus". Telegrams are neither sent nor received. Possible settings: 0 to 255 Tbits.

Setup Time (Tset): Release time. Maximum time passing by before an event arrives up to the execution of the required reaction. Possible settings: 1 to 255 Tbits.

Slot Time (Tsl): "Waiting for reception" time. Maximum time that the initiator is waiting for the immediate acknowledgement or response after emitting a call telegram (message cycle). Because of the direct influence of Tid2 the setting can be selected between Tid2 + 15 and 16383 Tbits.

Poll Timeout: Time monitoring for an acyclic service (DPV1). Maximum time that the initiator is waiting for the reception of the response after emitting an acyclic request.

The setting can be selected in steps of 10 ms in the range of 10 to 655350 ms.

Data_Control_Time: Slave-related monitoring time in the bus master; within this time period at least one user data transfer with the corresponding slave has to be carried out. Possible settings: 1 to 65535 ms

GAP Update Factor: Factor with which it is possible to control after how many bus cycles the new master can be recognized.

Possible settings: 1 to 10

Min. Tsdr: This is the time a responder needs at least to answer a request telegram.

Possible settings: 11 to 255 Tbits

Min. Tsdr: This is the time a responder needs at most to answer a request telegram.

Possible settings: 35 to 1023 Tbits

Min_Slave_Intervall: This is the minimum time period that has to pass by between to accesses of the bus master to the same slave.

Possible settings: 1 to 65535 (factor: 100 µs)

"Groups" tab

The group assignment refers to the sync and freeze commands of the "Global Control" services of Profibus DP. Activate in column "Sync" or "Freeze" the group(s) to which the master should send the freeze or sync commands.

Example: With the group assignment from the following figure the bus master is allowed to send the sync command to slaves of group 3 and 4 and the freeze commands to slaves of group 1 and 3.

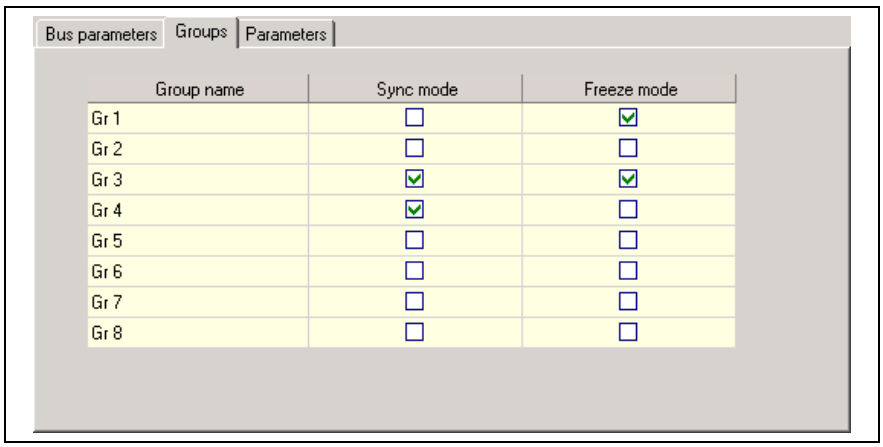


Fig. 4-24: "Profibus/M" window, "Groups" tab

Note: To assign slaves to a certain group, see section "Configuring Profibus DP Slaves", "Group assignment" tab on page 4-26.

Note: Control commands of the "sync" and "freeze" modes can be generated with function block DP_SYCFR (see section "DP_SYCFR" on page 6-37). If you specify in this connexion slaves of a "disabled" group, the master already freezes the transmission of the commands to the relevant slaves.

"Parameters" Tab Displaying and editing possibly available, vendor-specific bus master parameters. The integrated bus master of the IndraLogic L20 does not have any specific parameters. For that reason, this tab remains empty.

Inserting Profibus DP Slaves

All Profibus DP slaves provided for the IndraLogic L20 are in the "Periphery" library under "Profibus DP". Drag the required slaves from the library into the "Profibus/M" object. New slaves can be also inserted in the project explorer between already existing slaves.

Alternatively, you can also use the **Add slave** function in the context menu of the "Profibus/M" object, see Fig. 4-25. The new slave is inserted as last slave under "Profibus/M".

Note: If a required slave is not available in the library by default, it can be integrated in the library by importing its GSD file via the **Import GSD file...** function in the context menu of the "Profibus/M" object.

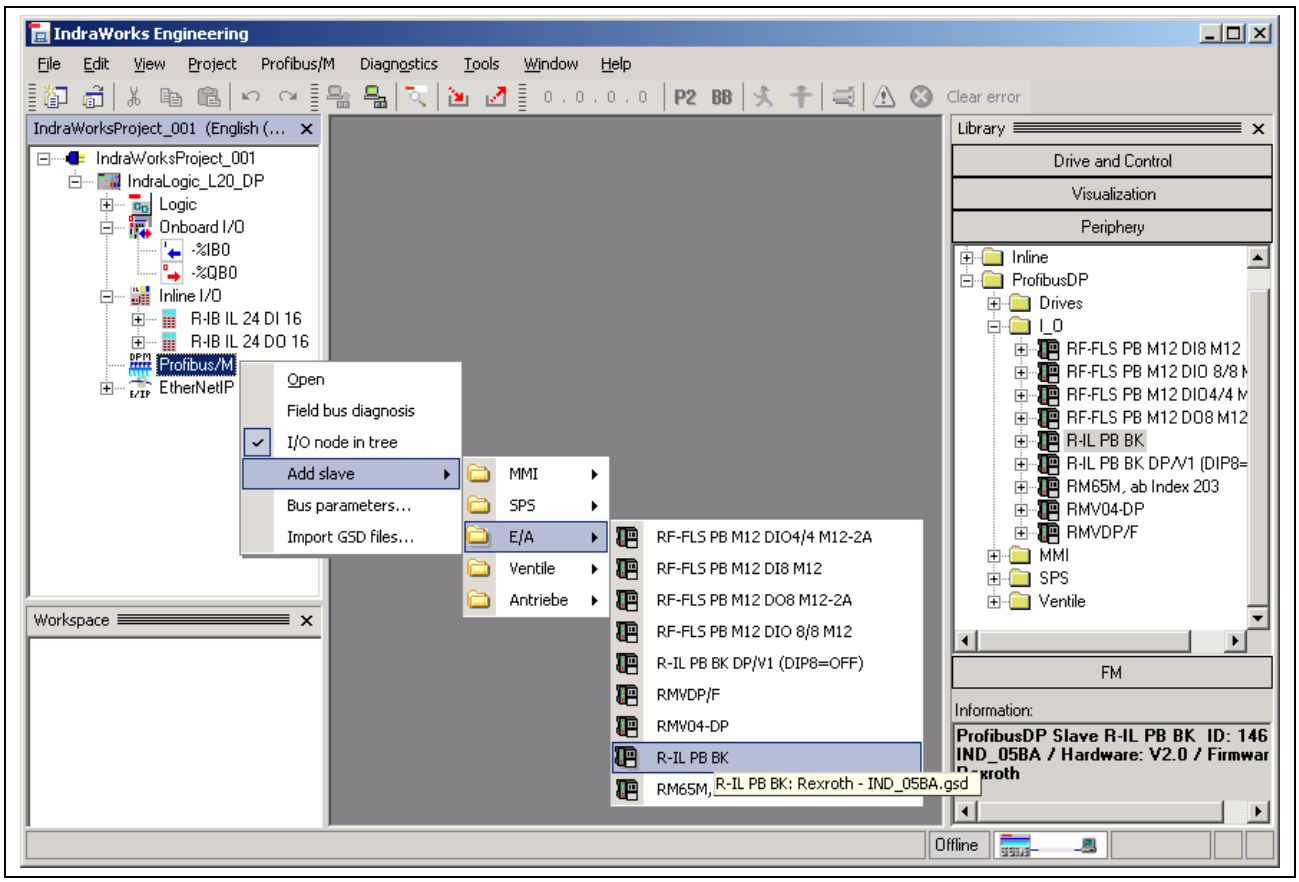
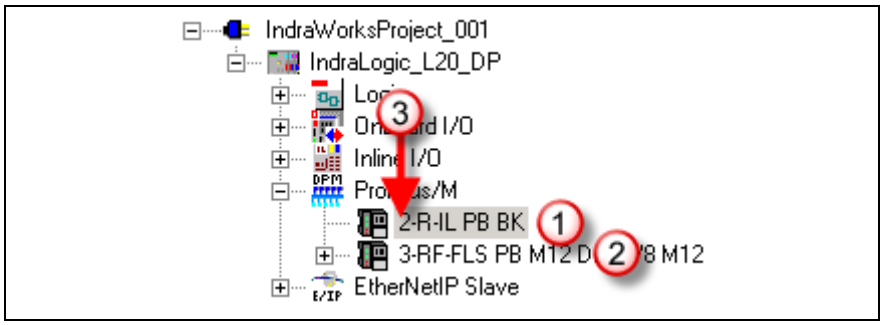


Fig. 4-25: Inserting a Profibus DP slave via the context menu of the "Profibus/M" object

Configuring Profibus DP Slaves

Profibus DP distinguishes between two slave types:

- **Compact**: A compact slave has a firmly defined module structure. After inserting a slave in the project explorer the modules below the slave object node are already completely available because of the compact design of the slaves.
- **Modular**: The module structure of the slave is variable. The modules can be arranged individually, but according to the fitting specification. Directly after inserting the slave in the project explorer no subordinated (I/O) device levels of the slave are available. For modular slaves the modules have to be manually assigned. To insert modules, see section "Inserting Modules in the Profibus DP Slave" on page 4-27.



(1) Modular Profibus DP slave; still without subordinated modules
(2) Compact Profibus DP slave
(3) Current bus addresses of the Profibus DP slaves

Fig. 4-26: Profibus DP slave (example)

To change the bus address of a slave, open the "DP Bus addresses" dialog via context menu item **Bus address**. Here, the complete address assignment of all Profibus DP devices is indicated. With a double-click on a free field in the "Status" table column the corresponding bus address is applied for the currently selected slave.

Note: Always use address values for the master being as low as possible. High address values deteriorate the bus performance!

To configure a Profibus DP slave, double-click on the corresponding slave entry in the project explorer. A window opens in the workspace, see the figure on the following page:

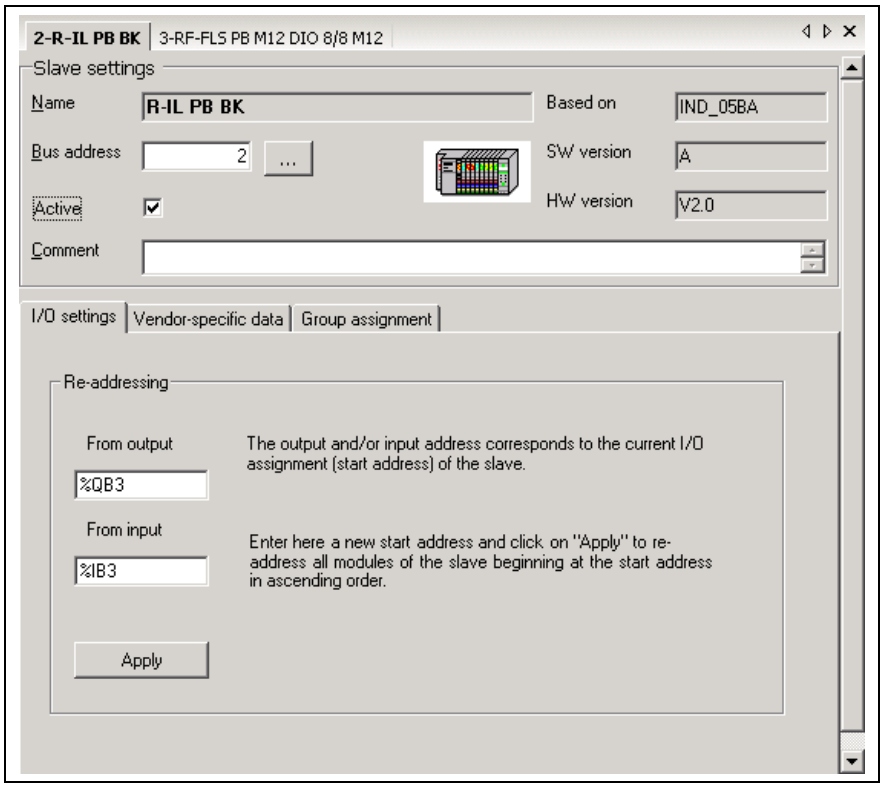


Fig. 4-27: Window of Profibus DP slave (example)

"Slave Settings"

Name: Slave name according to the GSD file.

Bus address: Bus address of the slave (FDL address). Here, IndraWorks enters the next free bus address automatically. If necessary, enter here another address. Alternatively, you can open the "**DP bus addresses**" dialog with the **"..."** button. Here, the complete address assignment of all Profibus DP devices is indicated. With a double-click on a free field in the "Status" table column the corresponding bus address is applied for the currently selected slave.

Please note the following restrictions for the address selection:

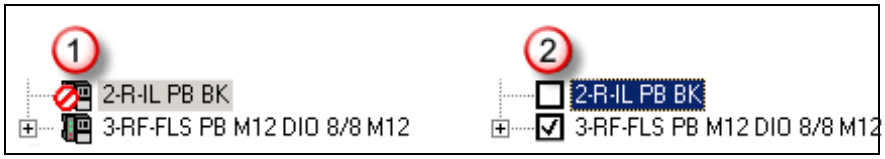
Address "0": reserved for engineering devices

Address "1": reserved for the Profibus DP master

The bus address of the slave also appears in the project explorer, see Fig. 4-26.

Active: Activate this option, so that the slave can be commissioned at the Profibus after the next program download (see chapter "Download and Commissioning" as of page 4-53). Deactivate this option, if the slave is to be configured and archived, but not to be commissioned at the Profibus.

The setting (active/not active) can be also recognized or made in the project explorer. If the mouse pointer is positioned on the slave, the setting can be changed by a left mouse-click, see the following figure (2).



(1) Display of the settings "not active" (first line) and "active" (second line).

(2) Display of the settings, if the mouse pointer is positioned on the slave.

Fig. 4-28: Setting "active"/"not active" in the project explorer

Based on: Ident number according to the GSD file.

SW version: Software version according to the GSD file.

HW version: Hardware version according to the GSD file.

Comment: Please enter here any comment to describe the slave in a detailed manner.

"I/O Settings" Tab

Start here the automatic assignment of I/O addresses.

From output / **From input**: Desired start addresses for the automatic numbering of the outputs and inputs of all subordinated modules of the Profibus DP slave.

"**Apply**": Renumbers automatically all outputs/inputs of the subordinated Profibus DP slave modules in ascending order starting with the indicated start addresses (see "From output / From input"). Please note, that existing address gaps are closed during this process!

If the automatic numbering causes collisions with already assigned address areas, IndraWorks indicates the collision cause and determines automatically the next free address area.

Note: The automatic numbering of subordinated modules can also be made in the module configuration. For this, observe section "Inserting Modules in the Profibus DP Slave" on page 4-27.

"Vendor-Specific Data" Tab

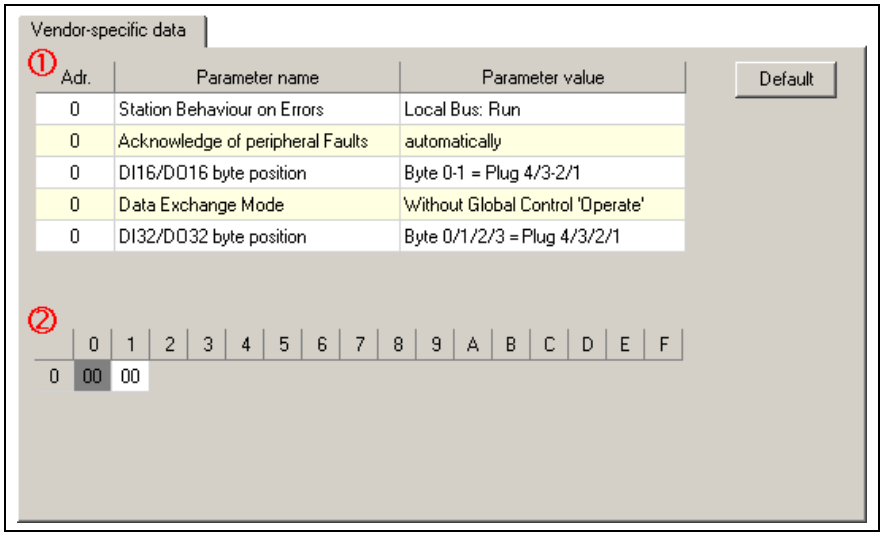
Displaying and editing possibly available, vendor-specific slave parameters. If there are "vendor-specific data" about the slave, they are indicated.

To change a parameter value, double-click on the "Parameter value" or "Value" field in the corresponding table line.

By activating **Default** all modified values are overwritten by the original values from the GSD file.

The data can – depending on the type of GSD file – be indicated in two different manners:

Representation with address, parameter name and value:



1: List with parameter name and value
2: Parameter data in byte view

Fig. 4-29: Vendor-specific data with address, parameter name and value

In this representation a parameter value is indicated and edited according to its data type (1), e. g. "Active" and "Not active" for boolean values. Additionally, all resulting parameter data are indicated in the byte view (2).

Representation of address and value:

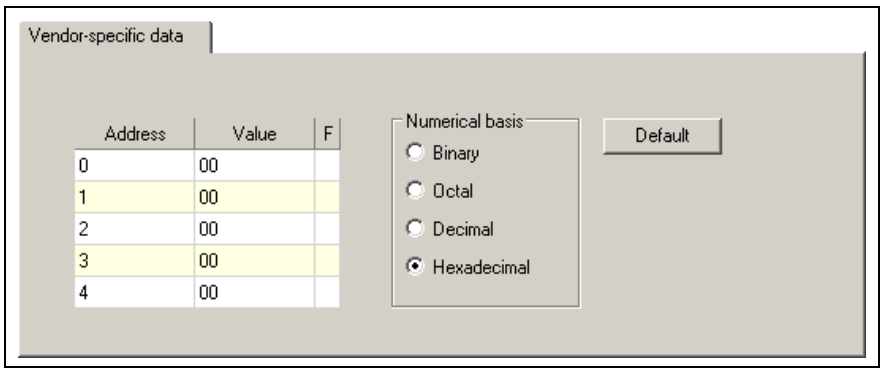


Fig. 4-30: Vendor-specific data with address and value

In this representation the values are divided up into bytes. To indicate the byte values, a **numerical basis** can be set.

"Group assignment" Tab

The group assignment refers to the sync and freeze commands of the "Global Control" services of Profibus DP. To assign the slave to one or several groups, activate the desired group via the "Member of" column.

The **Sync mode** and **Freeze mode** columns indicate to which groups the master actually sends the sync or freeze commands. These assignments are set in the "Groups" tab in the "Master settings", see section "Modifying Master-Specific Settings" on page 4-19.

Example: With the setting illustrated in the following figure sync commands are send from the master to the groups 3 and 4 and freeze commands to the groups 1 and 3. The outputs of the slave are frozen to the present value (synchronized), as soon as the master sends the sync command to group 3. The states of the inputs are frozen to the present value, as soon as the master sends the freeze command to group 1 or group 3.

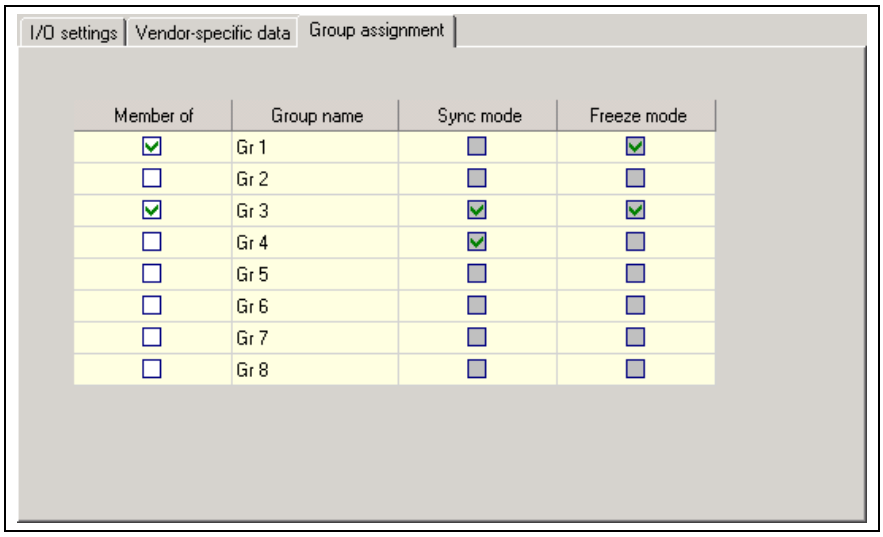


Fig. 4-31: Group assignment of a Profibus DP slave (example)

Inserting Modules in the Profibus DP Slave

Note: Modules can only be inserted for modular structured Profibus DP slaves, see section "Configuring Profibus DP Slaves" on page 4-23.

The modules fitting to the particular Profibus DP slave are in the "Periphery", "ProfibusDP" library below the particular Profibus DP slave. Drag the required modules from the library into the slave object. New modules can be also inserted in the project explorer between already existing modules.

Alternatively, you can also use the **Add slave** function in the context menu of the slave object, see Fig. 4-32. The new module is inserted as last module under the slave.

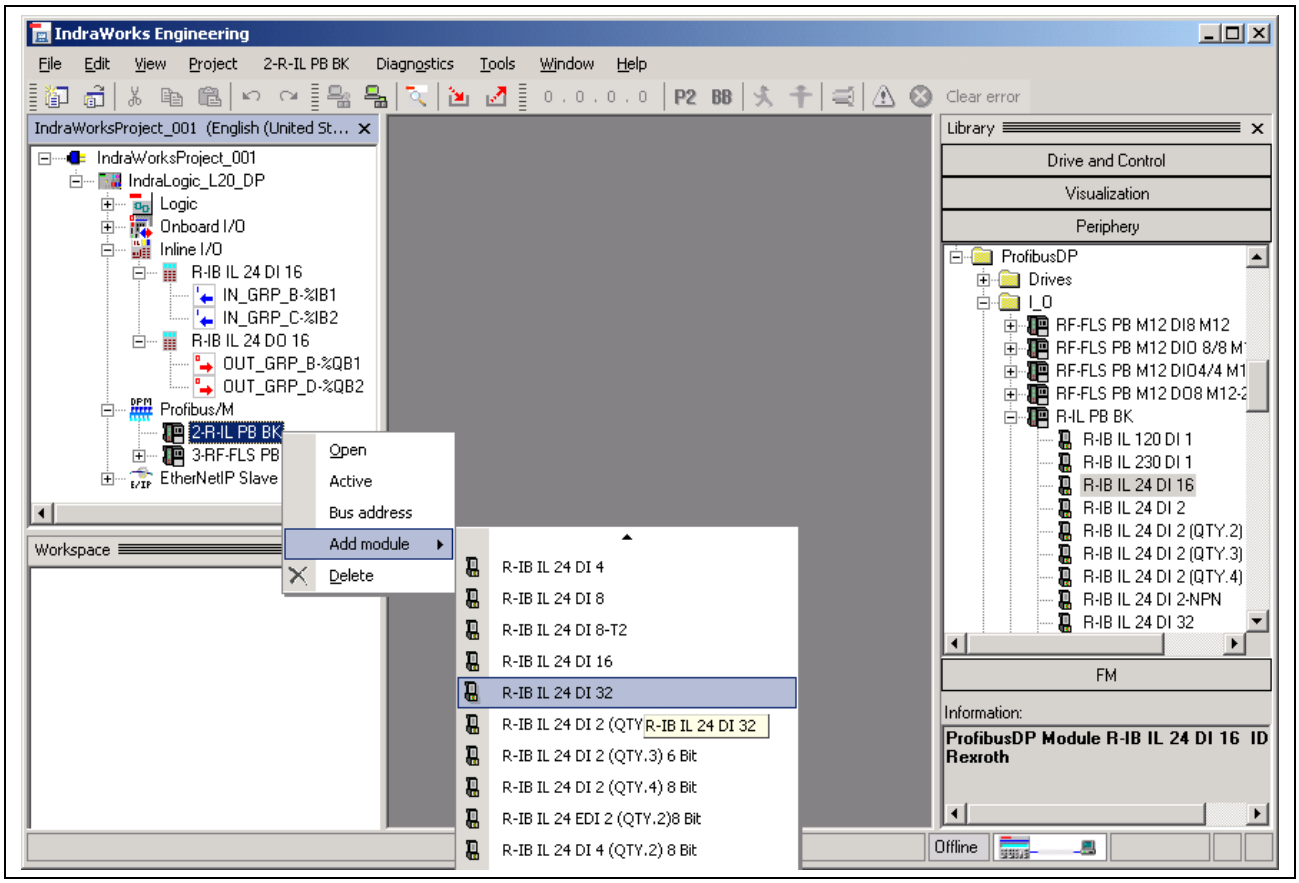


Fig. 4-32: Inserting a module

Configuring Modules of a Profibus DP Slave

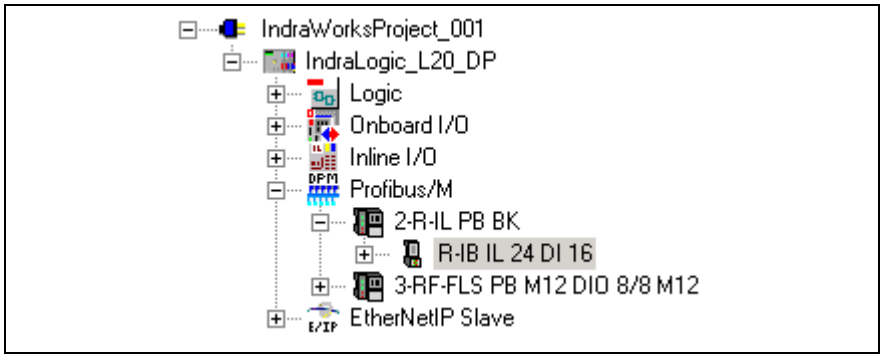


Fig. 4-33: Module of a Profibus DP slave (example)

To configure a module, double-click on the corresponding module entry in the project explorer.

A window opens in the workspace:

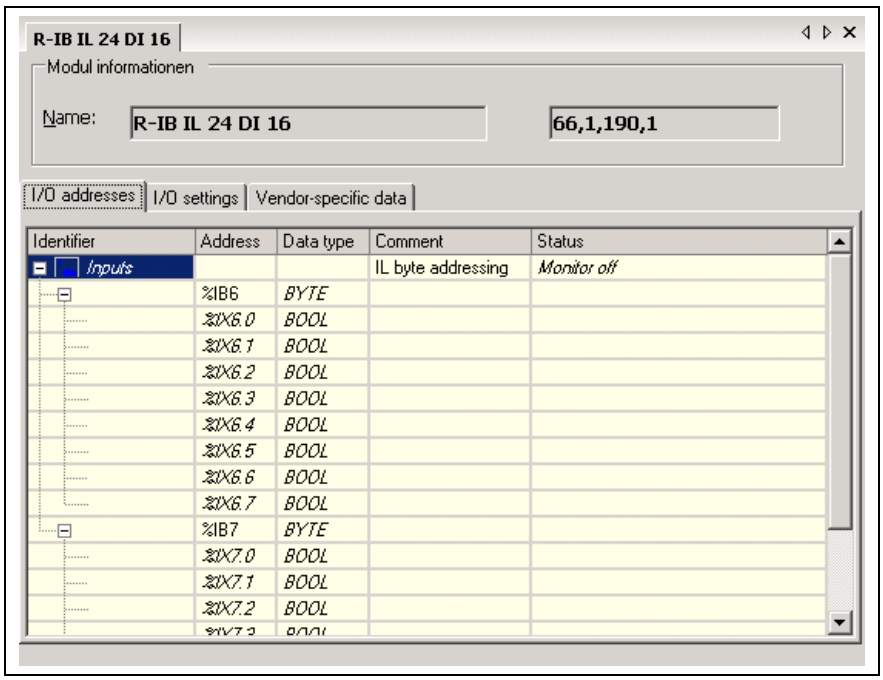


Fig. 4-34: Window of a module (example)

"Module Information"

Name: Module name (field on the left) and internal module identification according to the GSD file (field on the right).

"I/O Addresses" Tab

Assign here the I/O areas of the modules to the physical addresses of the control (I/O addresses of the PLC).

Identifier: This column shows the input and output structure. You can change between the byte and bit view with the plus and minus symbol.

For each absolute address the assignment of a symbolic address is possible (double-click on the particular field). After entering the symbolic address, it is created automatically as global variable in the PLC project.

The symbolic address of a node also appears in the project explorer. You will find an example for onboard I/O in Fig. 4-17.

Address: I/O address. Enter the desired I/O address as byte address (e. g. %IB10). Entries in italics serve only for display and cannot be edited.

Note: An automatic readdressing is possible in the "I/O settings" tab.

Data type: Byte addresses are identified with "BYTE", bit addresses with "BOOL".

Comment: Please enter here any comment about an address.

Status: Physical status of the input/output. The status is only indicated in diagnostic mode during the communication between IndraWorks and IndraLogic L20.

"I/O Settings" Tab

Start here the automatic assignment of I/O addresses.

From output / **From input**: Current or desired start addresses of the outputs or inputs. Corresponding to the used module functionality (module with inputs, module with outputs) only relevant input boxes are indicated. If you parameterize, for example, a module with only inputs, the input box "From output" is missing.

"**Apply**": Renumbers automatically all outputs/inputs of the module in ascending order starting with the indicated start addresses (see "From output / From input"). Please note, that existing address gaps are closed during this process!

If the automatic numbering causes collisions with already assigned address areas, IndraWorks indicates the collision cause and determines automatically the next free address area.

"Vendor-Specific Data" Tab

Displaying and editing possibly available, vendor-specific module parameters. If there are "vendor-specific data" about the module, they are indicated.

For more detailed information about "vendor-specific data" see name-confirm section under "Configuring Profibus DP Slaves" on page 4-23.

4.7 Configuring IndraLogic L20 as Profibus DP Slave

Overview

Via the Onboard Profibus DP interface (X7P) the IndraLogic L20 can be addressed by a superordinated Profibus DP master as Profibus DP slave.

The IndraLogic L20 has to be defined as Profibus DP slave, see section "Device and Function Module Settings" on page 4-4. Only in this case the required "Profibus/S" object is available in the project explorer.

Adapting Slave-Specific Basic Settings

Double-click in the project explorer on the "Profibus/**S**" object.

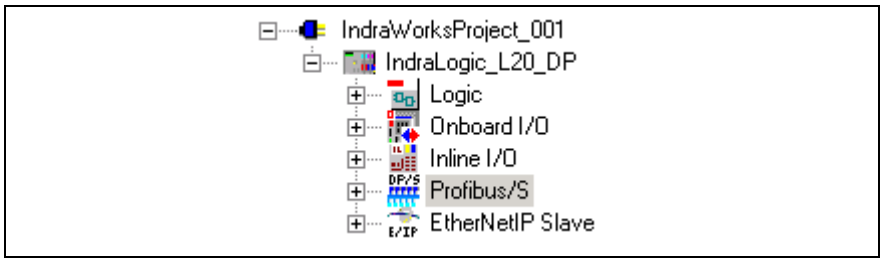


Fig. 4-35: "Profibus/S" object

A window opens in the workspace:

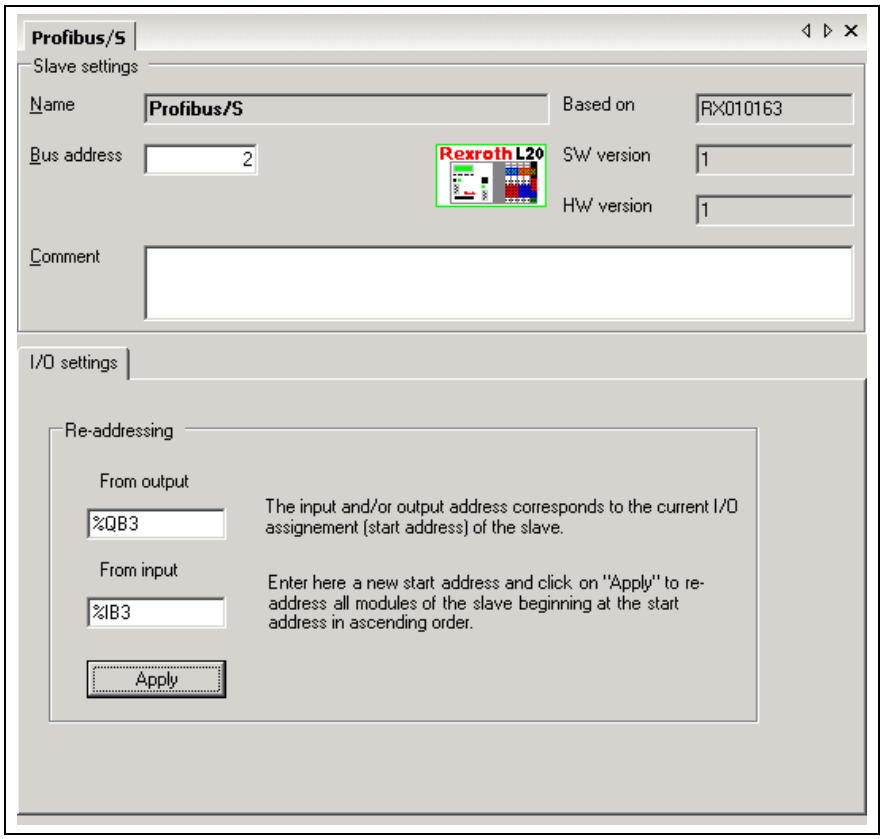


Fig. 4-36: "Profibus/S" window

"Slave Settings"

Name: Internal device name of the slave

Bus address: Bus address of the slave (FDL address: Fieldbus Data Link). Address "2" is automatically entered. If necessary, enter here another address.

Please note the following restrictions for the address selection:

Address "0": reserved for engineering devices

Address "1": reserved for the Profibus DP master

The bus address also appears in the project explorer, see Fig. 4-26.

Based on: Ident number according to the GSD file.

SW version: Software version according to the GSD file.

HW version: Hardware version according to the GSD file.

Comment: Please enter here any comment to describe the slave in a detailed manner.

"I/O Settings" Tab

Start here the automatic assignment of I/O addresses.

From output / **From input**: Desired start addresses for the automatic numbering of the I/O coupling area (description see page 4-32).

"**Apply**": Renumbers automatically all outputs/inputs of the I/O coupling area in ascending order starting with the indicated start addresses (see "From output / From input"). Please note, that existing address gaps are closed during this process!

If the automatic numbering causes collisions with already assigned address areas, IndraWorks indicates the collision cause and determines automatically the next free address area.

Parameterizing the Coupling Area

A coupling area is provided to exchange data between the IndraLogic L20 as Profibus DP slave and the superordinated control. The coupling area is a reserved I/O area of the IndraLogic L20 to exchange e. g. status and diagnostic data.

By default, the data capacity of the coupling area is set to 16 bytes. The data capacity can be independently set for the input and output area in steps of 8 bytes between 8 and 64 bytes. For this, use the **Replace module** function in the context menu of the "Profibus/S" object, see Fig. 4-37.

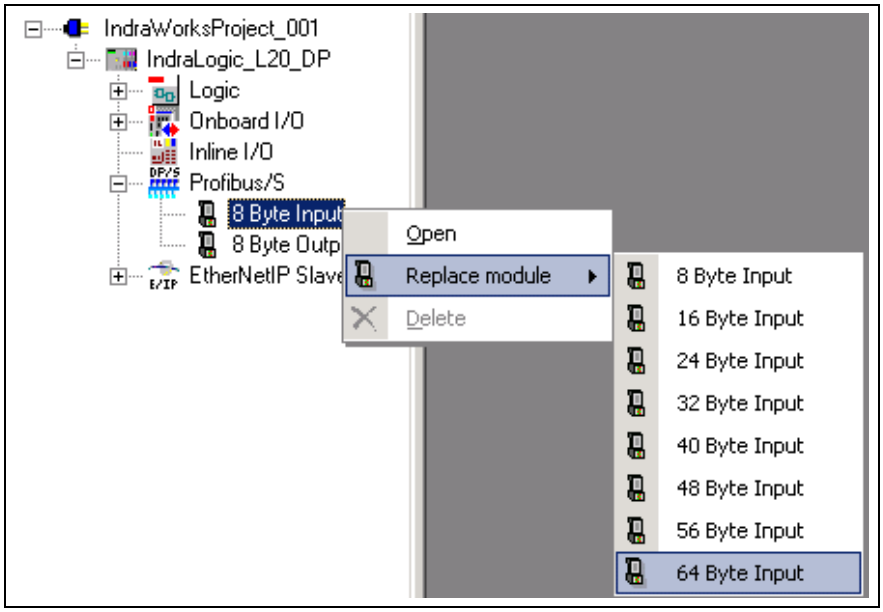


Fig. 4-37: Select data capacity for input/output area

To configure the inputs/outputs, double-click on the corresponding module entry in the project explorer.

A window opens in the workspace:

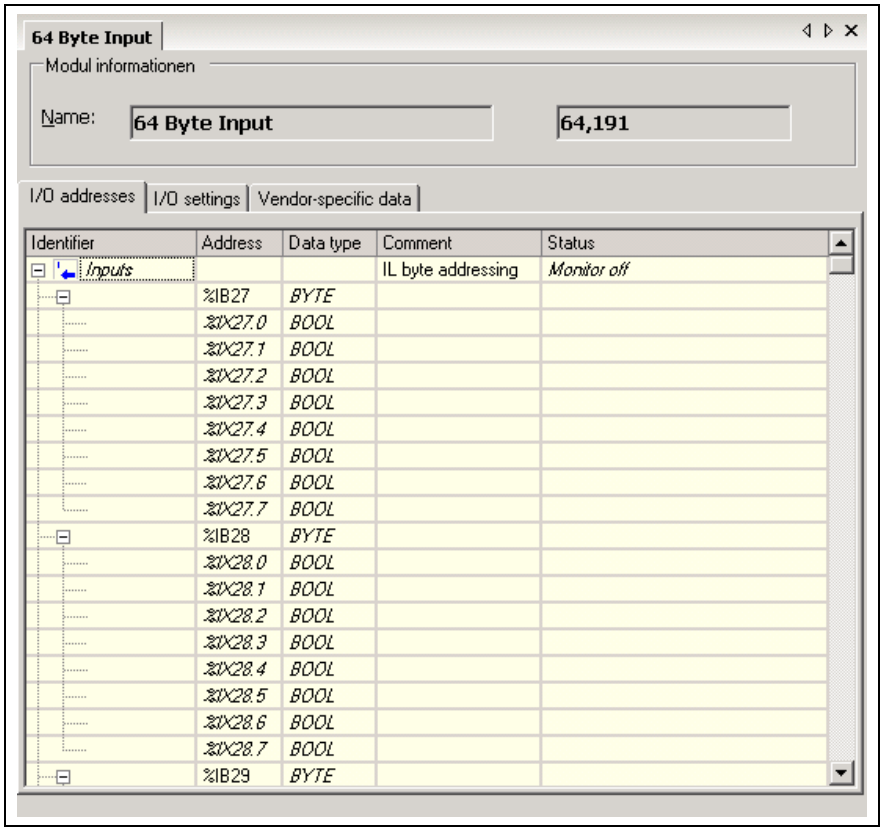


Fig. 4-38: Coupling area of a "Profibus/S" object (example)

"Module Information"

Name: Module name (field on the left) and internal module identification (field on the right).

"I/O Addresses" Tab

Assign here the coupling area to the physical addresses of the control (I/O addresses of the PLC). The addresses are the local I/O addresses of the IndraLogic L20.

Identifier: This column shows the input and output structure. You can change between the byte and bit view with the plus and minus symbol.

For each absolute address the assignment of a symbolic address is possible (double-click on the particular field). After entering the symbolic address, it is created automatically as global variable in the PLC project.

The symbolic address of a node also appears in the project explorer. You will find an example for onboard I/O in Fig. 4-17.

Address: I/O address. Enter the desired I/O address as byte address (e. g. %IB10). Entries in italics serve only for display and cannot be edited.

Note: An automatic readdressing is possible in the "I/O settings" tab.

Data type: Byte addresses are identified with "BYTE", bit addresses with "BOOL".

Comment: Please enter here any comment about an address.

Status: Physical status of the input/output. The status is only indicated in diagnostic mode during the communication between IndraWorks and IndraLogic L20.

"I/O Settings" Tab

Start here the automatic assignment of I/O addresses.

From output / **From input**: Current or desired start addresses for the automatic numbering of the I/O coupling area.

"**Apply**": Renumbers automatically all outputs/inputs of the I/O coupling area in ascending order starting with the indicated start addresses (see "From output / From input"). Please note, that existing address gaps are closed during this process!

If the automatic numbering causes collisions with already assigned address areas, IndraWorks indicates the collision cause and determines automatically the next free address area.

"Vendor-Specific Data" Tab

To configure the coupling area, this tab is not relevant.

Projecting a Superordinated Control

To project the IndraLogic L20 as Profibus DP slave for the Profibus operation within the superordinated control, use the "RX010163" GSD file or the corresponding device in the "Periphery" library under "Profibus", "PLC":

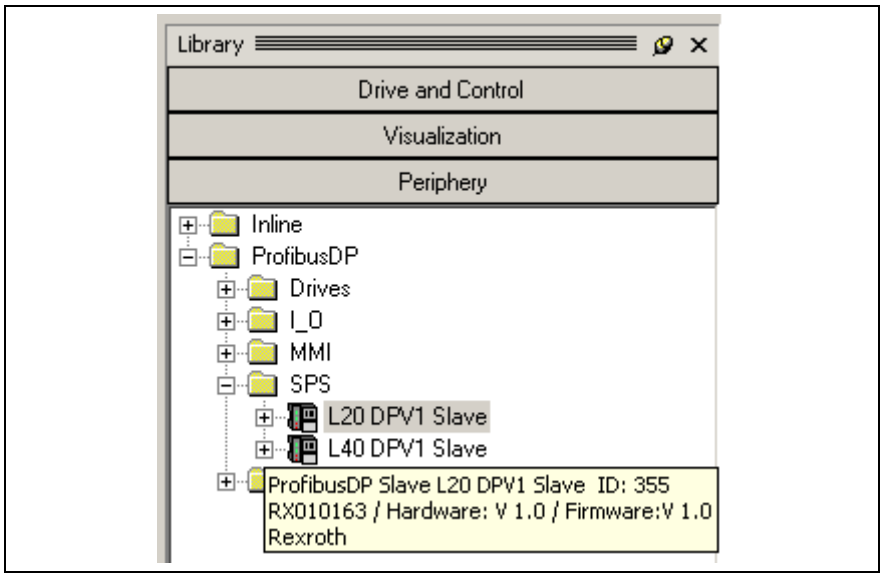


Fig. 4-39: Projecting the IndraLogic L20 in the superordinated control as slave (device in the library)

While projecting the device, select the modules in the superordinated control, that correspond to the data capacity of the coupling area of the inputs and outputs. The following figure shows an example with a coupling area of 8 byte inputs and 8 byte outputs.

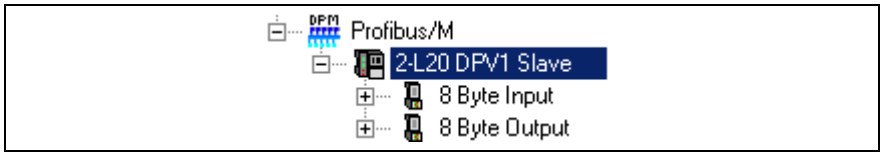


Fig. 4-40: Define the coupling area in the superordinated control

4.8 Configuring IndraLogic L20 as Ethernet/IP Slave

Overview

Via the Ethernet interface (X7E) the IndraLogic L20 can be addressed by a superordinated control (master/originator) as Ethernet/IP slave (target).

The IndraLogic L20 has to be defined as Ethernet/IP slave, see section "Device and Function Module Settings" on page 4-4. Only in this case the required subordinated "Ethernet/IP slave" object is available in the project explorer.

Adapting Slave-Specific Basic Settings

Double-click in the project explorer on the "Ethernet/IP slave" object.

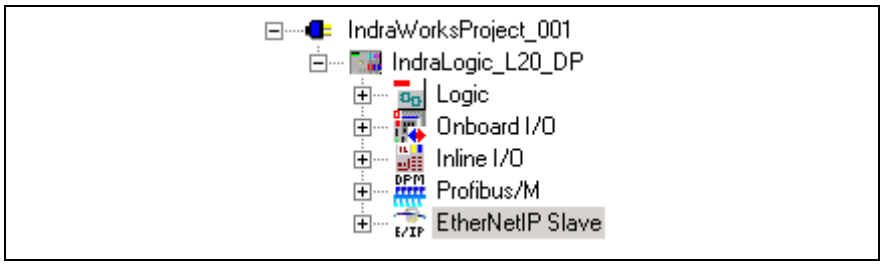


Fig. 4-41: "Ethernet/IP slave" object

A window opens in the workspace:

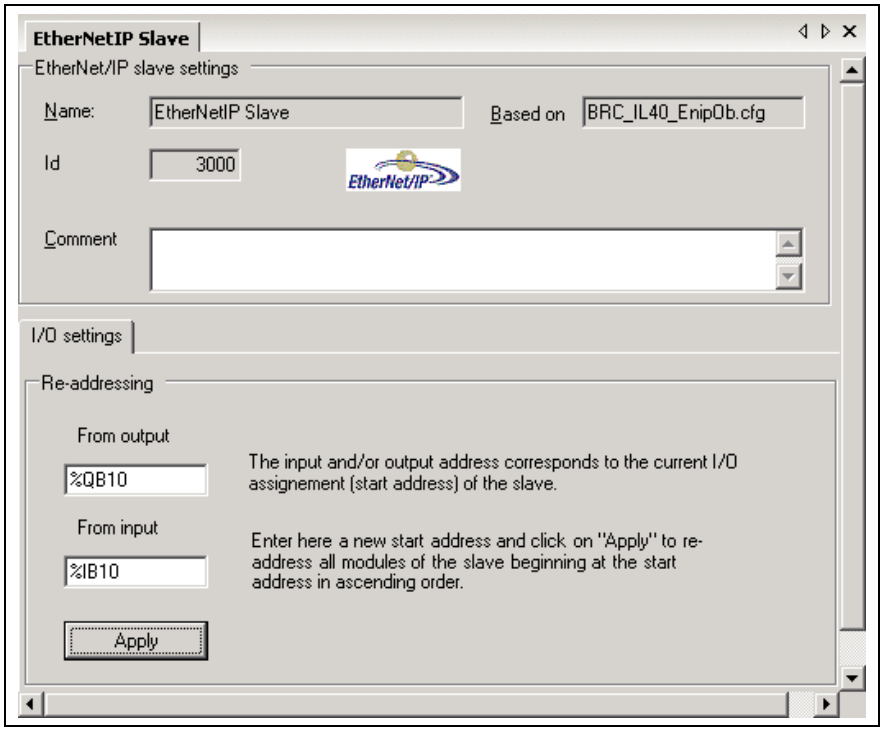


Fig. 4-42: "Ethernet/IP slave" window

"Ethernet/IP Slave" Settings

Name: Internal device name of the slave

Based on: Name of the used device description file. The file is a fixed part of the installed target system.

ID: Internal name, specified by the installed target system.

Comment: Please enter here any comment to describe the slave in a detailed manner.

"I/O Settings" Tab

Start here the automatic assignment of I/O addresses.

From output / **From input**: Desired start addresses for the automatic numbering of the I/O coupling area.

"**Apply**": Renumbers automatically all outputs/inputs of the I/O coupling area in ascending order starting with the indicated start addresses (see "From output / From input"). Please note, that existing address gaps are closed during this process!

If the automatic numbering causes collisions with already assigned address areas, IndraWorks indicates the collision cause and determines automatically the next free address area.

Parameterizing the Coupling Area

A coupling area is provided to exchange data between the IndraLogic L20 as Profibus DP slave and the superordinated control. The coupling area is a reserved I/O area of the IndraLogic L20 to exchange e. g. status and diagnostic data.

By default, the data capacity of the coupling area is preset to 8 byte input and 8 byte output plus 4 byte Run Header. The data capacity can be independently set for the input and output area in steps of 8 bytes between 0 and 128 bytes. For this, use the **Replace module** function in the context menu of the "Ethernet/IP slave" object, see Fig. 4-43.

Note: If the input and output box is parameterized with data capacities equal to 0 (module "0 Byte Input" and "0 Byte Output..."), cyclic communication is no longer possible.

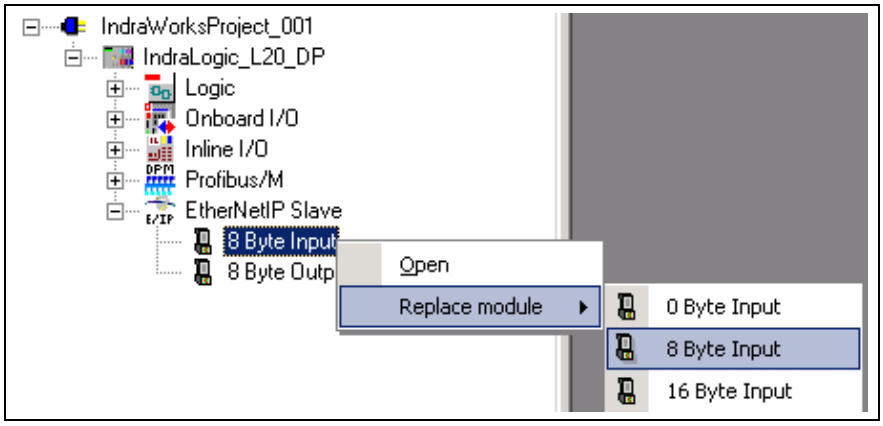


Fig. 4-43: Select the data capacity for the input and output box of the Ethernet/IP slave

To configure the inputs/outputs, double-click on the corresponding module entry in the project explorer.

A window opens in the workspace:

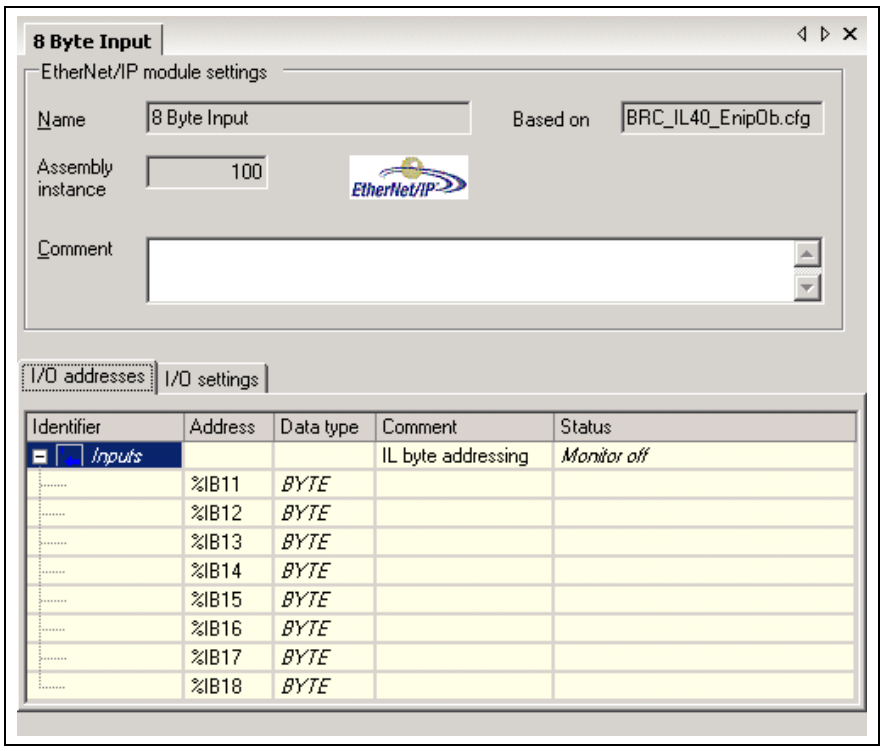


Fig. 4-44: Coupling area of an "Ethernet/IP slave" object (example)

"Ethernet/IP Module" Settings

Name: Internal name, specified by the installed target device.

Based on: Name of the used device description file. The file is a fixed part of the installed target system.

Assembly Instance: Internal identification.

Comment: Any text to describe the I/O field in a detailed manner.

"I/O Addresses" Tab

Assign here the coupling area to the physical addresses of the control (I/O addresses of the PLC). The addresses are the local I/O addresses of the IndraLogic L20.

Identifier: This column shows the input and output structure. You can change between the byte and bit view with the plus and minus symbol.

For each absolute address the assignment of a symbolic address is possible (double-click on the particular field). After entering the symbolic address, it is created automatically as global variable in the PLC project.

The symbolic address of a node also appears in the project explorer. You will find an example for onboard I/O in Fig. 4-17.

Address: I/O address. Enter the desired I/O address as byte address (e. g. %IB10). Entries in italics serve only for display and cannot be edited.

Note: An automatic readdressing is possible in the "I/O settings" tab.

Data type: Byte addresses are identified with "BYTE", bit addresses with "BOOL".

Comment: Please enter here any comment about an address.

Status: Physical status of the input/output. The status is only indicated in diagnostic mode during the communication between IndraWorks and IndraLogic L20.

"I/O Settings" Tab

Start here the automatic assignment of I/O addresses.

From output / **From input**: Current or desired start addresses for the automatic numbering of the I/O coupling area.

"**Apply**": Renumbers automatically all outputs/inputs of the I/O coupling area in ascending order starting with the indicated start addresses (see "From output / From input"). Please note, that existing address gaps are closed during this process!

If the automatic numbering causes collisions with already assigned address areas, IndraWorks indicates the collision cause and determines automatically the next free address area.

Ethernet/IP Connection Types

With the IndraLogic L20 as Ethernet/IP slave data blocks can be cyclically transmitted ("Implicit Messaging") via an "Exclusive Owner Connection" (Transport Class 1) and a "Listen Only Connection" (Transport Class 1). By means of the additional "Listen Only Connection" a second master can read the Ethernet/IP output image of the IndraLogic L20 via multicast, if the an "Exclusive Owner Connection" exists.

The input and output field size can be parameterized between 0 and 128 bytes with an incremental width of 8 bytes.

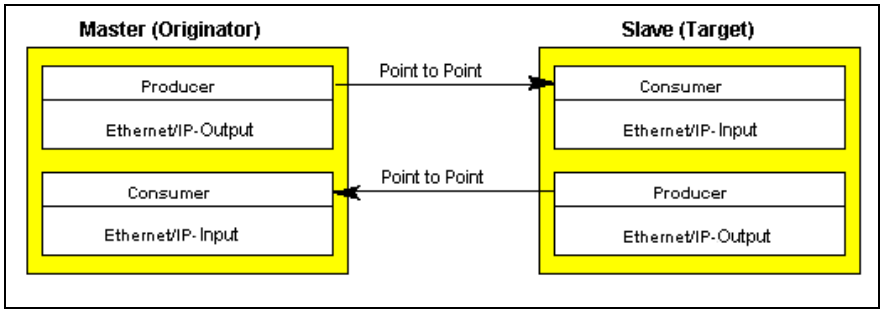


Fig. 4-45: Ethernet/IP connection type: Point to point

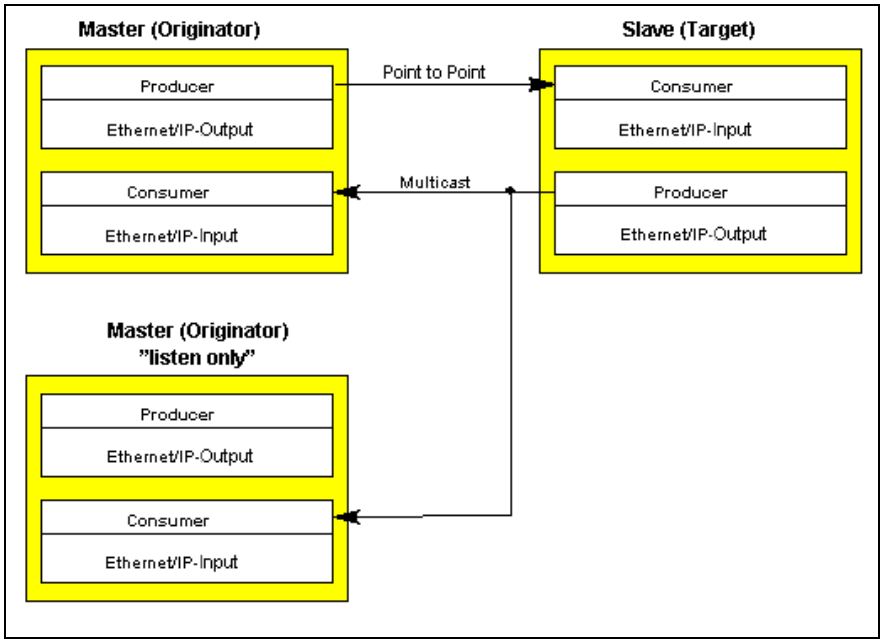


Fig. 4-46: Ethernet/IP connection type: Point to point / multicast

Features of the Cyclic Data Transmission

- The full-duplex operation of the Ethernet connection is required. Otherwise, transmission timeouts are not possible.
- The lowest cycle time RPI (Requested Packet Interval) supported by the IndraLogic L20 is 5 ms.
- The Ethernet/IP "Idle/Run" header (32 bits) is supported. The header is not visible in the cyclic I/O image of the IndraLogic L20.

Reaction of the IndraLogic L20:

- Idle status of the master (bit 0 in the Idle/Run header is not set): Input image is set to zero.
- Stop status of the IndraLogic L20: Output image is set to zero, input image is continually updated.
- Transmission timeout: Input image is set to zero.

The consumer instance of the IndraLogic L20 monitors the cyclic transmission of the master output image by means of a monitoring time (timeout time). For this, the IndraLogic L20 receives during the initialization of the cyclic data channel corresponding parameters from the master. The resulting monitoring time can be calculated using the following formulas:

$$TM = 2^{(TMV+2)}$$

$$t = TM * RPI$$

TMV: Timeout Multiplier Value
TM: Timeout Multiplier
RPI: Requested Packet Interval (in µs)
t: Monitoring time (in ms)

Fig. 4-47: Formulas to calculate the monitoring time t

Examples:

TMV	TM	RPI (in µs)	t (in ms)
0	4	5000	20
1	8	5000	40
2	16	5000	80
3	32	5000	160
4	64	5000	320
2	16	10000	160

TMV: Timeout Multiplier Value
TM: Timeout Multiplier
RPI: Requested Packet Interval
t: Monitoring time

Fig. 4-48: Resulting monitoring times (example)

4.9 Configuring IndraLogic L20 as CIP Data Server

Overview

As CIP data server (CIP: Communication Industrial Protocol): the IndraLogic L20 provides via the Onboard Ethernet interface (X7E) enabled variables via acyclic DataTable services ("Unconnected Explicit Messaging" and "Class 3 Connected Messaging").

Projecting

To use the IndraLogic L20 as CIP data server please execute the following steps:

1. Define the IndraLogic L20 as Ethernet/IP slave, see section "Device and Function Module Settings" on page 4-4. Only in this case the required subordinated "Ethernet/IP slave" object is available in the project explorer.
2. If no additional cyclic communication is desired, parameterize the data capacities of the I/O coupling area equal to 0. For this, use the particular context menu item **Replace module** and select the modules "0 Byte Input" or "0 Byte Output..." , see Fig. 4-43.
3. Define the variables relevant in IndraLogic.
4. Ensure that the options "Generate icon entries" and "Generate XML icon table" are activated in IndraLogic under **Project – Options – Icon configuration**.
5. Open in IndraLogic the "Set object attributes" dialog via **Project – Options – Icon configuration – Configure icon file**. Select the desired variables and activate the "Output variables of the object" option.

Note: The "Write access" option affects the write protection of a variable. To avoid an external overwriting of the variable value, deactivate the writing access.

Features of the Acyclic Data Transmission

- Maximum 6 simultaneous connections to CIP data clients are possible.
- The variables are addressed via their name.
- The access is possible to variables (INT, DINT, ...) as well as to one-dimensional variable arrays (up to 450 bytes).
- The data server acknowledges the read and write accesses. Reasons for negative acknowledgements are listed in the following table:

Description	Error code	Additional error code
No symbol information for variable available	0x1F	0x0101
Variable is read-only	0x1F	0x0106
Too many data sent during the write access	0x1F	0x0107
Too less data sent during the write access	0x1F	0x0108

Fig. 4-49: Error codes of the CIP data server

4.10 Configuring IndraLogic L20 as CIP Data Client

As CIP data client the IndraLogic L20 can access via the Onboard Ethernet interface (X7E) enabled variables of a CIP data server via acyclic DataTable services.

For this, besides the activation it is required to use the "RIL_EtherNetIP" library.

To use the IndraLogic L20 as CIP data client please execute the following steps:

1. Define the IndraLogic L20 as Ethernet/IP slave, see section "Device and Function Module Settings" on page 4-4. Only in this case the required subordinated "Ethernet/IP slave" object is available in the project explorer.
2. If no additional cyclic communication is required, parameterize the data capacities of the I/O coupling area equal to 0. For this, use the particular context menu item **Replace module** and select the modules "0 Byte Input" or "0 Byte Output...", see Fig. 4-43.
3. Ensure that the CIP data server provides the relevant variables.
4. Define in your PLC program all variables required for the used function blocks of the "RIL_EtherNetIP" library. Call the corresponding function blocks in an appropriate manner.

4.11 PLC Programming with IndraLogic

Overview

For the PLC project planning the following functions can be executed via program IndraLogic integrated in IndraWorks:

- Target Settings: special settings of the IndraLogic L20
- Task Configuration: Control of the program processing
- Library Manager: Managing PLC module libraries
- Creating a PLC Program: Creating modules in PLC programming languages

Note: The PLC configuration is not generated within IndraLogic, but in IndraWorks, see chapter "Defining Basic Settings (Wizard-Guided)" as of page 4-3. Thus, many input boxes of the IndraLogic PLC configuration are deactivated and cannot be changed.

The PLC project planning with IndraLogic is activated via the "Logic" object in the project explorer.

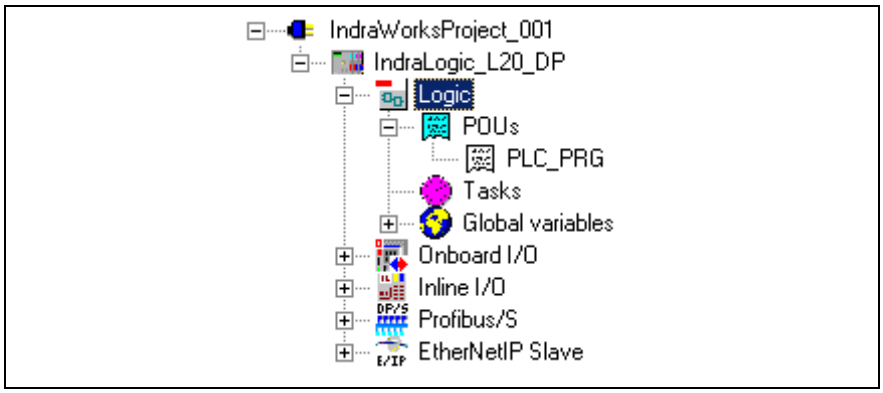


Fig. 4-50: "Logic" object in the project explorer

- A double-click on the "Logic" object starts IndraLogic with the target system "IndraLogic L20".
- By double-clicking on an entry below "POUs" (e. g. PLC_PRG) the corresponding module opens in IndraLogic for further editing, see section "Creating a PLC Program" on page 4-48.
- By double-clicking on "Tasks" the task configuration in IndraLogic starts, see section "Task Configuration" on page 4-45.

Update Project

All modules generated in IndraLogic as well as the modifications of the settings are applied in the IndraWorks project explorer as soon as the context menu function **Update** is selected in the "Logic" object:

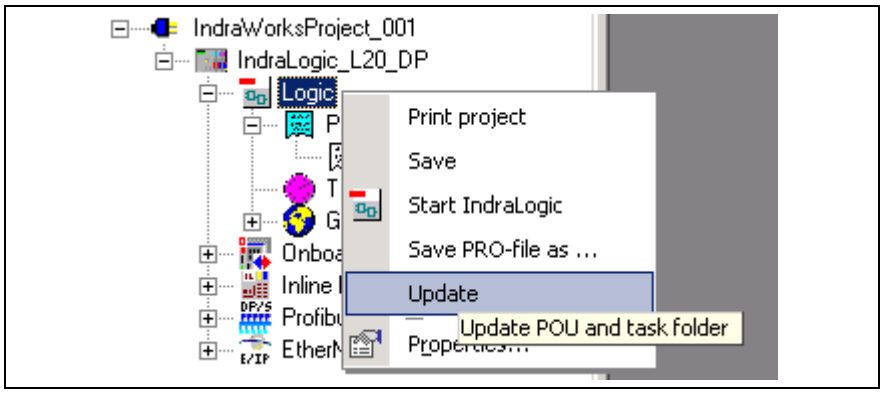


Fig. 4-51: Update "Logic" object

Target Settings

CAUTION

Each modification of the preset target configuration can have serious effects on the behavior of the target system!

⇒ The target settings may only be modified by instructed specialists, who are aware of the effects of this modifications!

The target settings are optimized to the most frequently used applications. For special applications some settings can be adapted. The target settings are modified via the properties dialog of the "Logic" object, when the application IndraLogic is closed.

Settings, that are not available on the IndraWorks level (e. g. "Network functions") can be made in IndraLogic under "Target settings". These settings can be reached in IndraLogic by clicking on "Target settings" in the "Resources" tab, see Fig. 4-52.

For further information about the target settings refer to the documentation or online help of IndraLogic /2/.

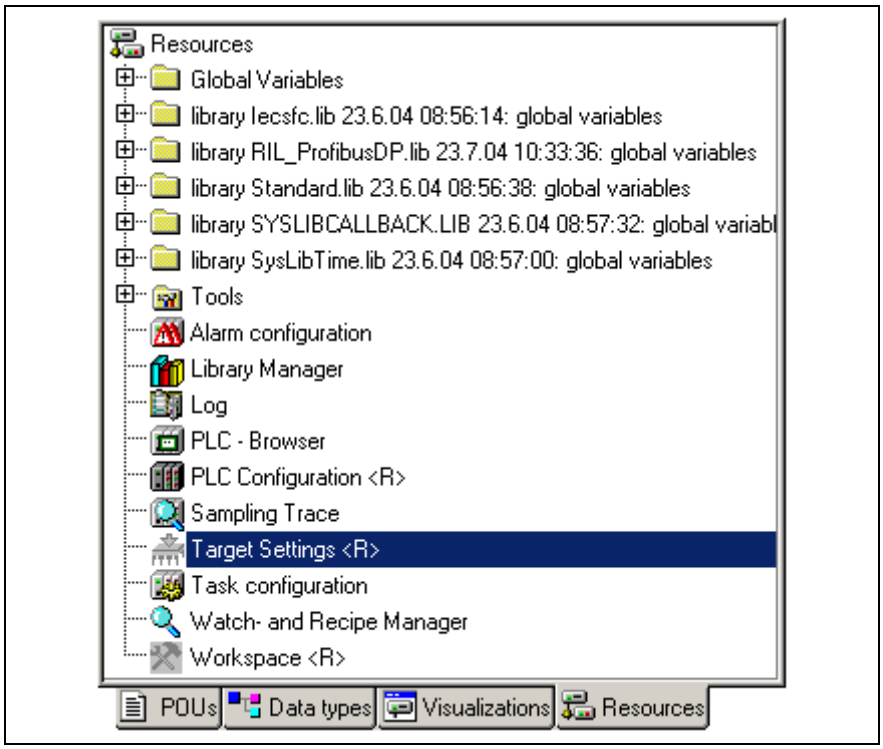


Fig. 4-52: IndraLogic, resources, "target settings" object

Interrupting the Database Connection

Usually, a connection to the database server is active. If a connection is active, you can recognize by the addition "<R>" in the name of the "Target settings" object.

To be able to modify the target settings, at first, you have to interrupt the connection to the database. For this, select in the context menu of the target settings menu item **Project database – Check Out**, see Fig. 4-53. After changing the target settings the database connection can be reestablished via **Project database – Check In**. After modifying the settings menu function **Project database – Get Latest Version** can be executed to synchronize the database between IndraWorks and IndraLogic.

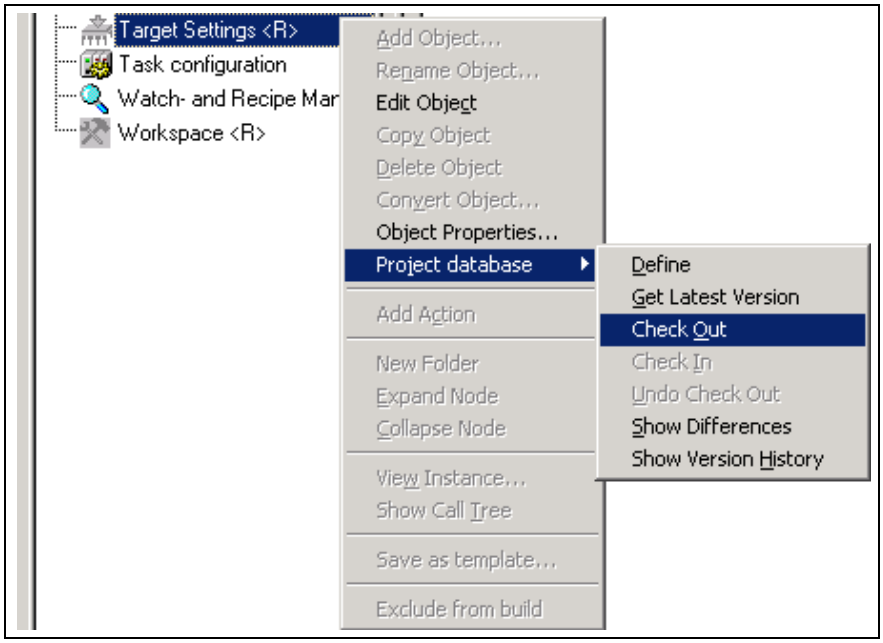


Fig. 4-53: Target settings: Check out from the database

Task Configuration

For many application cases it is sufficient to execute the program without special task management. The program processing is simply executed via the "main program" PLC_PRG. PLC_PRG is automatically generated as module of type "Program" and is called up exactly one time in each control cycle.

For special application cases it is possible to control the processing of the tasks. For this, the task configuration is required.

Calling the task configuration:

- In the project explorer of IndraWorks: Double-click on the "Tasks" object of the relevant "Logic" objects

- or -

- in the "Resources" tab of IndraLogic: Double-click on the "Task configuration" object.

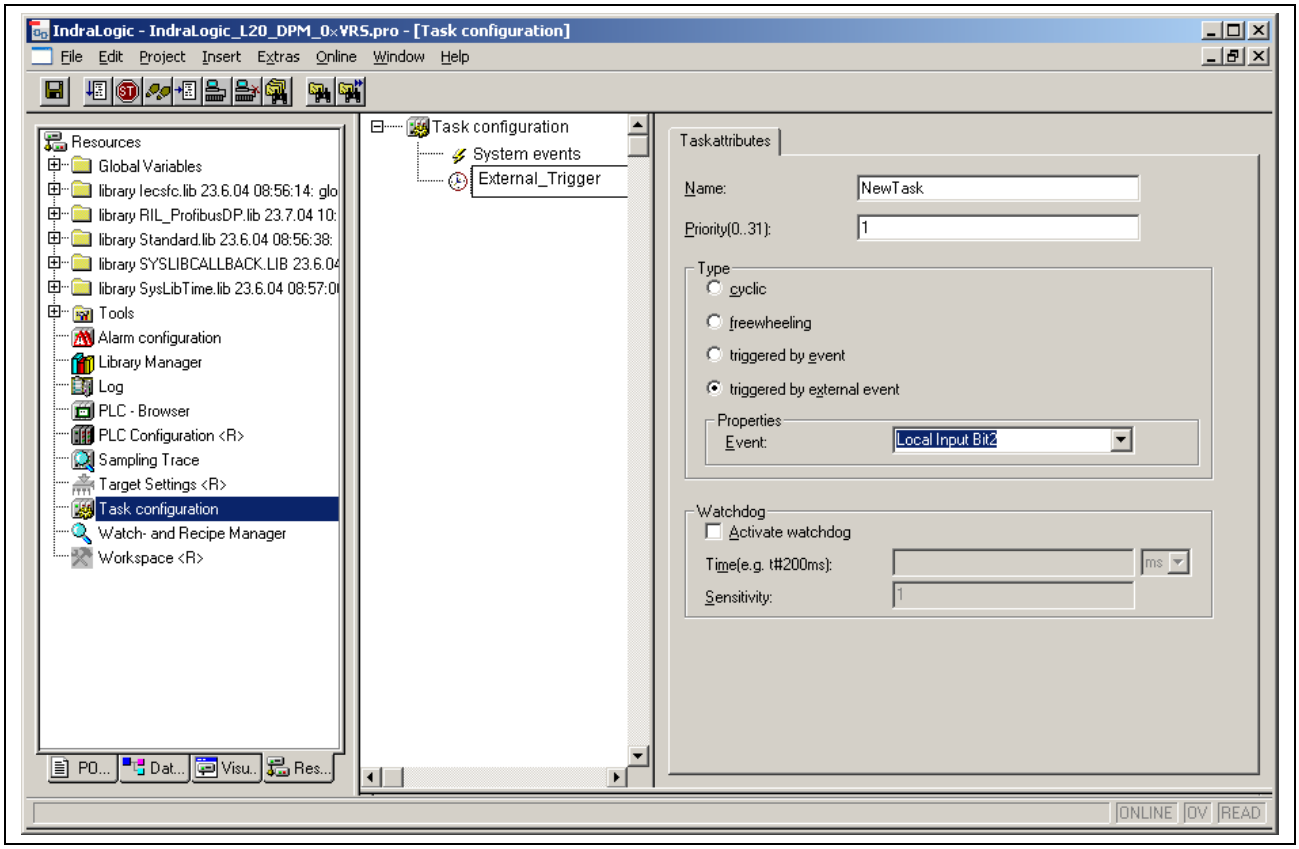


Fig. 4-54: IndraLogic task configuration

For further information about the task attributes refer to the documentation or online help of IndraLogic /2/.

The IndraLogic L20 provides a special task-controlled function:

Triggered by External Event

A task can be started by a rising edge/ by rising edges at one or several onboard inputs.

For this, select in the "Task properties" tab the "Triggered by external event" option and in the "Event" box the required event:

- **Local Input Bit 0** ... **Local Input Bit 7**: As soon as a rising edge is applied to the selected bit input of the onboard I/O, the task is started.

- **Local Input Byte**: As soon as a rising edge is detected at least one of the eight onboard inputs, the task is started. Even if positive signal level are already applied to the individual inputs, the task starts with each new rising edge of an input. Simultaneously incoming rising edges are recognized as event and, thus, do not start the task several times.

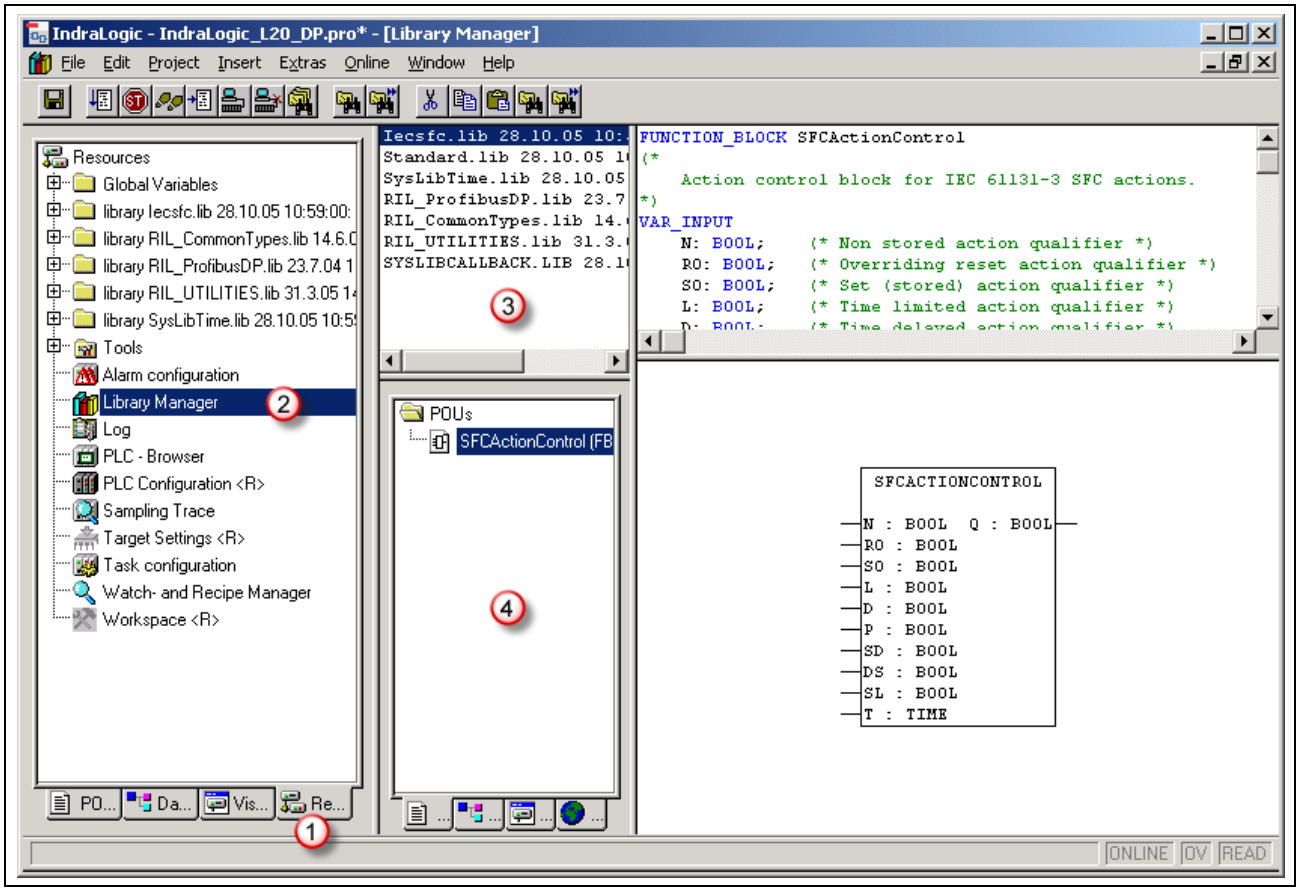
Library Manager

For further information about the library manager refer to the documentation or online help of IndraLogic /2/. For information about the libraries of the IndraLogic L20 refer to chapter 6 "Libraries".

With the IndraLogic L20 already a few libraries are indicated in IndraLogic in the "Resources" tab.

Note: IndraWorks loads libraries for internal use (see chapter 6 "Libraries") automatically as soon as they are referenced. It is not necessary to insert them manually with the library manager.

To link further libraries with the current project, double-click at first in the "Resources" tab on the "Library manager" object. The library manager opens:



(1) "Resources" tab
(2) "Library manager" object
(3) Display of the loaded libraries
(4) Display of all modules contained in a selected library

Fig. 4-55: Library manager

Select menu function **Insert – Further library** or in the context menu of area (3) **Further library...**. A dialog for library selection appears, see Fig. 4-56. Select the library directory of the installed IndraLogic L20 files (target files).

The data are archived by default in directory "...\Rexroth\IndraWorks\IndraLogic\Targets\<*TargetSystemName*>\lib".

Example: Library directory of the IndraLogic L20 DPM 03 VRS: "C:\Program Files\Rexroth\IndraWorks\IndraLogic\Targets\IndraLogic_VEP_01VRS\lib\"

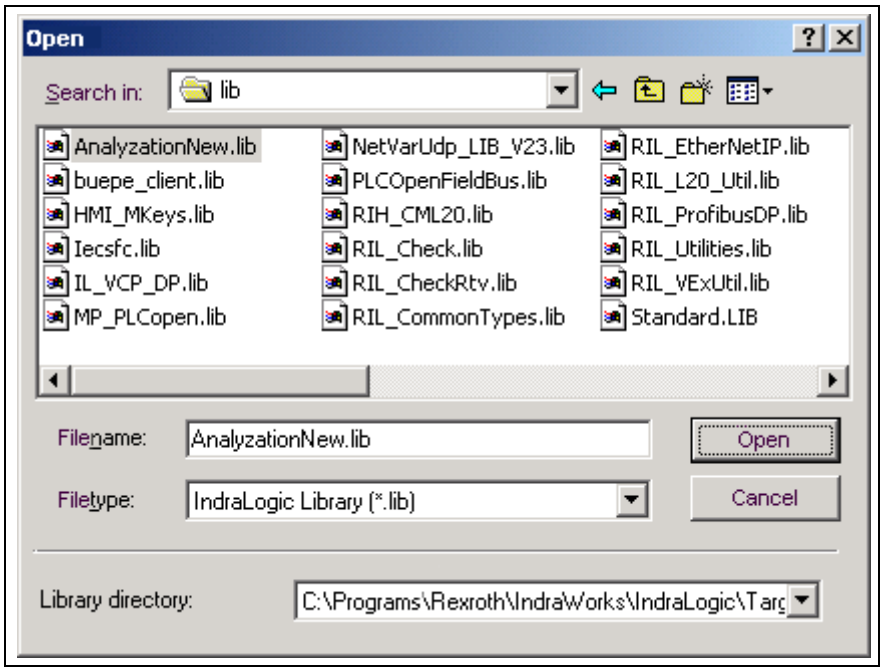


Fig. 4-56: Library directory of the IndraLogic L20 target

Select the required library and acknowledge the selection by pressing the "Open" button. The library can be applied and used in area (3) (see Fig. 4-55).

Creating a PLC Program

Create the PLC program in IndraLogic. For further information refer to the documentation or online help of IndraLogic /2/.

To create compatible IEC programs and to organize the memory of I/O addresses, please consider chapter "Compatible IEC Programming between Different Controls" as of 4-49.

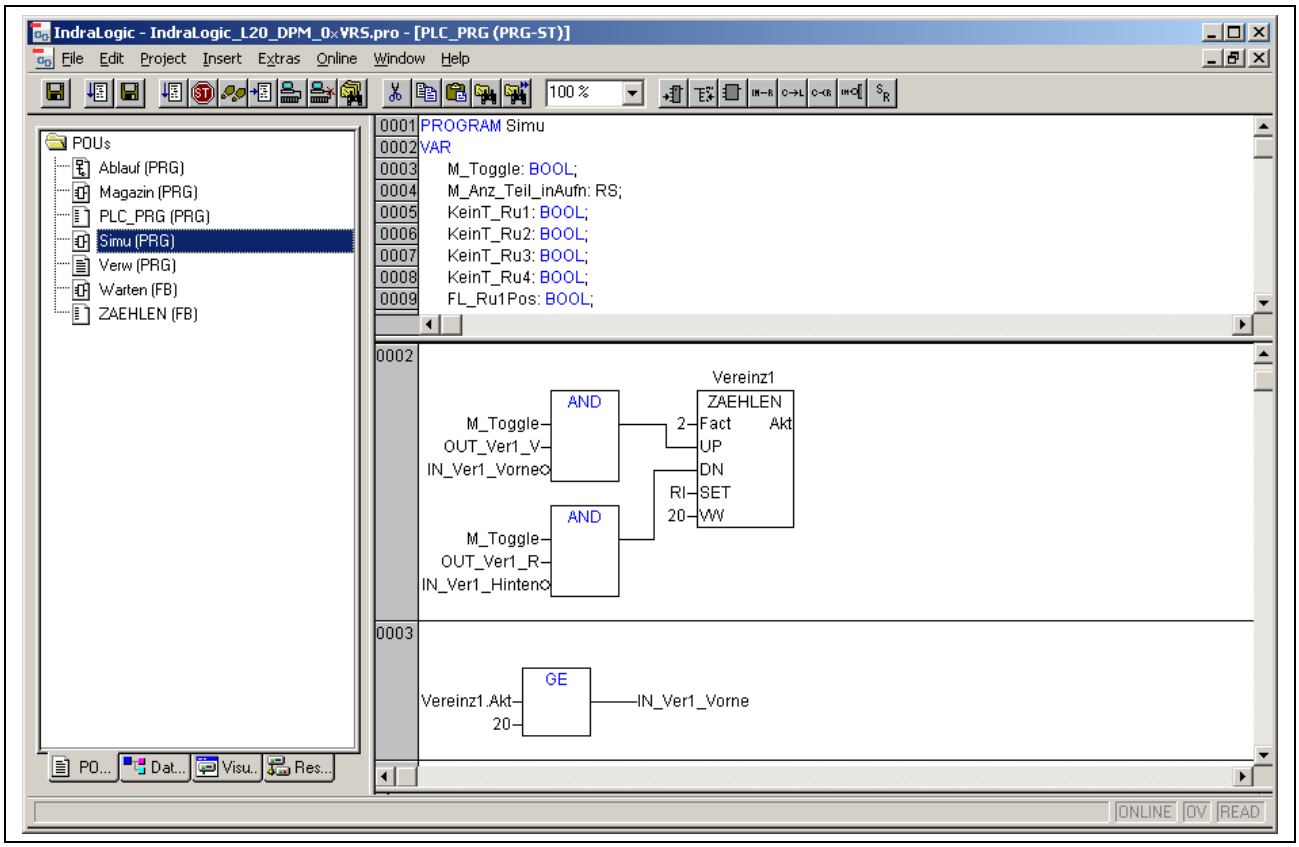


Fig. 4-57: PLC program example

Saving IndraLogic Project Data

With menu function **File – Save** all settings executed in IndraLogic and all edited modules can be saved.

Note: Always save the IndraLogic project data at first, before you change the current project in IndraWorks.

4.12 Compatible IEC Programming between Different Controls

Overview

To ensure a compatible programming between the systems, consider the following features: It is described, which programming methods can be used, so that no incompatibilities may occur:

- Use of pointers within structures
- Packing structures for IndraLogic L20
- Memory alignment for I/O addresses
- Assignment of structures to I/O addresses

Use of Pointers within Structures

If structures are created in the IEC program, the compiler maps this structures during the compilation process in the data memory of the control. The compiler recognizes exactly the possible restrictions of the used processor platform and archives the elements of the structure in the memory by using only addresses, that the processor can utilize for the corresponding data types.

If structures with elements containing different data types are created, the compiler inserts, e. g., for the IndraLogic L20, filling bytes, that are not visible for the user.

Example **Structural differences between the controls**

```
TYPE OutStruct :
STRUCT
      Out01 : BYTE;
      Out02 : WORD;
      Out03 : BYTE;
      Out04 : DWORD;
END_STRUCT
END_TYPE
```

Fig. 4-58: Example of a structure declaration in the IEC program

Depending on the platform this structure is mapped to the memory as follows (here a comparison between the controls IndraLogic L40 and L20):

IndraLogic L40	IndraLogic L20
ADR0 Out01 : BYTE; ADR1 Out02 : WORD; ADR3 Out03 : BYTE; ADR4 Out04 : DWORD; ADR8 :	ADR0 Out01 : BYTE; ADR1 (filling byte) ADR2 Out02 : WORD; ADR4 Out03 : BYTE; ADR5 (filling byte) ADR6 (filling byte) ADR7 (filling byte) ADR8 Out04 : DWORD; ADR12 :

Fig. 4-59: Resulting control-independent memory image

The structure elements are differently mapped, so that a compatible programming method being independent of the control's mapping must be used when accessing the structure elements.

Permissible Addressing

Thus, the compatible use of structures provides for a direct addressing of the elements via the point operator:

```
Structure name.Element name := Element value;
```

If the address of a structure is to be transferred to subfunctions via a pointer, the addressing of the elements is also only permissible by the offset calculation of the compiler:

```
Structure pointer^.Element name := Element value;
```

```
StructInst : OutStruct;            (* Declaration structure instance *)
pt : POINTER TO StructInst;        (* Declaration pointer to structure instance *)

StructInst.Out03 := 2#11110000;    (* Access to structure element *)

pt := ADR (StructInst);            (* Pointer initialization *)
pt^.Out02 := 16#FF00;              (* Access to structure element by pointer *)
```

Fig. 4-60: Examples

Impermissible Addressing

The addressing of a structure element by calculations in the code is **impermissible**:

```
Structure element pointer := ADR (Structure name.Element);
```

```
Structure element pointer := Structure element pointer + n;
```

```
Structure element pointer^ := value;
```

This is impermissible and causes incompatibilities, as it is not ensured, that the offset creation to address a structure element by calculation in the code is reliable. The number of filling bytes is different for the platforms.

Packing Structures for IndraLogic L20

Concerning the IndraLogic L40 structures are always packed, i. e. they are archived in the memory without filling bytes. Contrary to that, with an IndraLogic L20 the packing of structures must be forced by a corresponding compiler instruction (Pragma "pack") during the structure declaration.

The pragma {pack} is supported as of IndraWorks Logic version 02V09.

The pragma {pack} is ignored by the compiler of the IndraLogic L40, i. e. it has no effect on the application's compilability or ability to run in the IndraLogic L40.

Example

```
TYPE Struktur001 :
STRUCT
{pack}
Element_01 : USINT;
      Element_02 : DWORD;
      Element_03 : BYTE;
      Element_04 : INT;
END_STRUCT
END_TYPE
```

Fig. 4-61: Example for the compilation instruction "pack"

The structure of Fig. 4-61 is archived in the memory of the different systems as follows:

Byte address	IndraLogic L40	IndraLogic L20 without pragma {pack}	IndraLogic L20 with pragma {pack}
0	Element_01	Element_01	Element_01
1	Element_02 (byte 0)	Filling byte	Element_02 (byte 0)
2	Element_02 (byte 1)	Filling byte	Element_02 (byte 1)
3	Element_02 (byte 2)	Filling byte	Element_02 (byte 2)
4	Element_02 (byte 3)	Element_02 (byte 0)	Element_02 (byte 3)
5	Element_03	Element_02 (byte 1)	Element_03
6	Element_04 (byte 0)	Element_02 (byte 2)	Element_04 (byte 0)
7	Element_04 (byte 1)	Element_02 (byte 3)	Element_04 (byte 1)
8		Element_03	
9		Filling byte	
10		Element_04 (byte 0)	
11		Element_04 (byte 1)	

Fig. 4-62: Memory assignment

By specifying the pragma {pack} a structure archive compatible to the IndraLogic L40 is created. The individual structure elements can be accessed like in the IndraLogic L40.

```
VAR
      str001: struct001;
      len_struct001: INT := 0;
      array001: ARRAY [0..15] OF BYTE;
      ps001: POINTER TO BYTE;
      pa001: POINTER TO BYTE;
      i: INT := 0;
END_VAR
-------------------------------------------------------------
len_struct001:=SIZEOF(str001);

str001.s001_byte1:=16#01;
str001.s001_DW:=16#05040302;
str001.s001_byte2:=16#06;
str001.s001_Word:=16#0807;

ps001:=ADR (str001);
pa001:=ADR (array001);

FOR i :=0 TO len_struct001 BY 1 DO
      pa001^:=ps001^;
      pa001:=pa001+1;
      ps001:=ps001+1;
END_FOR;
```

Fig. 4-63: Example to pack structures

Note: To pack structures with the help of the compiler instruction {pack} causes a deceleration of the PLC program processing in the IndraLogic L20, as the access to word or double-word operands occurs bytewise.

Alignment for I/O Addresses

The processor of the IndraLogic L20 supports a so-called "Natural Alignment". That is, the memory accesses are optimized in a manner, that the variables are archived on addresses corresponding to the data capacity of the data type.

The following table shows the factor for the optimum start address for the particular data type:

Data type	Factor = Data capacity (bytes)
BYTE	1
WORD	2
DWORD	4

Fig. 4-64: Factor for start addresses

During the declaration of variables **without** assignment of an I/O address the IndraLogic compiler ensures an optimum alignment corresponding to the illustrated table.

Example

```
Variable1: WORD   → even address
Variable2: DWORD  → address divisible by 4
```

During the declaration of variables **with** assignment of an I/O address the alignment in the memory depends on the assigned address.

```
Out01 AT %QB0 : DWORD;  → even address
Out02 AT %QB5 : DWORD;  → uneven address
```

Note: To assign word/byte variables to uneven I/O addresses causes a deceleration of the PLC program processing in the IndraLogic L20, as the access to word or double-word operands occurs bytewise.

Assignment of Structures to I/O Addresses

A structure declared with the pragma instruction {pack} (see section "Packing Structures for IndraLogic L20" on page 4-50) can be directly mapped on a corresponding I/O area.

A non-packed structure causes a wrong addressing within the control because of the filling bytes.

4.13 Download and Commissioning

Loading Configurations and PLC Program into the Control

Verify to Exclude Errors

After finishing the programming the project can be verified in IndraLogic with menu function **Project – Rebuild all**.

Download and Online Mode

With menu function "Online, Login" " the communication between programming system and the IndraLogic L20 is started, and a change to the online mode occurs.

If the current project was not compiled after opening it or after the last modification, it is compiled now (as for **Project – Build**). If errors occur during the compilation, IndraLogic does not change to the online mode.

If the current project was modified after the last download into the control, but not closed, and if the last download information was not deleted with command **Project - Clean all**, a dialog with the following request is opened:

"The program has been changed. Load changes? (Online Change)"

By answering <Yes> you confirm during logging in, that the changed parts of the project are to be loaded into the control. With <Load all> the complete project is reloaded into the control. With <No> a logging in occurs, but the changes made after the last download are not loaded into the control.

For this, refer also to the "Online Functions" in the IndraLogic documentation or help /2/.

Online Functions

With IndraLogic information about the status of the control can be retrieved with the help of the "Online Functions", see IndraLogic documentation or help /2/.

Notes

5 Additional Functions

5.1 Firmware Management

Note: Firmware downloads can be only executed, if the control is in the stop status.

1. Copy the firmware files of the IndraLogic L20 required for the download (*.fw files) to folder "...\Rexroth\IndraWorks\IndraLogic\Targets".
2. Open the dialog of the firmware management. Use the **Firmware management** menu item in the context menu of the IndraLogic L20 device (in the project explorer).

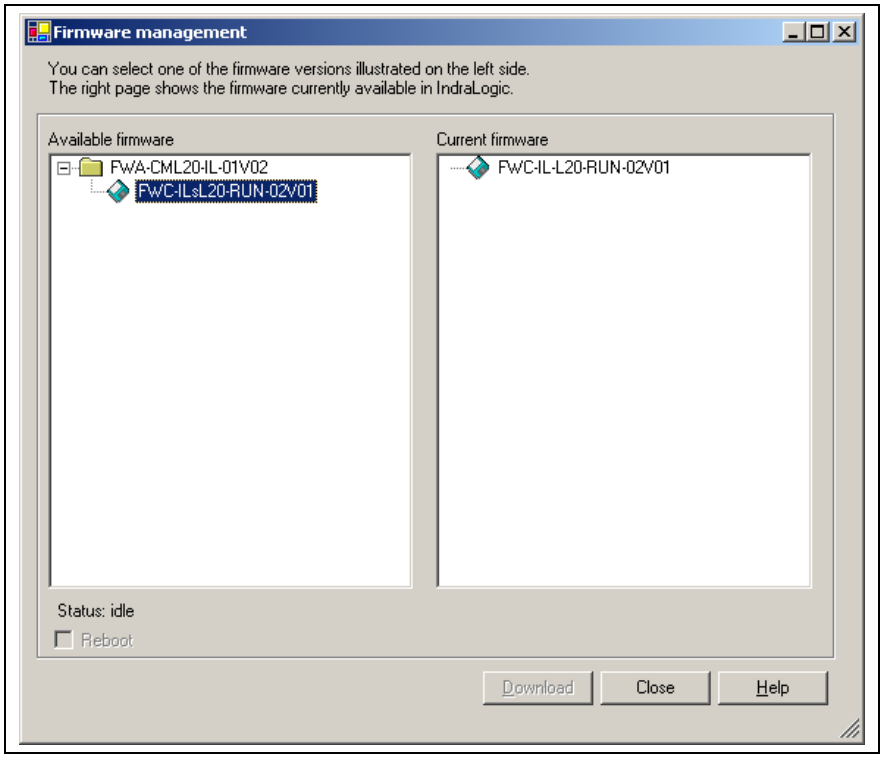


Fig. 5-1: Example for the "Firmware management" dialog

3. Select in the left area the desired firmware and confirm your selection by pressing the **Download** button. Follow the further screen instructions.

CAUTION

Risk to damage the device due to voltage breakdown during firmware change!

⇒ Ensure that the supply voltage of all participating devices is never interrupted during the firmware change!

For further information refer to the documentation or online help of IndraWorks /5/.

5.2 Importing IndraLogic Project Data

Via the context menu of the device project data (e. g. POUs, global data, control configurations) can be imported from an existing IndraLogic project file.

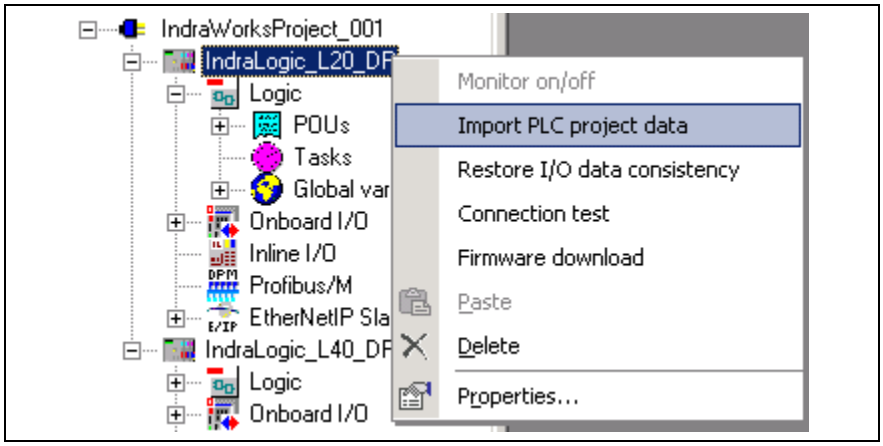


Fig. 5-2: Context menu of the IndraLogic L20: Import PLC project data

For further information refer to the documentation or online help of IndraWorks /5/.

5.3 IndraLogic Functions

The context menu functionality of the "Logic" object in the project explorer is dependent of the current system status:

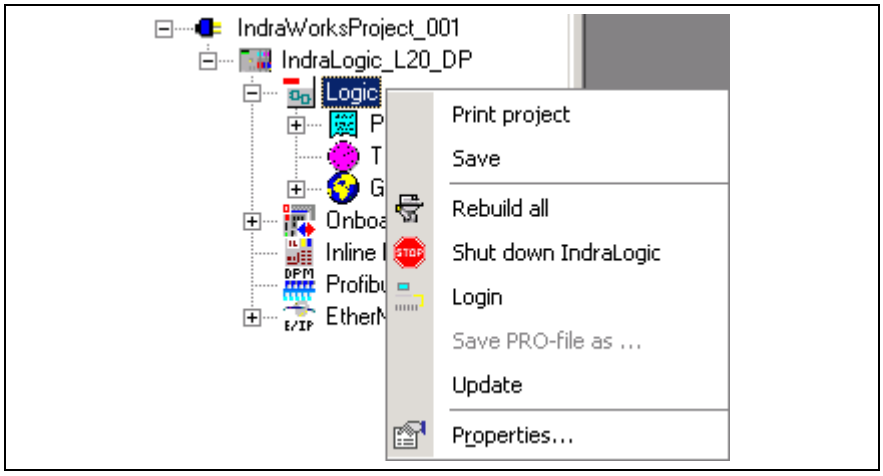


Fig. 5-3: Context menu of the "Logic" object when the IndraLogic is **started**

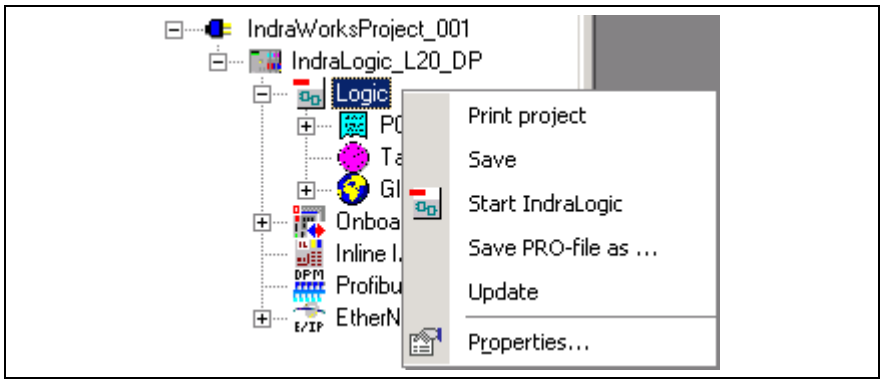


Fig. 5-4: Context menu of the "Logic" object when the IndraLogic is **not started**

Available functions:

- Printing, saving and compiling an IndraLogic project (**Rebuild all**)
- Starting and closing IndraLogic
- Logging the control in or out
- Updating all modules created in IndraLogic as well as modifications of configuration settings in the IndraWorks project explorer.

Note: The access on the context menu items **Save PRO file as...** and **Properties** is only possible, if IndraLogic is closed.

Further functions and settings of IndraLogic you will find in the "Further settings" tab:

- Saving IndraLogic settings
- Safety mode
- Offline operation
- Generating and sending symbol file
- Replacing constants
- Address verification
- Optimized IndraLogic call
- Automatic loading of the boot project

The properties can be called up via the context menu of the "Logic" object, if IndraLogic is not started.

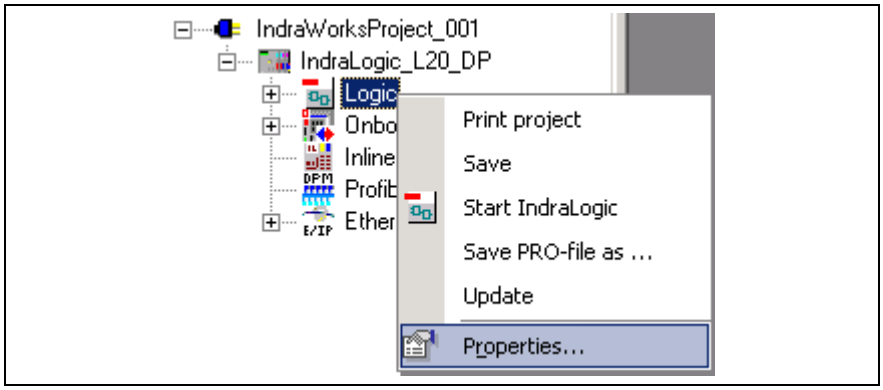


Fig. 5-5: Context menu for the IndraLogic L20: further settings

For further information refer to the documentation or online help of IndraWorks /5/.

Notes

6 Libraries

6.1 Overview

Libraries available for the IndraLogic L20:

Library	Function	See
AnalyzationNew	Analysis of expressions	/2/
BuepE_Client	Communication between the IndraLogic L20 and the Bosch controls of the CL series	In this chapter
HMI_Mkeys	M key functions of the HMI (IndraWorks HMI)	/5/
Iecsfc	For internal use	–
IL_VCP_DP	Coupling the Rexroth VCP small operator terminal with Profibus connection	In this chapter
MP_PLCopen	MotionControl modules	/7/
NetVarUdp_LIB_V23	Application of network variables and the parameter manager (for data exchange between two or several controls)	Network_ Functionality.pdf [1]
PLCOpenFieldBus	MotionControl on the basis of the PLCopen	/7/
ProViDiagnosis	Access to the PLC diagnostics (ProVi)	/5/
RIH_CML20	Modules about the status of the IndraLogic L20	In this chapter
RIL_Check	Automatic monitoring of internal over-ranges	In this chapter
RIL_CheckRtv	Like RIL_Check	See RIL_Check
RIL_CommonTypes	For internal use	–
RIL_EthernetIP	Functions in connection with Ethernet/IP	In this chapter
RIL_L20_Util	Do no longer use modules about the status of the IndraLogic L20 (comparable functions in RIH_CML20)	In this chapter
RIL_ProfibusDP	Profibus DPV1 services, diagnostic interface between Profibus master and PLC program, sync and freeze	In this chapter
RIL_Utilities	General IL functions and function blocks	In this chapter
RIL_VExUtil	Safe key transmission to an HMI device	In this chapter
Default	Default FBs and functions of the IEC 61131-3	/2/
SysLibCom	Serial communication with the IndraLogic L20	SysLibCom.pdf [1]
SysLibFile	File system support on the IndraLogic L20	SysLibFile.pdf [1]
SysLibFileAsync	Asynchronous file accesses from the IEC application	SysLibFileAsync.pdf [1]
SysLibIecTasks	Management of IEC tasks	SysLibIecTasks.pdf [1]
SysLibMem	Memory management	SysLibCom.pdf [1]
SysLibPlcCtrl	Start, stop and reset of the control	SysLibPlcCtrl.pdf [1]
SysLibRtc	Access to the real-time clock and the battery status of the IndraLogic L20	SysLibRtc.pdf [1]
SysLibSockets	Access to sockets for the communication via TCP/IP and UDP	SysLibSockets.pdf [1]
SysLibSocketsAsync		
SysLibStr	Character string functions	SysLibStr.pdf [1]

[1] You will find the documentation about the system libraries (SysLibXXX.pdf) in a sub-directory of your IndraLogic installation, e. g.: ".../Rexroth/IndraWorks/IndraLogic/Documents/English"

Library	Function	See
SysLibTime	Retrieve time and date	SysLibTime.pdf [1]
Util	As supplement to the Default.lib: BCD converting, bit/byte functions, mathematic help functions, controllers, signal generators, function manipulators and analog value processing	/2/

Fig. 6-1: PLC programming with IndraLogic

To link libraries with the current project, please consider section "Library Manager" on page 4-46 in chapter "PLC Programming with IndraLogic".

Note: IndraWorks loads libraries for internal use automatically as soon as they are referenced. It is not necessary to insert them manually with the library manager.

Concerning the system and firmware libraries please also consider the online help of IndraLogic.

6.2 BuepE_Client

General Information

The BuepE_Client library serves for the communication of the IndraLogic L20 with the Bosch controls of the CL series. The IndraLogic L20 maps the client functionality. The CL control represents always the server.

BuepE_Client

The BuepE_Client function block can be simultaneously used several times. A new instance is required for every order.

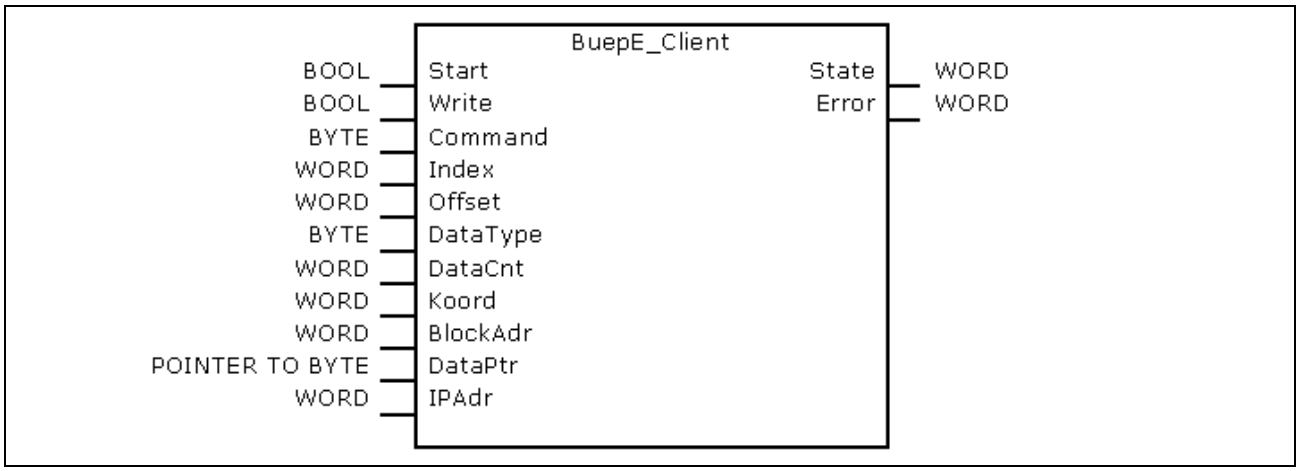


Fig. 6-2: BuepE_Client

	Name	Type	Comment
VAR_INPUT	Start	BOOL	TRUE: The transmission is started
	Write	BOOL	TRUE: Writing access FALSE: Reading access
	Command	BYTE	Field type of the Bosch control: 16#44: Data block 16#43: Data field 16#4D: Flag
	Index	WORD	Number of the data block (only for the "Data block" field type)
	Offset	WORD	Byte offset address within the selected data field or data block
	DataType	BYTE	Data type: 0: BYTE 1: WORD
	DataCnt	WORD	Number of the data = f(DataType), depending on the setting in DataType
	Koord	WORD	Coordination flag 0: Uncoordinated
	BlockAdr	WORD	Module block address, only for CL200, CL400 and CL500 16#FFFF: without block address
	DataPtr	POINTER TO BYTE	Pointer to the source data (for "Write"=TRUE), or pointer to the target data (for "Write"=FALSE)
	IPAdr	DWORD	IP address of the Bosch CL control

	Name	Type	Comment
VAR_OUTPUT	State	WORD	Status: 0: Without errors 16#0100: Order in process 16#0305: Order faulty 16#0405: Order not started Further states about the communication functionality of the particular control are described in the documentation.
	Error	WORD	Error: 0: Without error 16#FF04: No order in process 16#FF20: No UDP socket free (e. g. too many active instances at the same time) 16#FF21: Error during "bind" (e. g. too many active instances at the same time) 16#FF22: Error during "sendto" (e. g. wrong IP address or partner not activated) 16#FF23: Error during "UDP-receive" 16#FF24: Faulty UDP receive length 16#FF25: Too many repetitions (partner does not respond, e. g. wrong IP address Further states about the communication functionality of the particular control are described in the documentation .

Fig. 6-3: Interface of BuepE_Client

The following program extract shows an example of the BuepE_Client call.

```
VAR
  Bclient : BuepE_Client;              (* Declare instance *)
  MyDB : ARRAY [0..511] OF BYTE;       (* Data block-data array*)
END_VAR
(* Example: Read a DB from a CL500-ZS1 *)
Bclient(
  Start := TRUE,                       (* Start transmission *)
  Write := FALSE,                      (* Reading access)
  Command := 16#44,                    (* Data block *)
  Index := 0,                          (* DB number *)
  Offset := 0,                         (* No address offset *)
  DataType := 0,                       (* Byte *)
  DataCnt := 512,                      (* 512 bytes *)
  Koord := 0,                          (* Uncoordinated *)
  BlockAdr := 16#0008,                 (* ZS 1 *)
  DataPtr := ADR(MyDB),                (* Pointer to data block-data array *)
IPAdr := SysSockInetAddr('10.110.244.46'),
(* State=> ,*)
(* Error=>*) );
```

Fig. 6-4: Example for a BuepE_Client call

6.3 IL_VCP_DP

Overview

With this library the small operator terminals "VCP" with Profibus connection are coupled to the PLC. For this, three function blocks can be alternatively used:

- VCP_PBS16_A4096
- VCP_PBS32_A4096
- VCP_PBS32_A65536

VCP_PBS16_A4096

This function block (FB) activates the Profibus DP protocol for the small operator terminals VCPxx. Additionally, the I/O image of the physical addresses between PLC and operator terminal is transmitted.

The data capacity of the data transmission is 16 bytes. The size of the address area available via an ARRAY is 4096 bytes (inputs and outputs included).

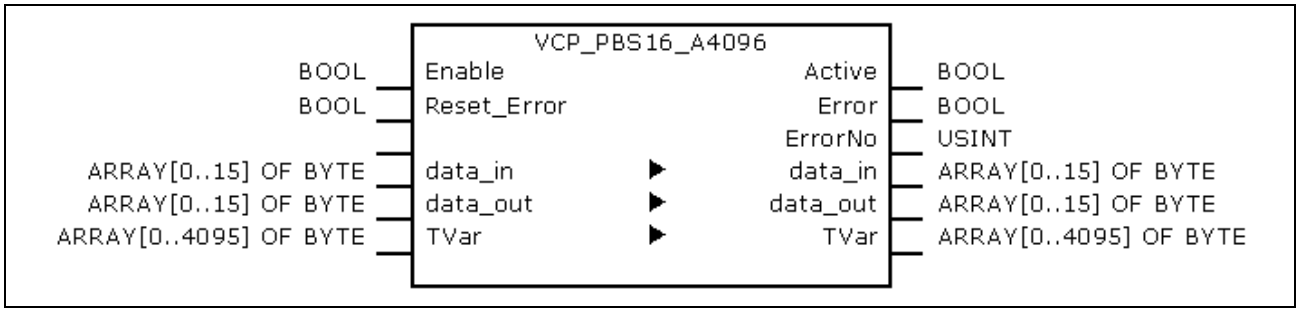


Fig. 6-5: VCP_PBS16_A4096

	Name	Type	Comment
VAR_INPUT	Enable	BOOL	TRUE: FB is executed FALSE: FB is not executed
	Reset_Error	BOOL	TRUE: Reset "Error" (to FALSE) and "ErrorNo" is set to 0.
VAR_IN_OUT	Data_in	ARRAY [0..15] OF BYTE	Data to connect the physical inputs of the small operator terminal
	Data_out	ARRAY [0..15] OF BYTE	Data to connect the physical outputs of the small operator terminal
	TVar	ARRAY [0..4095] OF BYTE	Array to read from and write to the small operator terminal
VAR_OUTPUT	Active	BOOL	TRUE, as long as "Enable" is also TRUE
	Error	BOOL	TRUE, when an error occurs. Can be reset with "Reset-Error".
	ErrorNo	USINT	Error type: 2: InputRangeError 4: Calculation error

Fig. 6-6: Interface of VCP_PBS16_A4096

VI-Composer During the configuration with the Rexroth VI-Composer the addresses of the variable list refer to the particular byte in array "TVar" that is used in the PLC program to exchange data.

Error Handling

As soon as an error occurs, the communication is interrupted and the display of the small operator terminal indicates "COMMUNICATION ERROR, ERROR CODE 110".

The error type (ErrorNo) shows, if there's an error in the address calculation (CalculationError), or if the selected data capacity for Data_in or Data_out is too high (InputRangeError).

VCP_PBS32_A4096

This function block (FB) activates the Profibus DP protocol for the small operator terminals VCPxx. Additionally, the I/O image of the physical addresses between PLC and operator terminal is transmitted.

The data capacity of the data transmission is 32 bytes. The size of the address area available via an ARRAY is 4096 bytes (inputs and outputs included).

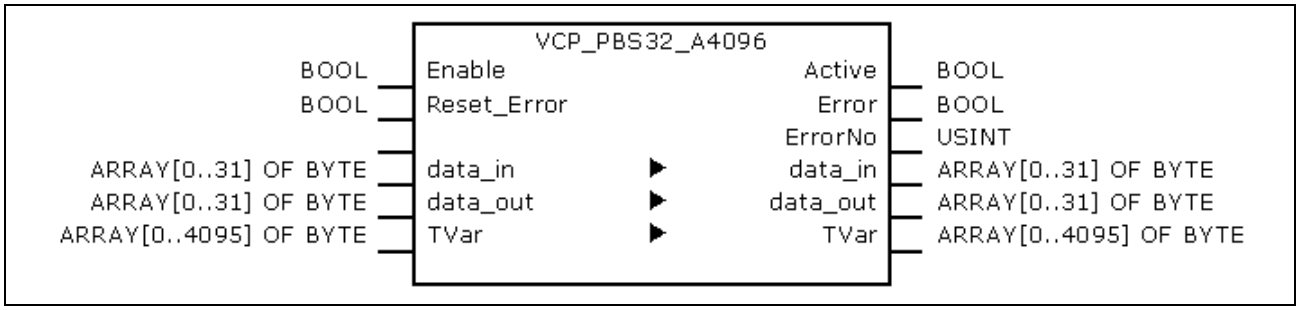


Fig. 6-7: VCP_PBS32_A4096

	Name	Type	Comment
VAR_INPUT	Enable	BOOL	TRUE: FB is executed FALSE: FB is not executed
	Reset_Error	BOOL	TRUE: Reset "Error" (to FALSE) and "ErrorNo" is set to 0.
VAR_IN_OUT	Data_in	ARRAY [0..31] OF BYTE	Data to connect the physical inputs of the small operator terminal
	Data_out	ARRAY [0..31] OF BYTE	Data to connect the physical outputs of the small operator terminal
	TVar	ARRAY [0..4095] OF BYTE	Array to read from and write to the small operator terminal
VAR_OUTPUT	Active	BOOL	TRUE, as long as "Enable" is also TRUE
	Error	BOOL	TRUE, when an error occurs. Can be reset with "Reset-Error".
	ErrorNo	USINT	Error type: 4: Calculation error

Fig. 6-8: Interface of VCP_PBS32_A4096

VI-Composer

During the configuration with the Rexroth VI-Composer the addresses of the variable list refer to the particular byte in array "TVar" that is used in the PLC program to exchange data.

Error Handling

As soon as an error occurs, the communication is interrupted and the display of the small operator terminal indicates "COMMUNICATION ERROR, ERROR CODE 110".

The error type (ErrorNo) indicates that the error is an address calculation error (CalculationError).

VCP_PBS32_A65536

This function block (FB) activates the Profibus DP protocol for the small operator terminals VCPxx. Additionally, the I/O image of the physical addresses between PLC and operator terminal is transmitted.

The data capacity of the data transmission is 32 bytes. The size of the address area available via an ARRAY is 65536 bytes (inputs and outputs included).

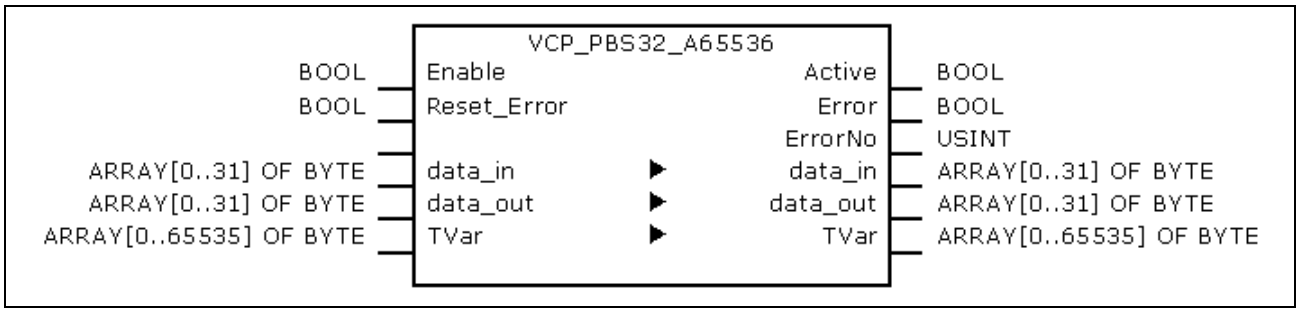


Fig. 6-9: VCP_PBS32_A65536

	Name	Type	Comment
VAR_INPUT	Enable	BOOL	TRUE: FB is executed FALSE: FB is not executed
	Reset_Error	BOOL	TRUE: Reset "Error" (to FALSE) and "ErrorNo" is set to 0.
VAR_IN_OUT	Data_in	ARRAY [0..31] OF BYTE	Data to connect the physical inputs of the small operator terminal
	Data_out	ARRAY [0..31] OF BYTE	Data to connect the physical outputs of the small operator terminal
	TVar	ARRAY [0..65535] OF BYTE	Array to read from and write to the small operator terminal
VAR_OUTPUT	Active	BOOL	TRUE, as long as "Enable" is also TRUE
	Error	BOOL	TRUE, when an error occurs. Can be reset with "Reset-Error".
	ErrorNo	USINT	Error type: 4: Calculation error

Fig. 6-10: Interface of VCP_PBS32_A65536

VI-Composer

During the configuration with the Rexroth VI-Composer the addresses of the variable list refer to the particular byte in array "TVar" that is used in the PLC program to exchange data.

Error Handling

As soon as an error occurs, the communication is interrupted and the display of the small operator terminal indicates "COMMUNICATION ERROR, ERROR CODE 110".

The error type (ErrorNo) indicates that the error is an address calculation error (CalculationError).

6.4 RIH_CML20

Overview

- IH_GetOhcCtrl: operating hours counter of the control
- IH_SetDisplay: user indications on the display
- IH_Temperature: current internal temperature of the control
- IH_TempWarning: over-temperature warning

IH_GetOhcCtrl

Supplies the operating time of the control in hours.

Note: It is not possible to reset the operating hours counter.

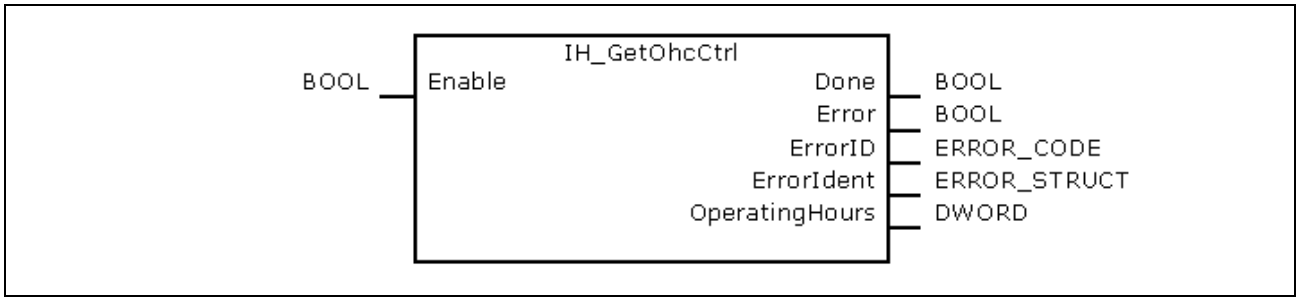


Fig. 6-11: IH_GetOhcCtrl

	Name	Type	Comment
VAR_INPUT	Enable	BOOL	TRUE: FB is executed FALSE: FB is not executed
VAR_OUTPUT	Done	BOOL	TRUE: OperatingHours relevant FALSE: OperatingHours in process
	Error	BOOL	Always 0
	ErrorID	ERROR_CODE	Undefined and cannot be evaluated
	ErrorIdent	ERROR_STRUCT	Undefined and cannot be evaluated
	OperatingHours	DWORD	Operating time of the control in hours

Fig. 6-12: Interface of IH_GetOhcCtrl

IH_SetDisplay

Indicates a user message of up to 80 characters on the LCD display of the control.

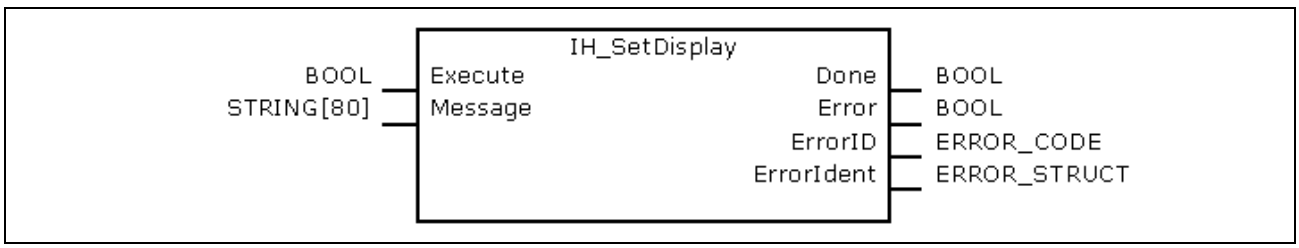


Fig. 6-13: IH_SetDisplay

	Name	Type	Comment
VAR_INPUT	Execute	BOOL	With a positive edge the output of the string transmitted at input Message is started.
	Message	STRING[80]	String with up to 80 characters to be output at the display. Empty string: Delete the active display.
VAR_OUTPUT	Done	BOOL	TRUE: String is output to the display FALSE: Processing not yet finished
	Error	BOOL	Always 0
	ErrorID	ERROR_CODE	Undefined and cannot be evaluated
	ErrorIdent	ERROR_STRUCT	Undefined and cannot be evaluated

Fig. 6-14: Interface of IH_SetDisplay

If there is no error message, an active message is indicated in the default display (Stop/Run) by the flashing illustration of "IL:" .

To read the message, switch with the <Up>/<Down> keys to the "IL display", see Fig. 7-4. The message is then output at the control display in a rotating manner with preceded "IL:" .

After transmitting an empty string the flashing display "IL:" is cleared.

IH_Temperature

Determines the internal temperature of the control (replaces the IL_GetTemp function from the RIL_L20_Util library).

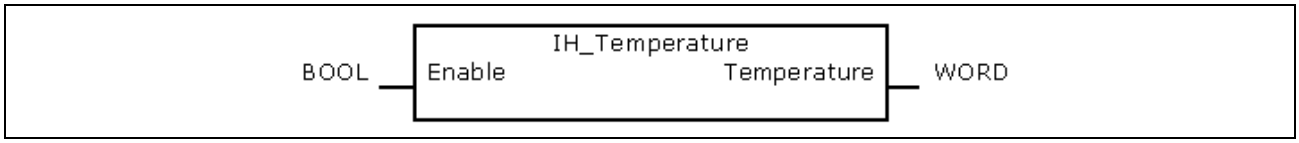


Fig. 6-15: IH_Temperature

	Name	Type	Comment
VAR_INPUT	Enable	BOOL	TRUE: The current temperature is output FALSE: No output
Function value	Temperature	WORD	Temperature: Bit 15: 0 = positive temperature 1 = negative temperature Bits 14 – 8: Temperature value in degree Celsius Bit 7: 1 = decimal place (half degree Celsius) 0 = no decimal place

Fig. 6-16: Interface of IH_Temperature

Examples:

Function value	Temperature
2#00010100_10000000	+ 20,5 °C
2#01001011_00000000	+ 75,0 °C
2#10000101_10000000	- 5,5 °C

Fig. 6-17: Examples for IH_Temperature

IH_TempWarning

Determines, if the internal temperature of the control has exceeded the critical value of **70 °C** (replaces the IL_TempWarning function from the RIL_L20_Util library).

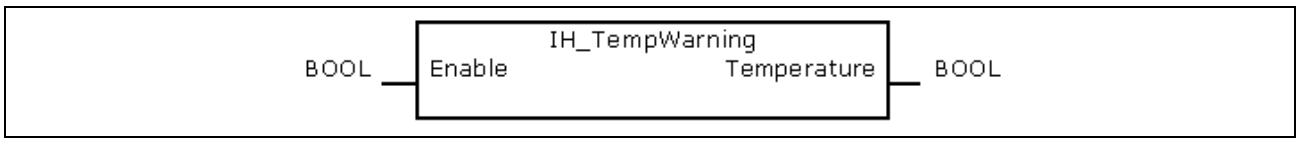


Fig. 6-18: IH_TempWarning

	Name	Type	Comment
VAR_INPUT	Enable	BOOL	TRUE: Output value is valid. FALSE: Output value is not representative.
Function value	Temperature	BOOL	TRUE: Temperature warning FALSE: No temperature warning

Fig. 6-19: Interface of IH_TempWarning

Note: If the internal temperature achieves **80°C**, the control switches automatically into operating mode "Stop". The outputs enter the safe status, and the warning “Temp !!!” appears on the display. This mode can only be quit by switching the power supply off/on.

6.5 RIL_Check

Faulty accesses (accidentally) programmed outside of the ranges of arrays and subrange types of variables, as well as the division by zero, are not compensated by the IndraLogic compiler and/or the runtime system and cause partly unpredictable errors during program processing. If you insert the RIL_Check.lib such over-ranges are monitored and prevented. Thereby, the access for arrays and subrange types are limited to the smallest or highest possible value. Thus, e. g. concerning an array, the element with the highest indices is accessed, even if a higher index value was specified in the PLC program. For a division by zero the divisor is replaced by "1".

Note: If the RIL_Check.lib is integrated in the PLC project, before each testable operation the corresponding test function is automatically inserted (invisible for the user).

Further function calls are not necessary!

Note: If the RIL_Check.lib is integrated in the PLC project, the PLC cycle time is charged, as each testable operation is automatically monitored. With respect to the robustness of a PLC project and the safety of the whole system we recommend to use the RIL_Check.lib in the PLC project.

When using the MP_PLCOpen.lib, the RIL_Check.lib is required. In this case the RIL_Check.lib must be integrated in the PLC project.

Functions All functions contained in the RIL_Check library (see Fig. 6-20) are automatically integrated in the PLC program and must not be explicitly called.

Designation	Description
CheckBounds	Automatic verification, if the permissible indices of the access to array elements is exceeded or fallen below the minimum value.
CheckDivByte	Automatic verification on division by zero (BYTE access).
CheckDivDWord	Automatic verification on division by zero (DWORD access).
CheckDivReal	Automatic verification on division by zero (REAL access).
CheckDivWord	Automatic verification on division by zero (WORD access).
CheckRangeSigned	Automatic verification, if the permissible value range of a signed variable is exceeded or fallen below the minimum value.
CheckRangeUnsigned	Automatic verification, if the permissible value range of an unsigned variable is exceeded or fallen below the minimum value.

Fig. 6-20: Functions of RIL_Check

CheckExceedingOccurred

You can exactly determine the error cause by directed requesting of the bits in the global variable "CheckExceedingOccurred". All error accesses are prevented by RIL_Check, so that the permissible value range is not exceeded or fallen below the minimum value and that no division by zero occurs. The single bits have the following meaning:

Bit variable	Value	Meaning
CheckExceedingOccurred.0	16#01	CheckBoundsLowerLimitation: The permissible indices of the access to array elements are fallen below the minimum value.
CheckExceedingOccurred.1	16#02	CheckBoundsUpperLimitation: The permissible indices of the access to array elements are exceeded.
CheckExceedingOccurred.2	16#04	CheckBoundsExceedingLimitation: The permissible indices of the access to array elements are exceeded or fallen below the minimum value.
CheckExceedingOccurred.3	16#08	CheckRangeLowerLimitation: The permissible value range of a variable is fallen below the minimum value.
CheckExceedingOccurred.4	16#16	CheckRangeUpperLimitation: The permissible value range of a variable is exceeded.
CheckExceedingOccurred.5	16#32	CheckRangeExceedingLimitation: The permissible value range of a variable is exceeded or fallen below the minimum value.
CheckExceedingOccurred.6	16#64	DivisionByZeroPrevention: Division by zero.

Fig. 6-21: CheckExceedingOccurred

Sample Program

The following PLC sample program shows the use of variable CheckExceedingOccurred. The limitation of arrays with CheckBounds always sets the respective bit in error case, when the permissible value range is exceeded or fallen below the minimum value (CheckBoundsLowerLimitation or CheckBoundsUpperLimitation) and the general bit of the access violation (CheckBoundsExceedingLimitation). Thus, a general or detailed verification, if a limit value was exceeded, can be realized.

```
CheckExceedingOccurred := 16#00;       (* Resetting the variable *)

IF Axis_Data[AxisNo].bCheckAccessOK    (* Array access *)
THEN
...

IF CheckExceedingOccurred.2 (* Array access violation detected? *)
THEN
          IF CheckExceedingOccurred.0 (* Bit set, if value below specified range? *)
          THEN
String := 'Access below the possible array range'
                  ...

          END_IF
          IF CheckExceedingOccurred.0 (* Bit set, if value above specified range? *)
          THEN
String := 'Access above the possible array range'
                  ...

END_IF
ELSE
String := 'Access successful'
END_IF
```

Fig. 6-22: Sample program: over-range for arrays

6.6 RIL_EtherNetIP

Overview

An IndraLogic L20 configured as CIP data client is able to read and – if write access is allowed – to write on variables of a CIP data server.

IL_ReadDataTable	FB	Read variables provided by the CIP data server
IL_WriteDataTable	FB	Write on variables provided by the CIP data server
IL_Status	FB	For diagnostics of the cyclic communication

Fig. 6-23: Function blocks contained in the RIL_EtherNetIP.lib

IL_ReadDataTable

Brief Description Reads variables provided by a CIP data server.

Interface Description

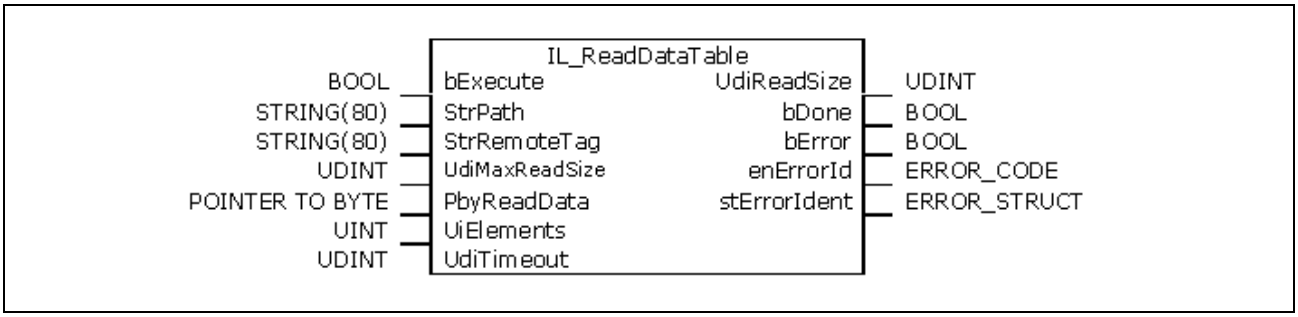


Fig. 6-24: Structure of IL_ReadDataTable

	Name	Type	Comment
VAR_INPUT	bExecute	BOOL	Positive edge starts the service.
	StrPath	STRING(80)	IP address of the CIP data server (e. g. "192.168.73.105")
	StrRemoteTag	STRING(80)	Name of the variable to be read (e. g. "Test1").
	UdiMaxReadSize	UDINT	Size of the receiver array "PbyReadData".
	PbyReadData	POINTER TO BYTE	Pointer to the array, in which the data are to be written. The array has to be provided by the user of the FB.
	UiElements	UINT	Number of array elements to be read. For atomic data types (SINT, DINT...) UiElements has to be set to value 1.
	UdiTimeout	UDINT	Specifies the maximum time in ms that the CIP data server may need to respond.
VAR_OUTPUT	UdiReadSize	UDINT	Number of received data bytes.
	bDone	BOOL	TRUE: The service is stopped or an error occurred.
	bError	BOOL	TRUE: An error occurred.
	enErrorID	ERROR_CODE	See error messages.
	stErrorIdent	ERROR_STRUCT	

Fig. 6-25: Interface signals of IL_ReadDataTable

Example Read the DINT variable "diCounter" from the CIP data server with the IP address "192.168.73.105". As "diCounter" is used in the PLC program "PLC_PRG", you have to enter "PLC_PRG.diCounter" as name of the variable to be read.

The reading process can be started by a positive edge change of variable "Read2Fb_bExecute" (FALSE -> TRUE). Status TRUE of variable "bDone" confirms the execution of the service.

```
PROGRAM PLC_PRG
VAR


(*********************************************
** Variables for IL_ReadDataTable
*********************************************)
  Read2In1:                 IL_ReadDataTable
  Read2Fb_bExecute:         BOOL;
  Read2Fb_strPath:          STRING;
  Read2Fb_strRemoteTag:     STRING;
  Read2Fb_udiMaxReadSize:   UDINT;
Read2Fb_diReadData:       DINT;
  Read2Fb_uiElements:       UINT;
  Read2Fb_udiTimeout:       UDINT;
  Read2Fb_udiReadSize:      UDINT;
  Read2Fb_bDone:            BOOL := FALSE;
  Read2Fb_bError:           BOOL;
  Read2Fb_enErrorID:        ERROR_CODE;        (*Type: RIL_CommonTypes.lib*)
  Read2Fb_stErrorIdent:     ERROR_STRUCT;      (*Type: RIL_CommonTypes.lib*)

  (*Diagnostics*)
  Read2Fb_udiCtrErr:         UDINT := 0;
  Read2Fb_udiCtrGood:        UDINT := 0;
  Read2Fb_CtrValidInput:     UDINT := 0;

    (*for automatic test*)
  Read2Fb_bAutoTest:         BOOL := FALSE;
```

Fig. 6-26: Example: IL_ReadDataTable, client-side variable definition

```
(*****************************************************************
** Explicit message: Read data table - 2 DINT
*****************************************************************)

  Read2Fb_strPath      := '192.168.73.105';      (*IP address of the data table server*)
  Read2Fb_strRemoteTag := 'PLC_PRG.diCounter';   (*Name of the variable to be read*)
  Read2Fb_uiElements   := 1;                     (*Number of elements to be read*)
  Read2Fb_udiTimeout   := 100;                   (*Timeout in ms*)


Read2In1(
  bExecute       := Read2Fb_bExecute,
  strPath        := Read2Fb_strPath,
  strRemoteTag   := Read2Fb_strRemoteTag,
  udiMaxReadSize := SIZEOF(Read2Fb_diReadData),
  pbyReadData    := ADR(Read2Fb_diReadData),
  uiElements     := Read2Fb_uiElements,
  udiTimeout     := Read2Fb_udiTimeout,
  udiReadSize    => Read2Fb_udiReadSize,
  bDone          => Read2Fb_bDone,
  bError         => Read2Fb_bError,
  enErrorID      => Read2Fb_enErrorID,
  stErrorIdent   => Read2Fb_stErrorIdent);


(*automatic test*)
IF Read2Fb_bAutoTest = TRUE THEN
  IF Read2Fb_bDone = TRUE THEN
          IF Read2Fb_bError = TRUE THEN
                  Read2Fb_udiCtrErr  := Read2Fb_udiCtrErr + 1;
          ELSE
                  Read2Fb_udiCtrGood := Read2Fb_udiCtrGood + 1;
          END_IF
          Read2Fb_bExecute := FALSE;
  ELSE
          Read2Fb_bExecute := TRUE;
  END_IF
END_IF
```

Fig. 6-27: Example: IL_ReadDataTable, client-side PLC program

IL_WriteDataTable

Brief Description Writes values in variables provided by a CIP data server.

Note: Write access at the CIP data server has to be allowed for the relevant variables.

Interface Description

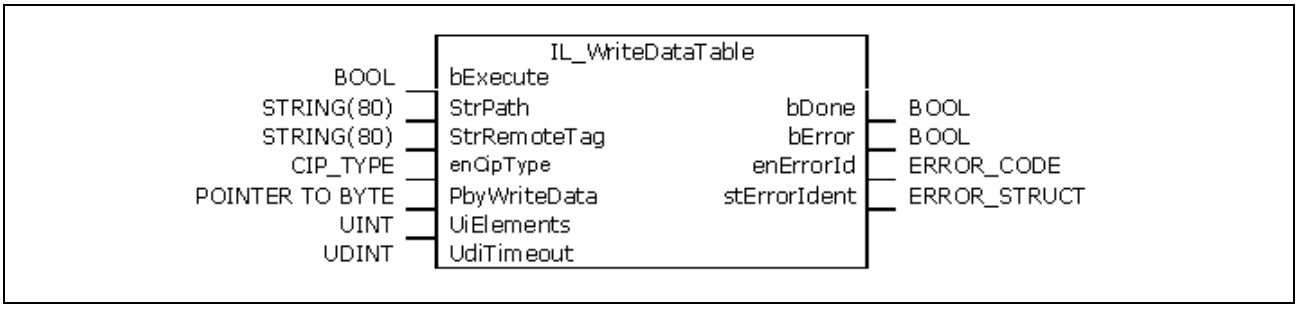


Fig. 6-28: Structure of IL_WriteDataTable

	Name	Type	Comment
VAR_INPUT	bExecute	BOOL	Positive edge starts the service.
	StrPath	STRING(80)	IP address of the CIP data server (e. g. "192.168.73.105")
	StrRemoteTag	STRING(80)	Name of the variable to be written (e. g. "Test1").
	EnCipType	CIP_TYPE	Type of the written data. Is verified by some CIP data server.
	PbyWriteData	POINTER TO BYTE	Pointer to the array that contains the data to be written on the CIP data client. The array has to be provided by the user of the FB.
	UiElements	UINT	Number of array elements to be written. For atomic data types (SINT, DINT...) UiElements has to be set to value 1.
	UdiTimeout	UDINT	Specifies the maximum time in ms that the CIP data server may need to respond.
VAR_OUTPUT	bDone	BOOL	TRUE: The service is stopped or an error occurred.
	bError	BOOL	TRUE: An error occurred.
	EnErrorID	ERROR_CODE	See error messages.
	StErrorIdent	ERROR_STRUCT	

Fig. 6-29: Interface signals of IL_WriteDataTable

Example Write the SINT variable "siWriteData" to the CIP data server with the IP address "192.168.73.105". As "siWriteData" is used in the PLC program "PLC_PRG", you have to enter "PLC_PRG.siWriteData" as name of the variable to be written.

The writing process can be started by a positive edge change of variable "WriteFb_bExecute" (FALSE -> TRUE). Status TRUE of variable "bDone" confirms the execution of the service.

```
(*********************************************
** Variables for Writing Data SINT
*********************************************)

    WriteIn1:                 IL_WriteDataTable;

  WriteFb_bExecute:         BOOL;
  WriteFb_strPath:          STRING;
  WriteFb_strRemoteTag:     STRING;
WriteFb_enCipType:        CIP_TYPE;
  WriteFb_uiElements:       UINT;
  WriteFb_siWriteData:      SINT;
  WriteFb_udiTimeout:       UDINT;
  WriteFb_bDone:            BOOL := FALSE;
  WriteFb_bError:           BOOL;
  WriteFb_enErrorID:        ERROR_CODE;
  WriteFb_stErrorIdent:     ERROR_STRUCT;

  (*Diagnostics*)
WriteFb_udiCtrErr:        UDINT := 0;
WriteFb_udiCtrGood:       UDINT := 0;
  WriteFb_udiCtrValidInput: UDINT := 0;

    (*For automatic test*)
  WriteFb_bAutoTest:        BOOL := FALSE;
```

Fig. 6-30: Example: IL_WriteDataTable, client-side variable definition

```
(*****************************************************************
** Explicit message: Write data table - 1 SINT
*****************************************************************)

  WriteFb_strPath               := '192.168.73.105';
  WriteFb_strRemoteTag          := 'PLC_PRG.siWriteData';
  WriteFb_uiElements            := 1;
  WriteFb_enCipType             := CIPTYPE_SINT;
  WriteFb_udiTimeout            := 1000;  (*ms*)

WriteIn1(
  bExecute              := WriteFb_bExecute,          (*FALSE FB starts after TRUE*)
  strPath               := WriteFb_strPath,           (*IP address of the server*)
  strRemoteTag          := WriteFb_strRemoteTag,      (*Name of the variable to be written*)
  enCipType             := WriteFb_enCipType,         (*Data type of the variables*)
  uiElements            := WriteFb_uiElements,        (*Number of array elements to be written.*)
                                                            (* For atomic data types *)
                                                            (* (SINT, DINT...)always 1.*)
  pbyWriteData   := ADR(WriteFb_siWriteData),          (*pointer to the data, you want to write*)
  udiTimeout            := WriteFb_udiTimeout,         (*Timeout of the Explicit Message*)
  bDone                 => WriteFb_bDone,              (*TRUE => FB is ready with order.*)
  bError                => WriteFb_bError,             (*TRUE => Error*)
  enErrorID             => WriteFb_enErrorID,
  stErrorIdent   => WriteFb_stErrorIdent);

(*automatic test*)
IF WriteFb_bAutoTest = TRUE THEN
  IF WriteFb_bDone = TRUE THEN
          IF WriteFb_bError = TRUE THEN
                  WriteFb_udiCtrErr  := WriteFb_udiCtrErr + 1;
          ELSE
                  WriteFb_udiCtrGood := WriteFb_udiCtrGood + 1;

                  WriteFb_siWriteData   := WriteFb_siWriteData + 1;
          END_IF
          WriteFb_bExecute := FALSE;
  ELSE
          WriteFb_bExecute := TRUE;
  END_IF
END_IF
```

Fig. 6-31: Example: IL_WriteDataTable, client-side PLC program

Error Messages of IL_ReadDataTable and IL_WriteDataTable

Error ID	Error table	Additional1	Additional2	Error name	Description
2	151	0x20000001	0x00000000	InvalidState	State machine reached invalid internal status.
2	151	0x20000002	0x00000000	UnKnownIndex	Internal system error.
2	151	0x20000003	0x00000000	Timeout	The service could not be executed within the specified time.
2	151	0x000000XX	0xXXXXXXXX	Remote Error	Error code was generated by the CIP data server. See configuration of the CIP data server.

IL_Status

The FB IL_Status serves to make a diagnostics of the cyclic communication ("Implicit Messaging"), see also chapter "Configuring IndraLogic L20 as Ethernet/IP Slave" as of page 4-35.

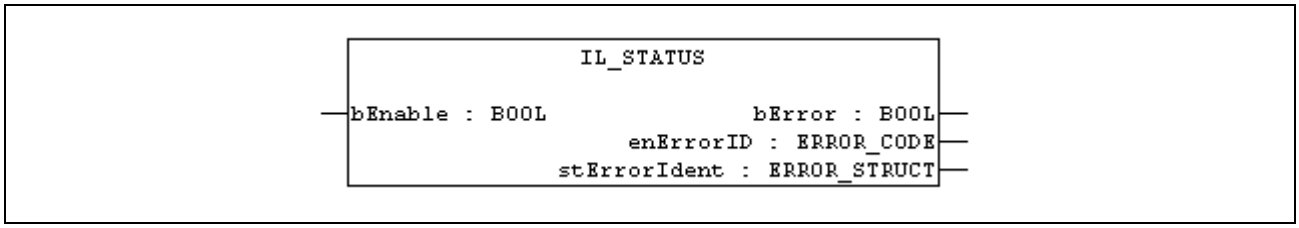


Fig. 6-32: IL_Status

	Name	Type	Comment
VAR_INPUT	bEnable	BOOL	If the status is positive (TRUE), the status of the cyclic EtherNet/IP communication is requested with each call of the FB.
VAR_OUTPUT	bError	BOOL	TRUE: An error occurred.
	EnErrorID	ERROR_CODE	See error messages.
	StErrorIdent	ERROR_STRUCT	

Fig. 6-33: Interface of IL_WriteDataTable

Example

The following example shows how to use FB IL_Status.

```
(*************************************************
** Variables for IL_Status
*************************************************)
PROGRAM PLC_PRG
VAR
  Diag:          IL_Status;
 CtrValidInput: DINT
  ...
END_VAR;  ...
  ...
  ...
(*************************************************
** Program
*************************************************)
Diag(bEnable:=TRUE);
IF(Diag.bError = TRUE) THEN
  (*Insert here the error handling.*)
...

  Diag(bEnable:=FALSE ); (*Reset error (bEnable)*)
ELSE
  (*Valid data: Insert here input and output data processing *)
  CtrValidInput := CtrValidInput +1;
END_IF
...
```

Fig. 6-34: Example: IL_Status

Error Messages of IL_Status

Error ID	Error table	Additional1	Additional2	Error name	Description
COMMUNICATION_ ERROR (2)	ETHERNET_IP(151)	0x10000001	0x00000000	NoCyclicCom	Master has not yet established the cyclic communication with the IndraLogic EtherNet/IP slave.
COMMUNICATION_ ERROR (2)	ETHERNET_IP(151)	0x10000002	0x00000000	Idle	EtherNet/IP master has started cyclic communication, but set the idle flag in the cyclic channel. Thus, the master indicates that its output image is not valid.
COMMUNICATION_ ERROR (2)	ETHERNET_IP(151)	0x10000003	0x00000000	Timeout	Time monitoring of the cyclic communication indicates errors.
COMMUNICATION_ ERROR (2)	ETHERNET_IP(151)	0x10000004	0x00000000	Closed	Zyclic communication was actively finished by the EtherNet/IP master.

Fig. 6-35: Error messages of IL_ReadDataTable and IL_WriteDataTable

6.7 RIL_L20_Util

The functions IL_GetTemp and IL_TempWarning provided in the previous RIL_L20_Util library are replaced by the functions IH_Temperature and IH_TempWarning in the RIH_CML20 library (see chapter "RIH_CML20" as of page 6-8).

6.8 RIL_ProfibusDP

Overview

When using this library, not only DPV1 services of the Profibus master (DP master class 1) are provided, but also a diagnostic interface between Profibus master and PLC program is created. Additionally, sync and freeze control commands can be realized.

DPV1 Services

Function Blocks

The function blocks serve for reading and writing access for the acyclic data exchange (DPV1):

- DP_RDREC
- DP_WRREC

Functions

Moreover, help functions for addressing are available:

- DP_ADDR
- DP_ID
- DP_SLOT

Diagnostic Information

Diagnostic information can be determined via function blocks. The diagnostics differs in

- Slave Diagnostic Data According to the Profibus DP Standard: "DP_RDIAG" and "DP_RDIAG_EXT"
- General Field Bus Diagnostics: "fbd..."

Data Types

Some of these data are administrated in special data types (arrays, structures):

- tFBD_BM_INFO
- tFBD_BIT_LIST
- tFBD_KSD_LIST

Function Blocks

- DP_RDIAG
- DP_RDIAG_EXT
- fbdBaudrateGet
- fbdBmErrorGet
- fbdBmInfoGet
- fbdBmStateGet
- fbdKsdListGet
- fbdPdTypeGet
- fbdPrjSlaveListGet
- fbdSlaveDiagListGet

Functions Moreover, help functions for addressing are available:

- DP_ADDR
- DP_ID
- DP_SLOT

Sync and Freeze

Control commands to synchronize inputs or outputs.

- DP_SYCFR

Functions Moreover, help functions for addressing are available:

- DP_ADDR
- DP_ID
- DP_SLOT

Slave Diagnostic Data According to the Profibus DP Standard

Profibus-specific diagnostic information according to the Profibus DP standard can be read via the following function blocks:

- DP_RDIAG
- DP_RDIAG_EXT

The slave diagnostic data are available with the parameters DINFO or DINFO_PTR. While for "DINFO" all diagnostic data are archived in an array, "DINFO_PTR" is a pointer to the diagnostic data.

The slave diagnostic data are subdivided into two parts:

- general part with a fixed length of 6 bytes
- extended diagnostics (slave-specific, with variable length)

Offset	Type	Designation	Description
0	BYTE	Station status_1	See below
1	BYTE	Station status_2	See below
2	BYTE	Station status_3	See below
3	BYTE	Master_Add	Bus address of the master having parameterized the slave
4	WORD	Ident_Number	Ident_Number of the slave
6 – 243		Ext_Diag_Data	Extended diagnostics according to Profibus DP standard

Fig. 6-36: Slave diagnostic data

The following description of station status 1 to 3 is an extract from the Profibus DP standard.

Bit	Designation	Description
7	Master_Lock	The DP slave was parameterized by another master. This bit is set by the DP master (class 1), if the address in octet 4 is not equal to 255 and not equal to the own address. The DP slave sets this bit permanently to zero.
6	Prm_Fault	This bit is set by the DP slave, if the last parameter telegram was faulty, e. g., wrong length, wrong Ident_Number, invalid parameters.
5	Invalid_Slave_Response	This bit is set by the DP master, as soon as one of the addressed DP slaves receives an implausible response. The DP slave sets this bit permanently to zero.
4	Not_Supported	This bit is set by the DP slave, as soon as a function was requested, that is not supported by this DP slave.
3	Ext_Diag	This bit is set by the DP slave. If the bit is set, a diagnostic entry must be available in the slave-specific diagnostic area (Ext_Diag_Data). If the bit is not set, a status message might be present in the slave-specific diagnostic area (Ext_Diag_Data). The meaning of this status message must be defined according to the respective application.
2	Cfg_Fault	This bit is set by the DP slave, as soon as the configuration data lastly received by the DP master do not correspond to the data determined by the DP slave.
1	Station_Not_Ready	This bit is set by the DP slave, if the DP slave is not yet ready for data exchange.
0	Station_Non_Existent	This bit is set by the DP master, if this DP slave cannot be reached via the bus. If this bit is set, the diagnostic bits contain the status of the last diagnostic message or the initial value. The DP slave sets this bit permanently to zero.

Fig. 6-37: Station status_1

Bit	Designation	Description
7	Deactivated	This bit is set by the DP master, as soon as the DP slave in the DP slave parameter set is not indicated as active and was removed from the cyclic processing. The DP slave sets this bit permanently to zero.
6	Reserved	-
5	Sync_Mode	This bit is set by the DP slave, as soon as it received the sync control command.
4	Freeze_Mode	This bit is set by the DP slave, as soon as it received the freeze control command.
3	WD_On (Watchdog)	This bit is set by the DP slave, as soon as its response monitoring is activated.
2	1	The DP slave sets this bit permanently to zero.
1	Stat_Diag (static diagnostics)	If the DP slave sets this bit, the DP master has to retrieve diagnostic information as long as this bit is deleted again. The DP slave sets this bit, for example in the case, if it can't provide any valid user data.
0	Prm_Req	If the DP slave sets this bit, it must be reparameterized and reconfigured. The bit remains set, as long as a parameterization occurred. This bit is set by the DP slave.

Fig. 6-38: Station status_2

Bit	Designation	Description
7	Ext_Diag_Overflow	If this bit is set, there're more diagnostic information as specified in Ext_Diag_Data. The DP slave sets this bit, for example, if there're more channel diagnostics than the DP slave can enter in its transmitter buffer; or the DP master sets this bit, if the DP slave sends more diagnostic information than the DP master is able to enter in its diagnostic buffer.
6	Reserved	-
5	Reserved	-
4	Reserved	-
3	Reserved	-
2	Reserved	-
1	Reserved	-
0	Reserved	-

Fig. 6-39: Station status_3

You will find further information on the diagnostic functions in the description of the Function Blocks as of page 6-30.

General Field Bus Diagnostics

The general field bus diagnostics provides the following data areas:

- BmState: Bus master status word, see "tFBD_BM_INFO"
- BmError: Bus master error word, see "tFBD_BM_INFO"
- SD: Slave diagnostic list, see "tFBD_BIT_LIST"
- KSD: Classified slave diagnostics, see "tFBD_KSD_LIST"
- Projected slave list, see "tFBD_BIT_LIST"

Bus Master Status Word "BmState" provides an overview of the status of the bus master and the slaves at the field bus. Here, you can recognize for example, if at least for one slave a diagnostics is applied.

Bus Master Error Word In "BmError" fatal errors rendering the operation at the field bus impossible are indicated.

Slave Diagnostic List SD indicates which slaves signal diagnostics.

Classified Slave Diagnostics KSD provides detailed information. It indicates, which slaves signal diagnostics and divides the diagnostics into error classes.

Projected Slave List The projected slave list contains all available slaves according to the master configuration file. For example, with this list projected slaves can be compared with the slaves currently available at the field bus.

You will find further information on the data areas in the description of the Data Types as of page 6-26 and on the diagnostic functions in the description of the Function Blocks as of page 6-30.

Data Types

tFBD_BM_INFO

This structure combines the variables "PdType", "BmStatus" and "BmError". It is used in function block "fbdBmInfoGet".

```
0001 TYPE tFBD_BM_INFO :
0002 STRUCT
0003     PdType  : INT;
0004     BmState : WORD;
0005     BmError : WORD;
0006 END_STRUCT
0007 END_TYPE
```

Fig. 6-40: tFBD_BM_INFO

PdType: Peripheral Driver

Function block "fbdBmInfoGet" determines the type of the installed peripheral driver and indicates it in "PdType":

Driver	Description
FBD_PDT_NONE	No peripheral driver installed
FBD_PDT_PCI_BMDP	Peripheral driver: Profibus DP
FBD_PDT_PCI_BMCAN	Peripheral driver: CAN Open (at present, not available)
FBD_PDT_PCI_BMIBS	Peripheral driver: Interbus (at present, not available)

Fig. 6-41: Installed peripheral driver in "PdType"

BmState: Bus Master Status Word

The bus master status word "BmState" provides an overview of the status of the bus master and the slaves at the field bus. Here, you can recognize for example, if at least for one slave a diagnostics is applied. Each set bit (TRUE) in "BmState" represents a status:

Bit	Status	Description
0	BMS_BMF	Bus master error: This bit indicates, that there's a bus master error. In this case, the bus master error word contains more detailed information.
1	BMS_KSD	Classified slave diagnostics: If this bit is set, at least one slave indicates a classified diagnostics. Which classified diagnostics is set, can be determined by the bits 8 to 13.
2	BMS_SD	Slave diagnostics: If this bit is set, at least one slave indicates a slave diagnostics.
3	-	- Reserved -
4	-	- Reserved -
5	-	- Reserved -
6	-	- Reserved -
7	BMS_AKTIV	Active identification: This bit must always have value 1. If this is not the case, there's a fatal error in the software of the bus master.
8	BMS_SNE	One or several slaves are not accessible via the bus.
9	BMS_SKF	One or several slaves indicate configuration errors.
10	BMS_DPS	One or several slaves indicate static diagnostics.

Bit	Status	Description
11	BMS_EXD	One or several slaves indicate extended diagnostics.
12	BMS_SNB	One or several slaves are not ready for the cyclic data exchange.
13	BMS_SF	One or several slaves indicate another error.
14	-	- Reserved -
15	-	- Reserved -

Fig. 6-42: Status coding in "BmState"

BmState is used in the following function blocks:

- fbdBmInfoGet
- fbdBmStateGet

BmError: Bus Master Error Word

In the bus master error word "BmError" fatal errors rendering the operation at the field bus impossible are indicated. Each set bit (TRUE) in "BmError" represents an error:

Bit	Error	Description
0	BMF_HW_ERR	Hardware error
1	BMF_MPS_ERR	Master parameter set (field bus configuration file) is missing or faulty
2	BMF_BUS_ERR	Error at the field bus (e. g. short-circuit, ...)
3	BMF_SW_ERR	System error in the peripheral driver (i. e. the driver software has detected a fatal error)

* "BMF_OK" indicates, that no error occurred

Fig. 6-43: Error coding in "BmError"

BmError is used in the following function blocks:

- fbdBmErrorGet
- fbdBmInfoGet

tFBD_BIT_LIST

The bit list "tFBD_BIT_LIST" has a defined length of 16 bytes (128 bits).

```
0001 TYPE tFBD_BIT_LIST :
0002     ARRAY [0..15] OF BYTE;
0003 END_TYPE
```

Fig. 6-44: tFBD_BIT_LIST

Each bit of the bit list is assigned to a bus address of the slave (Profibus: FDL address). Thus, e. g. the lowest-order bit in the first array element (ARRAY[0]) is assigned to the Profibus device with address 0:

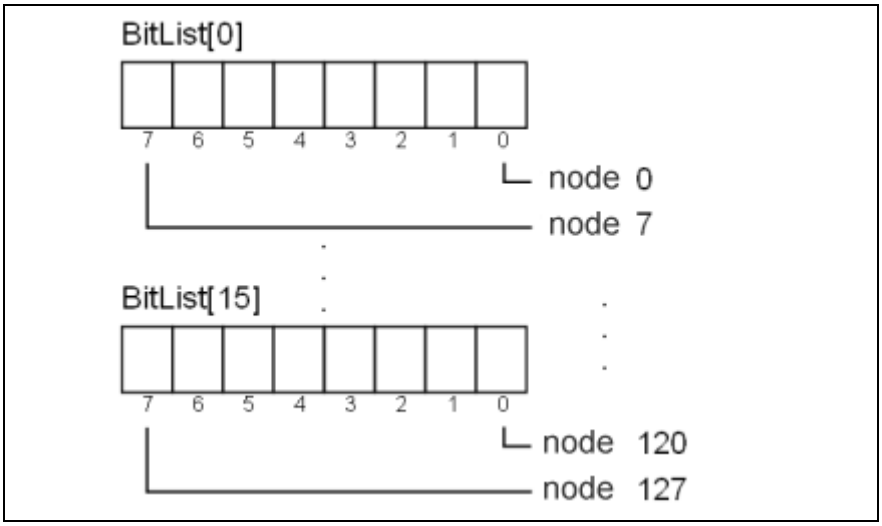


Fig. 6-45: Coding of the bit list

The bit list is used in the following function block:

- fbdPrjSlaveListGet
- fbdSlaveDiagListGet

Additionally, the bit list is used in the tFBD_KSD_LIST (classified slave diagnostics).

tFBD_KSD_LIST

The list of the classified slave diagnostics (KSD list) consists of six bit lists. Thus, a bit list exists for each error type.

```
0001 TYPE tFBD_KSD_LIST :
0002     STRUCT
0003         SNE : tFBD_BIT_LIST;
0004         SKF : tFBD_BIT_LIST;
0005         DPS : tFBD_BIT_LIST;
0006         EXD : tFBD_BIT_LIST;
0007         SNB : tFBD_BIT_LIST;
0008         SF  : tFBD_BIT_LIST;
0009     END_STRUCT
0010 END_TYPE
```

Fig. 6-46: tFBD_KSD_LIST

Classified Slave Diagnostics (KSD)

The classified slave diagnostics distinguishes the following error types:

SNE	Slave not accessible. The slave is not accessible at the bus. Possible causes: – Slave not available – Voltage at the slave switched off – Faulty bus installation – Physical malfunctions
SKF	Slave configuration error. The slave type or the I/O configuration of the slave does not correspond to the projected values in the field bus configuration file of the master.
DPS	Slave indicates static diagnostics: The slave can't provide valid user data. The application layer of the slave is not ready for data exchange with the master.
EXD	Slave indicates extended diagnostics. The extended diagnostics is slave-specific and can be taken from the description of the slave. Possible causes: – Load voltage missing/switched off (e. g. in case of Emergency Stop) – Short-circuit at one output – Overload – Over-temperature – Line break
SNB	Slave is not ready. The slave is not ready for data exchange, as it is not yet put into operation by the master (message from the protocol layer of the slave).
SF	Slave indicates another error.

Fig. 6-47: Classified slave diagnostics

Each bit of a bit list is assigned to a bus address of the slave (Profibus: FDL address). Thus, e. g. the lowest-order bit in the first array element (ARRAY[0]) is assigned to the Profibus device with address 0:

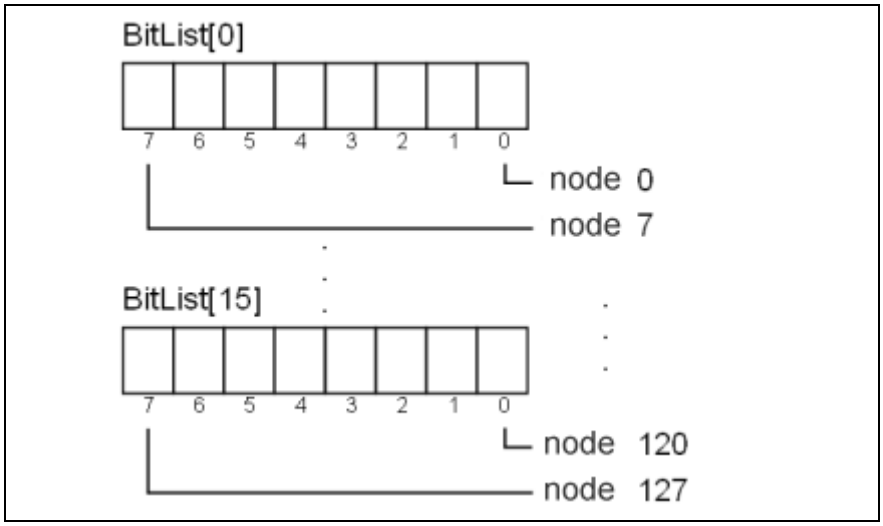


Fig. 6-48: Coding of the bit list

Bit[x] = FALSE	Slave[x] has no diagnostics
Bit[x] = TRUE	Slave[x] has diagnostics

Fig. 6-49: Bit list: Classified slave diagnostics

Function Blocks

DP_RDIAG

The diagnostic data of a slave are read from the DP master (DPM1) using the DP_RDIAG function block. The data buffer of the diagnostic data must be provided to address it via a POINTER.

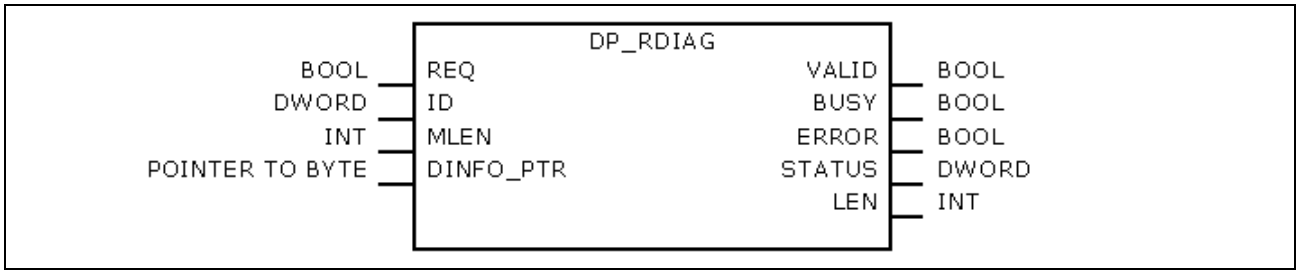


Fig. 6-50: DP_RDIAG

	Name	Type	Comment
VAR_INPUT	REQ	BOOL	TRUE: Start reception
	ID	DWORD	Slot handle, see the following table
	MLEN	INT	Maximum length of the data to be read
	DINFO_PTR	POINTER TO BYTE	Pointer to the data buffer of the Slave Diagnostic Data According to the Profibus DP Standard
VAR_OUTPUT	VALID	BOOL	TRUE: New, valid diagnostic data available
	ERROR	BOOL	TRUE: Error/s occurred
	BUSY	BOOL	TRUE: The function block is busy. As long as BUSY = TRUE, the data cannot yet be evaluated.
	STATUS	DWORD	Lastly defined status
	LEN	INT	Length of the diagnostic data in bytes

Fig. 6-51: Interface of DP_RDIAG

Bytes	Contents	Description
0	MASTER	ID of the DP system: Has permanently value 0, as the IndraLogic L20 contains exactly one DP master.
1	SEGMENT	Number of the DP segment
2	STATION	Number of the DP slave (bus address)
3	SLOT	Number of the slot within the slave

Fig. 6-52: Slot handle: parameter "ID"

Example: To address the slave with bus address 12, the ID has value 16#000C0000.

The DP_ID function serves for creating the ID from the individual components.

DP_RDIAG_EXT

The diagnostic data of a slave are read from the DP master (DPM1) using DP_RDIAG_EXT function block. The diagnostic data are stored in an ARRAY.

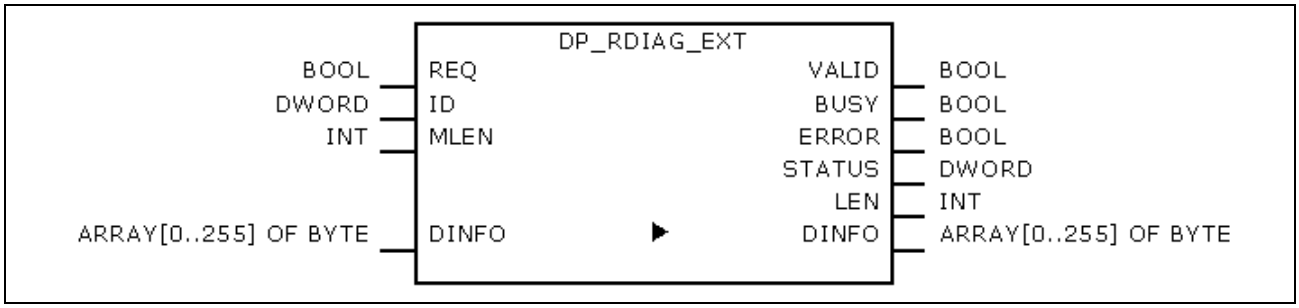


Fig. 6-53: DP_RDIAG_EXT

	Name	Type	Comment
VAR_INPUT	REQ	BOOL	TRUE: Start reception
	ID	DWORD	Slot handle, see the following table
	MLEN	INT	Maximum length of the data to be read
VAR_IN_OUT	DINFO	ARRAY [0..255] OF BYTE	Slave Diagnostic Data According to the Profibus DP Standard
VAR_OUTPUT	VALID	BOOL	TRUE: New, valid diagnostic data available
	ERROR	BOOL	TRUE: Error/s occurred
	BUSY	BOOL	TRUE: The function block is busy. As long as BUSY = TRUE, the data cannot yet be evaluated.
	STATUS	DWORD	Lastly defined status
	LEN	INT	Length of the diagnostic data in bytes

Fig. 6-54: Interface of DP_RDIAG_EXT

Bytes	Contents	Description
0	MASTER	ID of the DP system: Has permanently value 0, as the IndraLogic L20 contains exactly one DP master.
1	SEGMENT	Number of the DP segment
2	STATION	Number of the DP slave (bus address)
3	SLOT	Number of the slot within the slave

Fig. 6-55: Slot handle: parameter "ID"

Example: To address the slave with bus address 12, the ID has value 16#000C0000.

The DP_ID function serves for creating the ID from the individual components.

DP_RDREC

The DP_RDREC function block serves for reading access for the acyclic data exchange (DPV1). For the process data to be read a target area must be defined via a pointer addressing (POINTER).

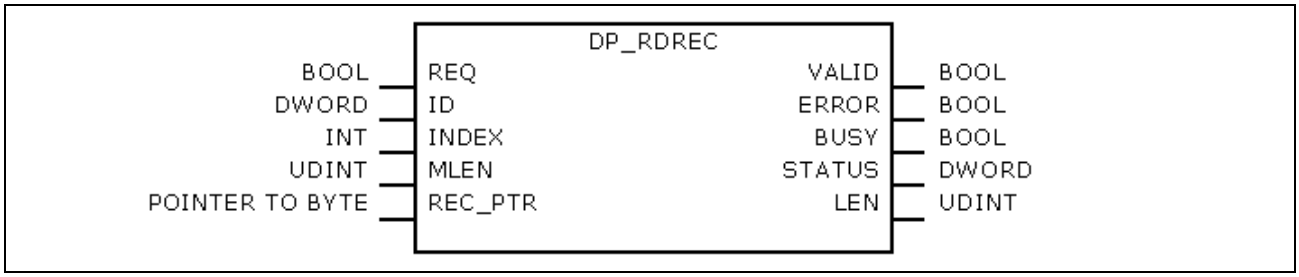


Fig. 6-56: DP_RDREC

	Name	Type	Comment
VAR_INPUT	REQ	BOOL	TRUE: Start reading
	ID	DWORD	Slot handle, see the following table
	INDEX	INT	Index of the process data (offset)
	MLEN	UDINT	Maximum length of the process data in bytes
	REC_PTR	POINTER TO BYTE	Pointer to the target area
VAR_OUTPUT	VALID	BOOL	TRUE: New, valid data available
	ERROR	BOOL	TRUE: Error/s occurred
	BUSY	BOOL	TRUE: The function block is busy. As long as BUSY = TRUE, the data cannot yet be evaluated.
	STATUS	DWORD	Lastly defined status
	LEN	UDINT	Length of the process data in bytes

Fig. 6-57: Interface of DP_RDREC

Bytes	Contents	Description
0	MASTER	ID of the DP system: Has permanently value 0, as the IndraLogic L20 contains exactly one DP master.
1	SEGMENT	Number of the DP segment
2	STATION	Number of the DP slave (bus address)
3	SLOT	Number of the slot within the slave

Fig. 6-58: Slot handle: parameter "ID"

Example: To address the slave with bus address 12, the ID has value 16#000C0000.

The DP_ID function serves for creating the ID from the individual components.

DP_WRREC

The DP_WRREC function block serves for writing access for the acyclic data exchange (DPV1). The process data to be written must provided via a pointer addressing (POINTER).

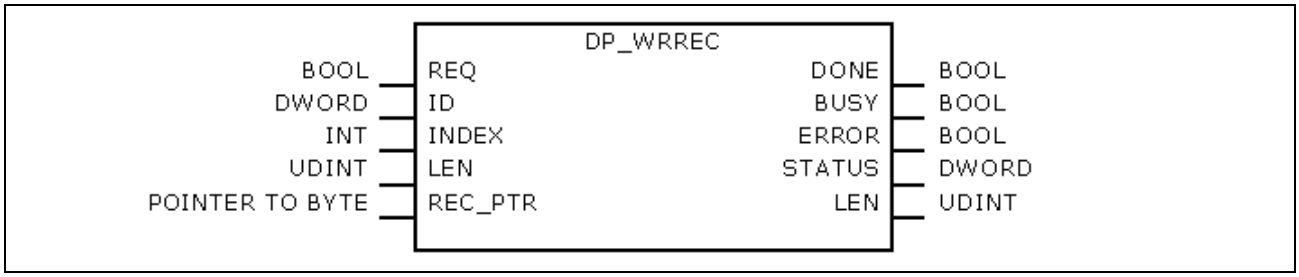


Fig. 6-59: DP_WRREC

	Name	Type	Comment
VAR_INPUT	REQ	BOOL	TRUE: Start writing
	ID	DWORD	Slot handle, see the following table
	INDEX	INT	Index of the process data (offset)
	LEN	UDINT	Length of the process data in bytes
	REC_PTR	POINTER TO BYTE	Pointer to the process data to be written
VAR_OUTPUT	DONE	BOOL	TRUE: Call completed
	ERROR	BOOL	TRUE: Error/s occurred
	BUSY	BOOL	TRUE: The function block is busy. As long as BUSY = TRUE, the data cannot yet be evaluated.
	STATUS	DWORD	Lastly defined status

Fig. 6-60: Interface of DP_WRREC

Bytes	Contents	Description
0	MASTER	ID of the DP system: Has permanently value 0, as the IndraLogic L20 contains exactly one DP master.
1	SEGMENT	Number of the DP segment
2	STATION	Number of the DP slave (bus address)
3	SLOT	Number of the slot within the slave

Fig. 6-61: Slot handle: parameter "ID"

Example: To address the slave with bus address 12, the ID has value 16#000C0000.

The DP_ID function serves for creating the ID from the individual components.

fbdBaudrateGet

The fbdBaudrateGet function block reads the baud rate of the connected field bus. The baud rate is indicated in bits per second.

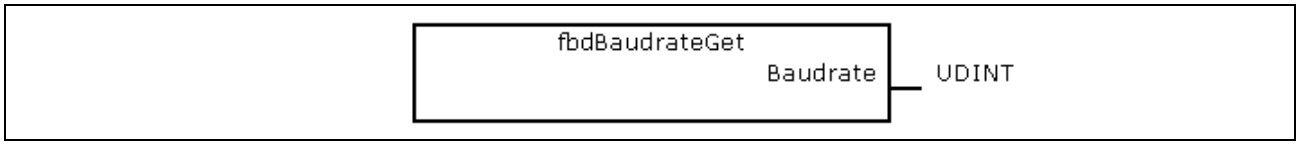


Fig. 6-62: fbdBaudrateGet

	Name	Type	Comment
VAR_OUTPUT	Baud rate	UDINT	Baud rate at the field bus (bit/s)

Fig. 6-63: Interface of fbdBaudrateGet

fbdBmErrorGet

The fbdBmErrorGet function block reads the current bus master error word.

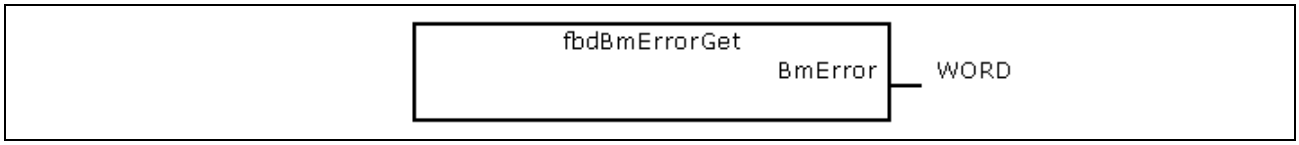


Fig. 6-64: fbdBmErrorGet

	Name	Type	Comment
VAR_OUTPUT	BmError	WORD	See tFBD_BM_INFO

Fig. 6-65: Interface of fbdBmErrorGet

fbdBmInfoGet

The fbdBmInfoGet function block provides the tFBD_BM_INFO structure. This structure contains PdType, BmStatus and BmError. Thus, the information is provided by one single call, instead of calling several differing functions.

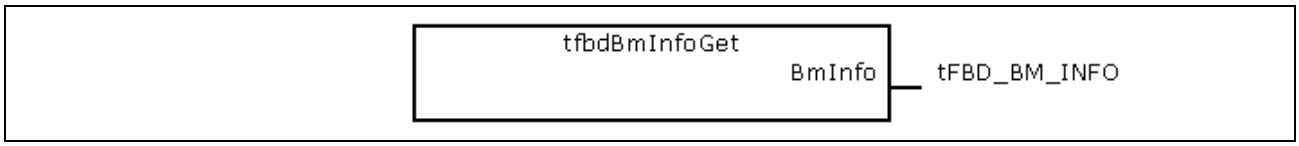


Fig. 6-66: fbdBmInfoGet

	Name	Type	Comment
VAR_OUTPUT	BmInfo	tFBD_BM_INFO	See tFBD_BM_INFO

Fig. 6-67: Interface of fbdBmInfoGet

fbdBmStateGet

The fbdBmStateGet function block reads the current bus master status word.

Fig. 6-68: fbdBmStateGet

	Name	Type	Comment
VAR_OUTPUT	BmState	WORD	See tFBD_BM_INFO

Fig. 6-69: Interface of fbdBmStateGet

fbdKsdListGet

The fbdKsdListGet function block reads the current KSD list.

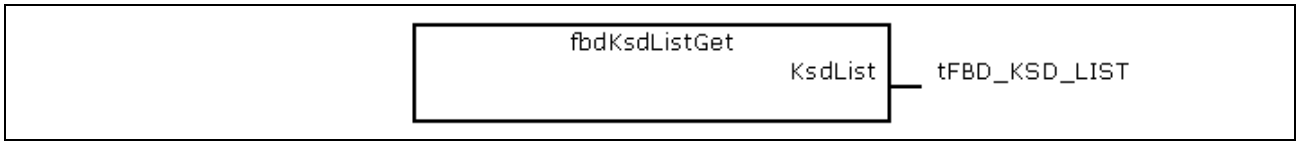


Fig. 6-70: fbdKsdListGet

	Name	Type	Comment
VAR_OUTPUT	KsdList	tFBD_KSD_LIST	See tFBD_KSD_LIST

Fig. 6-71: Interface of fbdKsdListGet

fbdPdTypeGet

The fbdPdTypeGet function block determines the type of the installed peripheral driver.

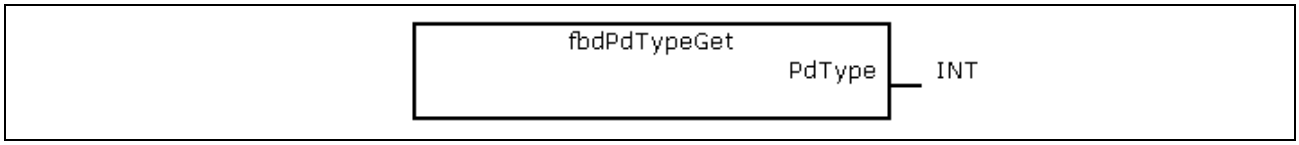


Fig. 6-72: fbdPdTypeGet

	Name	Type	Comment
VAR_OUTPUT	PdType	INT	Peripheral driver type: PDT_NONE: No peripheral driver installed PDT_PCI_BMDP: Profibus DP PDT_PCI_BMCAN: CAN Open PDT_PCI_BMIBS: INTERBUS S

Fig. 6-73: Interface of fbdPdTypeGet

fbdPrjSlaveListGet

Function block fbdPrjSlaveListGet reads the list of the projected slaves. The list contains all available slaves according to the master configuration file.

Fig. 6-74: fbdPrjSlaveListGet

	Name	Type	Comment
VAR_OUTPUT	PrjSlaveList	tFBD_BIT_LIST	See tFBD_BIT_LIST

Fig. 6-75: Interface of fbdPrjSlaveListGet

Each bit of the bit list is assigned to a bus address of the slave:

Bit[x] = TRUE	Slave[x] is projected
Bit[x] = FALSE	Slave[x] is not projected

Fig. 6-76: Bit list: projected slaves

fbdSlaveDiagListGet

The fbdSlaveDiagListGet function block reads the current slave diagnostic list.

Fig. 6-77: fbdSlaveDiagListGet

	Name	Type	Comment
VAR_OUTPUT	SlaveDiagList	tFBD_BIT_LIST	See tFBD_BIT_LIST

Fig. 6-78: Interface of fbdSlaveDiagListGet

Each bit of the bit list is assigned to a bus address of the slave:

Bit[x] = TRUE	Slave[x] has diagnostics
Bit[x] = FALSE	Slave[x] has no diagnostics

Fig. 6-79: Bit list: slave diagnostics

DP_SYCFR

Using the DP_SYCFR function block control commands to synchronize inputs or outputs can be realized. Thus, e. g. the outputs of several drives can be synchronized (e. g. Rexroth EcoDrive, IndraDrive), several axes can start at the same time, etc.

The basis for this is the possibility, that a Profibus DP master can send a "Global Control Telegram" to a defined slave or to complete slave groups. One of the following control commands can be transmitted with the Global Control Telegram:

- **Freeze** (control command code: **16#08**): Causes that all addressed slaves (see following table, input parameter "ID" and "GROUP) switch into the freeze mode. During the transition in this status a slave "freezes" the current status of its **input data**. The input data is one time updated when the next freeze command arrives.
- **Unfreeze** (control command code: **16#04**): Causes that all addressed slaves, terminate the freeze mode.
- **Sync** (control command code: **16#20**): Causes that all addressed slaves, switch into the sync mode. During the transition in this status a slave "freezes" the current status of its **outputs** corresponding to its present output image. The outputs are one time updated corresponding to the present output image when the next sync command arrives.
- **Unsync** (control command code: **16#10**): Causes that all addressed slaves, update their outputs corresponding to the internal present output image and terminate the sync mode.

Note: In order that the control commands of the sync and freeze modes are actually transmitted from the master to all addressed slaves, the following conditions must be fulfilled:

- each slave is assigned to a slave group, see Fig. 4-31.
- the relevant mode (sync, freeze) is enabled for the corresponding slave group, see Fig. 4-24.

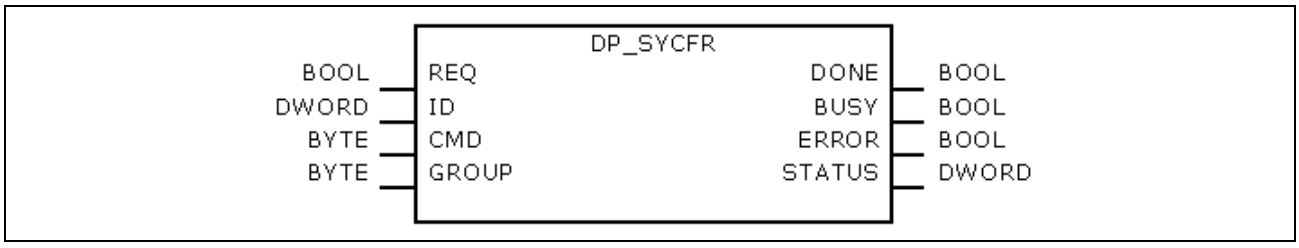


Fig. 6-80: DP_SYCFR

	Name	Type	Comment
VAR_INPUT	REQ	BOOL	TRUE: Execute the function
	ID	DWORD	Slot handle, see the following table
	CMD	BYTE	Control command
	GROUP	BYTE	Selects one or several groups, to which the command refers. Each bit is assigned to a group.
VAR_OUTPUT	DONE	BOOL	TRUE: Call completed
	BUSY	BOOL	TRUE: The function block is busy. As long as BUSY = TRUE, the data cannot yet be evaluated.
	ERROR	BOOL	TRUE: Error/s occurred
	STATUS	DWORD	Lastly defined status

Fig. 6-81: Interface of DP_SYCFR

Bytes	Contents	Description
0	MASTER	ID of the DP system: Has permanently value 0, as the IndraLogic L20 contains exactly one DP master.
1	SEGMENT	Number of the DP segment
2	STATION	Number of the DP slave (bus address): If the command is to be valid for only one special slave, then the bus address of the slave must be entered here (0..125). However, if the command is to be entered for all slaves of a group, a global address (= 127) must be entered here.
3	SLOT	Number of the slot within the slave

Fig. 6-82: Slot handle: parameter "ID"

Example: Slot handle

Address all slaves of a group: ID = 16#007F0000.

The DP_ID function serves for creating the ID from the single components.

Note: To synchronize outputs, it has to be ensured, that all slaves received the current output data before having received the sync or unsync control commands! For this, call the DP_SYCFR function block of the same PLC task, from which the output data of the slave/s are written.

In a PLC task, at first, the output data for the synchronized slaves are written. Then, the sync or unsync command is started with DP_SYCFR. As long as the function block is BUSY, the output must not be modified.

Example: Starting several axes at the same time

1. Send sync control command to the participating Profibus DP devices. This freezes their outputs.
2. Transmit the command to start the axes to the participating devices (e. g. "MoveAbsolut" for the drives).
3. Send unsync control command to the participating Profibus DP devices. Thus, the devices update their outputs corresponding to the internal present output image, start at the same time the axis motion and terminate the sync mode.

Functions

DP_ADDR

This function is not realized. This function can be called up as it is compatible to Profibus Guideline 2182, but it passes a handle unchanged.

Fig. 6-83: DP_ADDR

	Name	Type	Comment
VAR_INPUT	ID	DWORD	Slot handle
Function value		DWORD	

Fig. 6-84: Interface of DP_ADDR

DP_ID

This function considers Profibus Guideline 2182. It provides the handle for a physical address of a slot.

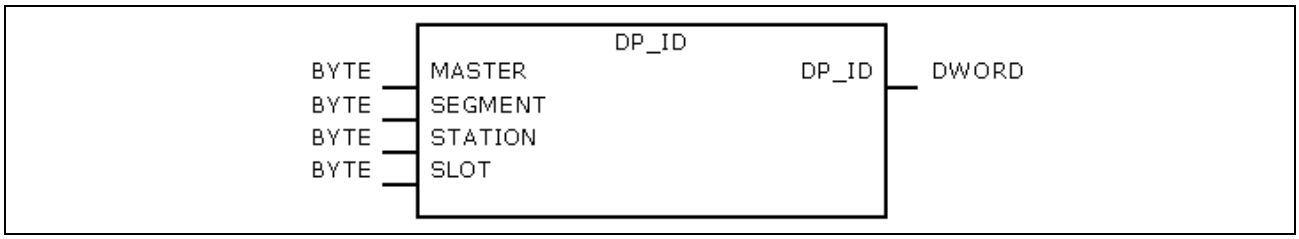


Fig. 6-85: DP_ID

	Name	Type	Comment
VAR_INPUT	MASTER	BYTE	ID of the DP system: Has permanently value 0, as the IndraLogic L20 contains exactly one DP master.
	SEGMENT	BYTE	Number of the DP segment
	STATION	BYTE	Number of the DP slave (bus address): If the command is to be valid for only one special slave, then the bus address of the slave must be entered here (0..125). Only for DP_SYCFR: However, if the command is to be entered for all slaves of a group, a global address (= 127) must be entered here.
	SLOT	BYTE	Number of the slot within the slave
Function value		DWORD	Slot handle

Fig. 6-86: Interface of DP_ID

DP_SLOT

This function considers Profibus Guideline 2182. It sets the slot number defined in the slot handle.

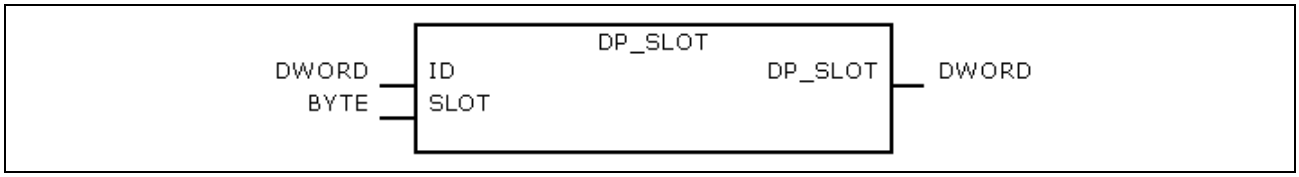


Fig. 6-87: DP_SLOT

	Name	Type	Comment
VAR_INPUT	ID	DWORD	Slot handle
	SLOT	BYTE	Slot number
Function value		DWORD	Slot handle

Fig. 6-88: Interface of DP_SLOT

6.9 RIL_Utilities

Overview

Designation	Type	Description
Version_RIL_Utilities_01V*	FNC	Version management of the RIL_Utilities.lib.
IL_HighResTimeTick	FNC	Reading the high-resolution time tick of the system.
IL_HighResTimeDiff	FNC	Calculation of the time difference of two high-resolution time ticks of the system in micro seconds.
IL_Date	FNC	Reading the current system date.
IL_TimeOfDay	FNC	Reading the current system time.
IL_DateAndTime	FNC	Reading the current system date and the current system time (format according to IEC61131-3).
IL_SysTime64	FB	Reading the current system date and the current system time (in microseconds since 01.01.1970).
IL_SysTimeDate	FB	Reading the current system date and the current system time (in system format).
IL_ExtSysTimeDate	FB	Reading the current system date and the current system time (in extended system format).
IL_SysTime64ToSysTimeDate	FB	Format conversion of the system date and the system time.
IL_SysTimeDateToSysTime64	FB	Format conversion of the system date and the system time.

Fig. 6-89: Overview of the function blocks and functions contained in the RIL_Utilities.lib

Data Types

The RIL_Utilities.lib does not contain any separate data types.

Global Variables

The RIL_Utilities.lib does not contain any separate global variables.

Version_RIL_Utilities_01V*

The Version_RIL_Utilities_01V*[1] function serves for version management of the RIL_Utilities.lib.

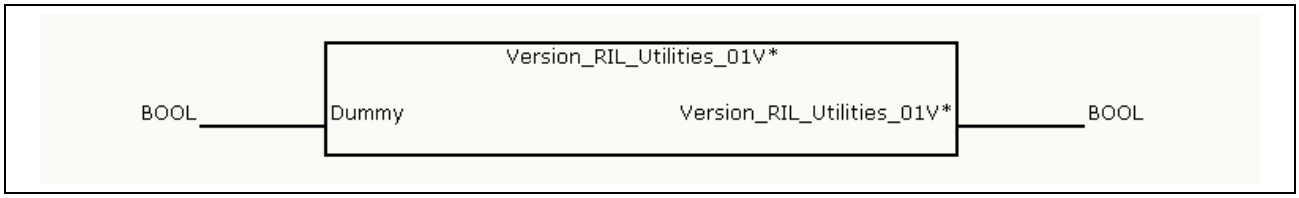


Fig. 6-90: Interface of: Version_RIL_Utilities_01V*

	Name	Type	Comment
VAR_INPUT	Dummy	BOOL	Dummy
Function value	Version_ RIL_Utilities_01V*	BOOL	Acknowledgment that the library is valid

Fig. 6-91: Interface of version_RIL_Utilities_01V*

Specification

The Version_RIL_Utilities_01V* restricts the use of the RIL_Utilities.lib to the valid system as well as to a certain number of valid releases.

The Version_RIL_Utilities_01V* function also shows the user the current release status of the RIL_Utilities.lib and contains an overview of all previously made modifications.

Note: The RIL_Utilities.lib is available on several platforms (systems) in the programming system "IndraLogic; its functional range is adapted to the respective system.

Functional Description

If the RIL_Utilities.lib is integrated in a project, the Version_RIL_Utilities_01V* function restricts the download of the whole project to the valid system as well as to a valid release. If a system or a release is invalid, the corresponding system function is not available and, thus, cannot be addressed, see Fig. 6-92.

Note: The verification of the system and the releases is also active, if the Version_RIL_Utilities_01V* function is not used.

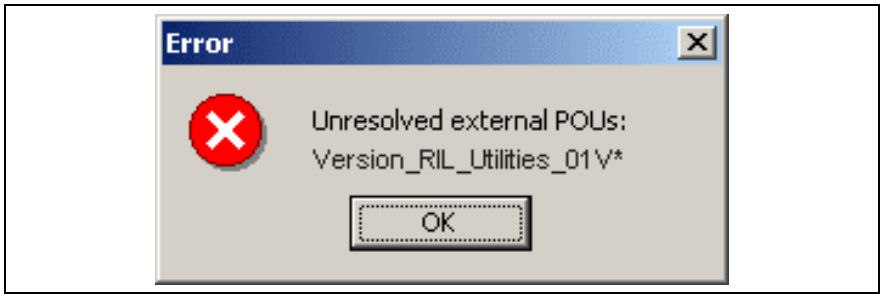


Fig. 6-92: Message: Version check of the system failed

[1] (*) Release

Note: Generally, besides the version function also further functions are indicated as not existing and, thus, cannot be integrated in the project.

If the Version_RIL_Utilities_01V* function is error-free addressed, it signals permanently TRUE as return value.

Note: The Version_RIL_Utilities_01V* function has no functional meaning for parts of the library and must not be used in the project.

IL_HighResTimeTick

The IL_HighResTimeTick function reads the high-resolution time tick of the system.

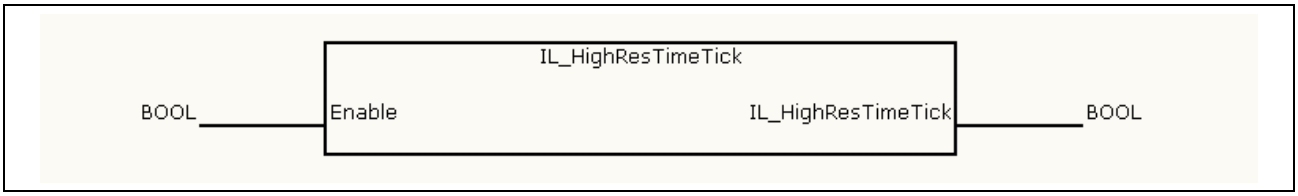


Fig. 6-93: IL_HighResTimeTick

	Name	Type	Comment
VAR_INPUT	Enable	BOOL	Processing enable of the function (cyclic, status-controlled)
Function value	IL_HighResTimeTick	UDINT	High-resolution time tick of the system

Fig. 6-94: Interface of IL_HighResTimeTick

Specification

The IL_HighResTimeTick function in combination with the IL_HighResTimeDiff function is used to determine the runtime of a code segment, see Fig. 6-95.

Note: The return value of the IL_HighResTimeTick function should not be used as time value due to the special system-specific time basis.

Functional Description After the processing enable with "Enable" the IL_HighResTimeTick function retrieves cyclically the high-resolution time tick of the system.

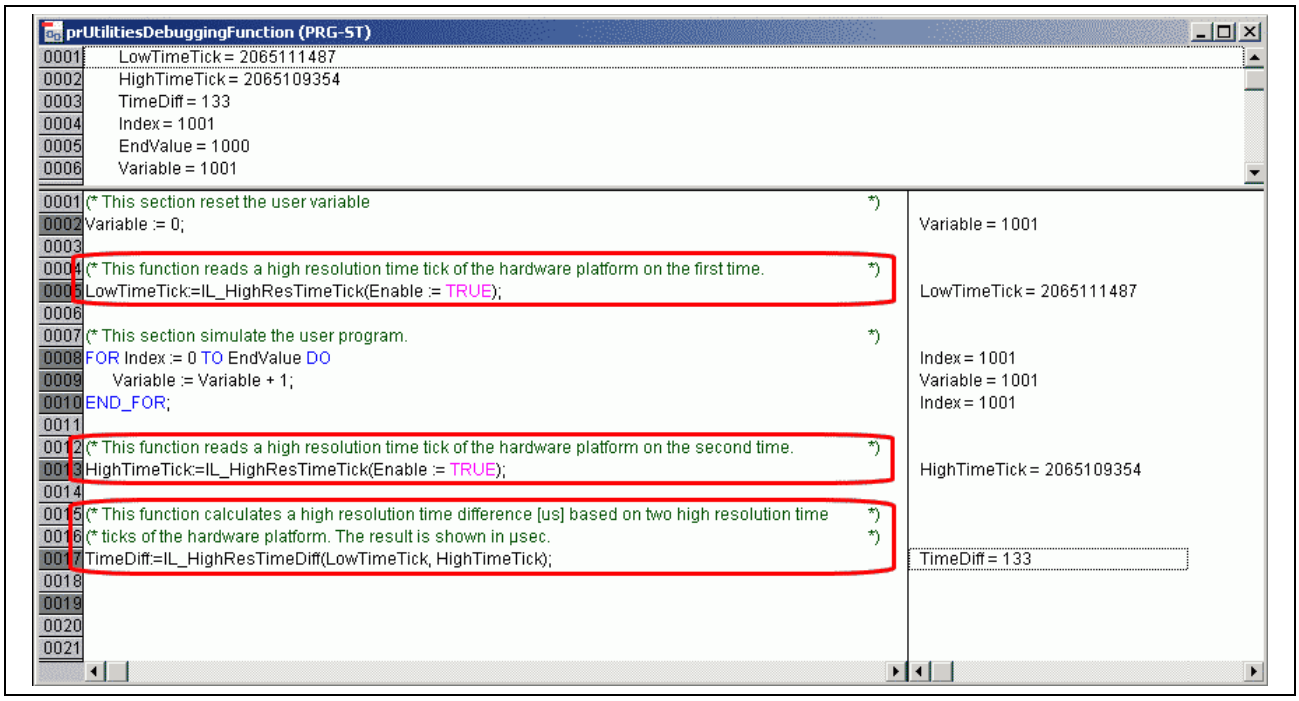


Fig. 6-95: Application example of the IL_HighResTimeTick function

IL_HighResTimeDiff

The IL_HighResTimeDiff function calculates the time difference of two high-resolution time ticks of the system in microseconds.

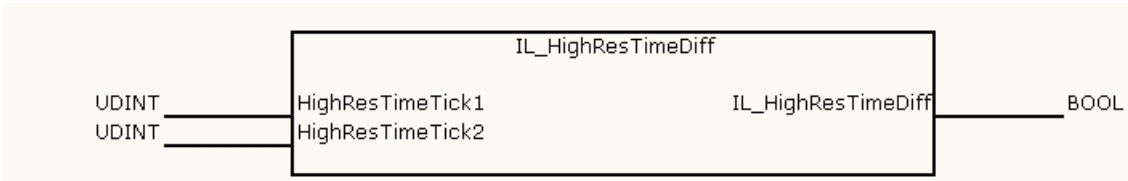


Fig. 6-96: IL_HighResTimeDiff

	Name	Type	Comment
VAR_INPUT	Enable	BOOL	Processing enable of the function (cyclic, status-controlled)
	HighResTimeTick1	UDINT	High-resolution time tick of the system before the code segment
	HighResTimeTick2	UDINT	High-resolution time tick of the system after the code segment
Function value	IL_HighResTimeDiff	UDINT	Time difference of two high-resolution time ticks of the system in [us]

Fig. 6-97: Interface of IL_HighResTimeDiff

Specification

The IL_HighResTimeTick function in combination with the IL_HighResTimeDiff function is used to determine the runtime of a code segment, see Fig. 6-95.

Note: Reading out the high-resolution time tick requires approx. one to two microseconds depending on the system. If necessary, the user can add this time, when he determines the time difference.

Functional Description

After the processing enable with "Enable" the IL_HighResTimeDiff function calculates cyclically the time difference of two high-resolution time ticks of the system in microseconds.

IL_Date

The IL_Date function reads the current system date.

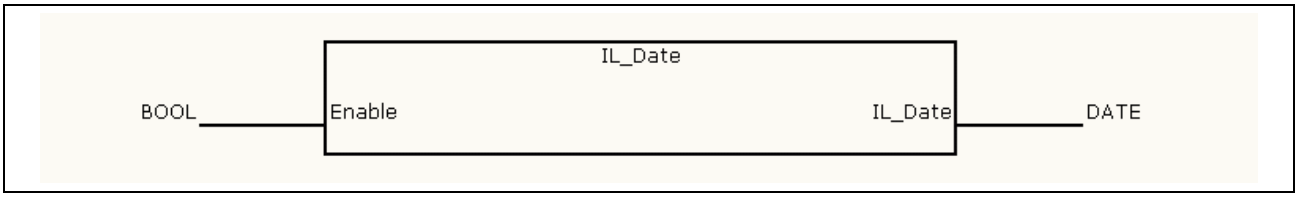


Fig. 6-98: IL_Date

	Name	Type	Comment
VAR_INPUT	Enable	BOOL	Processing enable of the function (cyclic, status-controlled)
RETURN	IL_Date	DATE	Current system date according to IEC61131-3

Fig. 6-99: Interface of IL_Date

Functional Description

After the processing enable with "Enable" the IL_Date function retrieves cyclically the system date formatted according to IEC61131-3.

IL_TimeOfDay

The IL_TimeOfDay function reads the current system time.

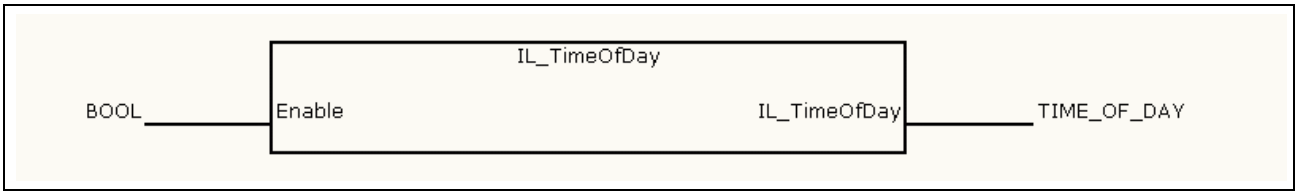


Fig. 6-100: IL_TimeOfDay

	Name	Type	Comment
VAR_INPUT	Enable	BOOL	Processing enable of the function (cyclic, status-controlled)
RETURN	IL_TimeOfDay	TOD	Current system time according to IEC61131-3

Fig. 6-101: Interface of IL_TimeOfDay

Functional Description

After the processing enable with "Enable" the IL_TimeOfDay function retrieves cyclically the system time formatted according to IEC61131-3.

IL_DateAndTime

The IL_DateAndTime function reads the current system date and time.

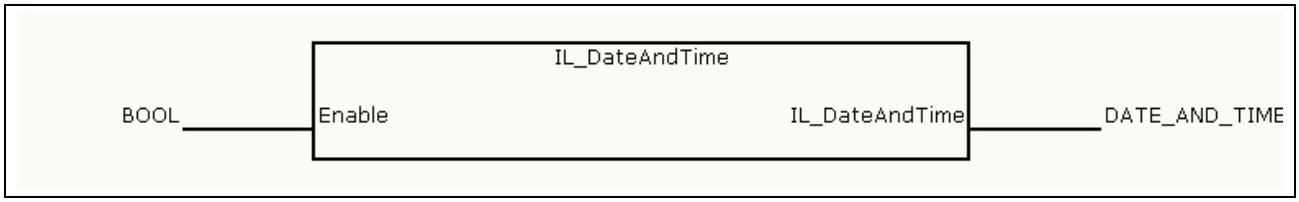


Fig. 6-102: IL_DateAndTime

	Name	Type	Comment
VAR_INPUT	Enable	BOOL	Processing enable of the function (cyclic, status-controlled)
RETURN	IL_DateAndTime	DT	Current system date and time according to IEC61131-3

Fig. 6-103: Interface of IL_DateAndTime

Functional Description

After the processing enable with "Enable" the IL_DateAndTime function retrieves cyclically the current system date as well as the current system time in formatted manner according to IEC61131-3.

IL_SysTime64

The IL_SysTime64 function block reads the current system date and time.

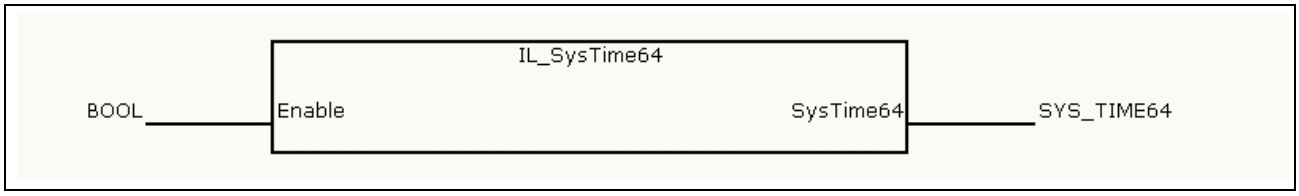


Fig. 6-104: IL_SysTime64

	Name	Type	Comment
VAR_INPUT	Enable	BOOL	Processing enable of the function block (cyclic, status-controlled)
VAR_OUTPUT	SysTime64	SYS_TIME64	Current system date and time in microseconds since 1970-01-01

Fig. 6-105: Interface of IL_SysTime64

Functional Description

After the processing enable with "Enable" the IL_SysTime64 function block retrieves cyclically the current system date and time in microseconds since 1970-01-01.

IL_SysTimeDate

The IL_SysTimeDate function block reads the current system date and time.

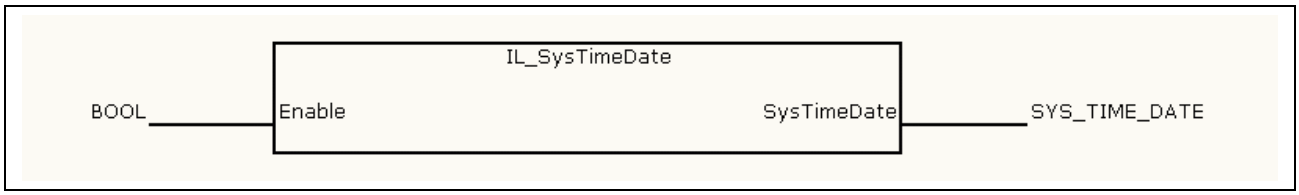


Fig. 6-106: IL_SysTimeDate

	Name	Type	Comment
VAR_INPUT	Enable	BOOL	Processing enable of the function block (cyclic, status-controlled)
VAR_OUTPUT	SysTimeDate	SYS_TIME_DATE	Current system date and time in system format

Fig. 6-107: Interface variables – function block: IL_SysTimeDate

Functional Description

After the processing enable with "Enable" the IL_SysTimeDate function block retrieves cyclically the current system date and time in system format.

IL_ExtSysTimeDate

The IL_ExtSysTimeDate function block reads the current system date and time.

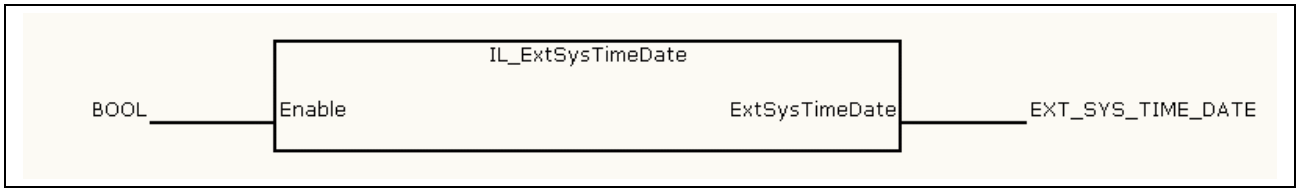


Fig. 6-108: IL_ExtSysTimeDate

	Name	Type	Comment
VAR_INPUT	Enable	BOOL	Processing enable of the function block (cyclic, status-controlled)
VAR_OUTPUT	ExtSysTimeDate	EXT_SYS_TIME_DATE	Current system date and time in extended system format

Fig. 6-109: Interface variables – function block: IL_ExtSysTimeDate

Functional Description

After the processing enable with "Enable" the IL_ExtSysTimeDate function block retrieves cyclically the current system date and time in extended system format.

Note: The extended system format consists of the system formats SYS_TIME64 as well as SYS_TIME_DATE.

IL_SysTime64ToSysTimeDate

The IL_SysTime64ToSysTimeDate function block converts the format of the system date and time.

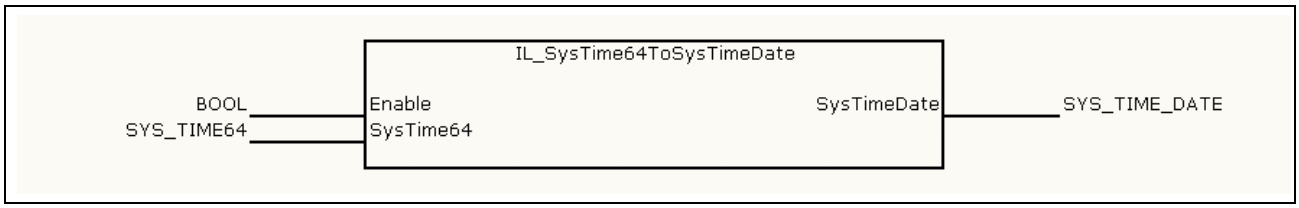


Fig. 6-110: IL_SysTime64ToSysTimeDate

	Name	Type	Comment
VAR_INPUT	Enable	BOOL	Processing enable of the function block (cyclic, status-controlled)
	SysTime64	SYS_TIME64	System date and time in microseconds since 1970-01-01
VAR_OUTPUT	SysTimeDate	SYS_TIME_DATE	System date and time in system format

Fig. 6-111: Interface variables – function block: IL_SysTime64ToSysTimeDate

Functional Description

After the processing enable with "Enable" the IL_SysTime64ToSysTimeDate function block converts cyclically the system date including system time in microseconds applied at input "SysTime64" (basis: 1970-01-01) into the system date and time in system format.

IL_SysTimeDateToSysTime64

Brief Description

The IL_SysTimeDateToSysTime64 function block converts the format of the system date and time.

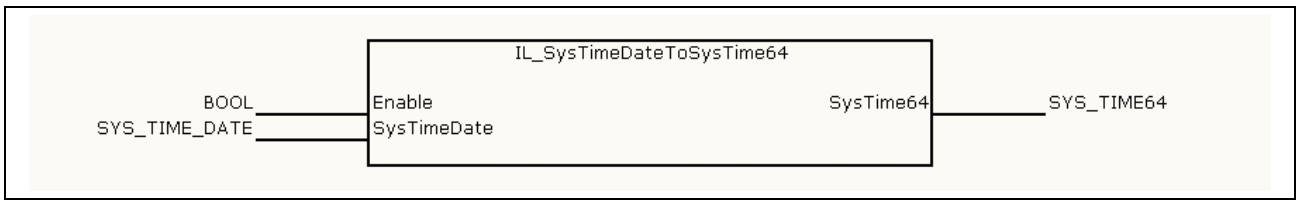


Fig. 6-112: IL_SysTimeDateToSysTime64

	Name	Type	Comment
VAR_INPUT	Enable	BOOL	Processing enable of the function block (cyclic, status-controlled)
	SysTimeDate	SYS_TIME_DATE	System date and time in system format
VAR_OUTPUT	SysTime64	SYS_TIME64	System date and time in microseconds since 1970-01-01

Fig. 6-113: Interface variables – function block: IL_SysTimeDateToSysTime64

Functional Description

After the processing enable with "Enable" the IL_SysTimeDateToSysTime64 function block converts cyclically the system date including system time in microseconds applied at input "SysTimeDate" (basis: 1970-01-01).

6.10 RIL_VExUtil

IL_VExKeys

Brief Description

With the IL_VExKeys function block the user can transmit the keystrokes on an HMI device safely to the control. The connection occurs via UDP. Each time when the function block is called, the last data is copied. If the connection is interrupted, all outputs are reset to 0. Thus, the user is able to program safe properties, whereby in the output data of the function block the safety mode is assigned to 0.

Interface Description

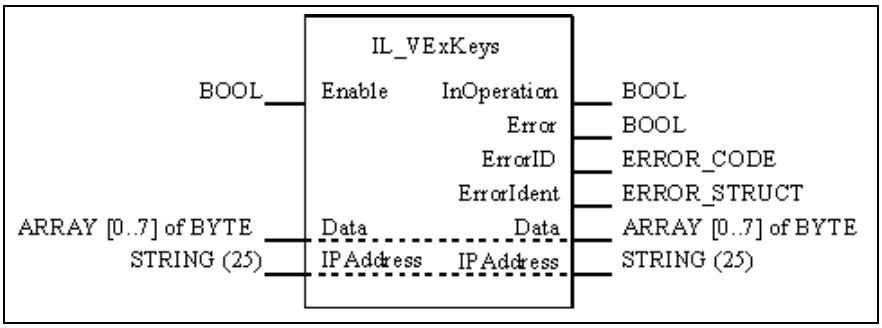


Fig. 6-114: IL_VexKeys function block

	Name	Type	Comment
VAR_IN_OUT	Data	ARRAY [0..7] OF BYTE	Contains data as byte array. The meaning of the particular bit is described in the documentation of the HMI.
	IPAddress	STRING(25)	Entered IP address "."§ notation, e. g. "10.104.73.193"
VAR_INPUT	Enable	BOOL	As long as the input is TRUE, the data is permanently updated. If it is inactive, the output data are all 0.
VAR_OUTPUT	InOperation	BOOL	If TRUE, the output data are valid and are updated.
	Error	BOOL	Indicates that an error has occurred within the function block. The output data are all set to 0.
	ErrorID	ERROR_CODE	Error recognition (see next table "Error codes").
	ErrorIdent	ERROR_STRUCT	Error structure with further division of the errors.

Fig. 6-115: IL_VexKeys function block

Error Codes

The function block generates in Additional1/Additional2 for table "**MLC_ETHERNET**" the following error messages:

ErrorID	Additional1	Additional2	Description
INPUT_INVALID_ERROR (16#0001)	10	0	Wrong IP syntax
COMMUNICATION_ERROR (16#0002)	11	0	Unknown IP address or no connection
DEVICE_ERROR (16#0008)	12	0	Invalid internal buffer size
STATE_MACHINE_ERROR (16#0005)	1	0	Invalid status of the state machine

Fig. 6-116: Generated error numbers of the IL_VexKeys function block

Notes

7 Display and Operating Components

On its front, the IndraLogic L20 is provided with the following display and operating components: a single-line display, operating keys, reset button and the Stop LED.

7.1 Display and Operating Keys

Display LCD with 8 characters (5 x 10 dot matrix).

Fig. 7-1: Display with operating keys

Operating Keys

- **<Esc>**:

Moves back to the last lower level. Thereby, changes in the exited menu are rejected.

- **<Down>** (arrow down):

Navigates downwards within a menu

- or -

Decrements a selected parameter value

- **<Up>** (arrow up):

Navigates upwards within a menu

- or -

Increments a selected parameter value

- **<Enter>**:

Confirms an input/change

- or -

Calls the next (lower) menu level

7.2 Reset Button S1 and Stop LED

Fig. 7-2: Stop LED and countersunk reset button S1

Reset Button S1

Resets the IndraLogic L20 and forces a restart. The reset button can only be actuated with an additional tool (e. g. tip of a pencil).

Note: A reset interrupts a running program processing!

Stop LED

The Stop LED indicates basic PLC states.

LED OFF	Normal status (Run, PLC program is running)
LED red	PLC stopped
LED red flashing	Outputs are disabled

Fig. 7-3: Meaning of the stop LED displays

7.3 Available Menu Levels

Via the display and the operating keys you can retrieve information at the device and make settings. To structure the data they are subdivided into function-related menu levels.

Default and Status Displays

After switching on the IndraLogic L20 the default setting appears on the display.

Use the <Up>, <Down>, <Enter> or <Esc> keys to navigate between the single menu items and levels.

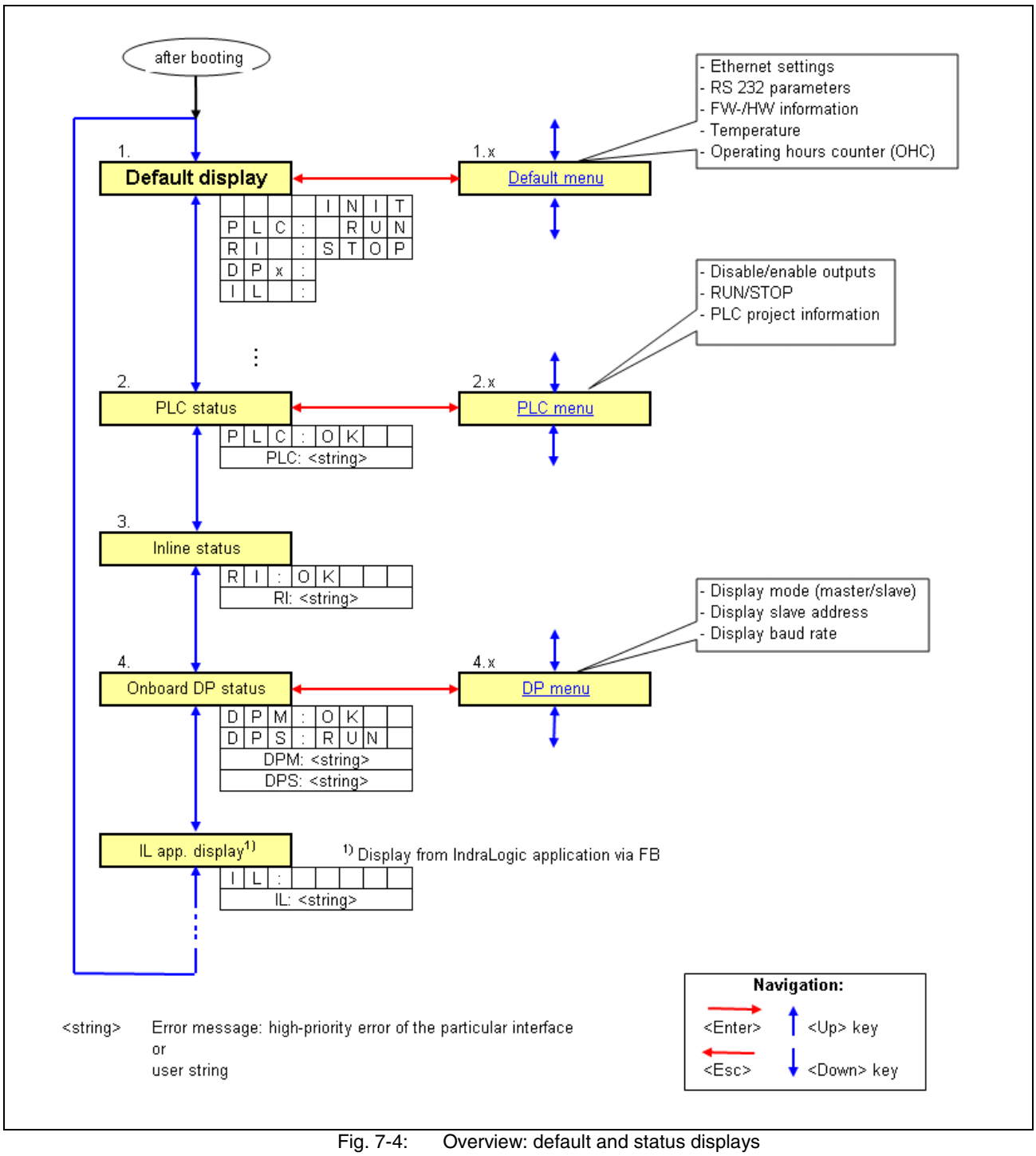


Fig. 7-4: Overview: default and status displays

Default Display

"INIT": PLC boots.

"PLC: RUN": Normal mode. PLC is running.

"RI" or "DP" flashes: An Inline (RI) or Profibus DP error (DPM: Master; DPS: Slave) is active. Details on the Inline error can be found in the "Inline Status" menu or on the Profibus DP error in the "Onboard DP Status" menu. If several errors are applied, only the error with the highest priority is displayed.

"IL:" Flashes: A user output is pending. Its display is possible via menu "IL app. display".

Use the <Enter> key to switch into the "Default menu".

PLC Status

Indicates the current PLC status.

Use the <Enter> key to switch into the "PLC menu".

Inline Status

Indicates an active Rexroth Inline error.

Example: "RI: Rexroth Inline configuration error at module: xxx"

Onboard DP Status

DPM:...": (only when the IndraLogic L20 is used as Profibus master with the Onboard DP interface X7P) indicates the Profibus status of the master.

DPS:...": (only when the IndraLogic L20 is used as Profibus slave with the Onboard DP interface X7P) indicates the Profibus status of the slave.

Use the <Enter> key to switch into the "Profibus DP menu".

IL Display

Indicates a pending IndraLogic user output, see also Fig. 6-13.

Default Menu

After calling the default menu with the <Enter> key in the default display the "Ethernet" menu item is indicated in the default display.

Use the <Up>, <Down>, <Enter> or <Esc> keys to navigate between the single menu items and levels.

.

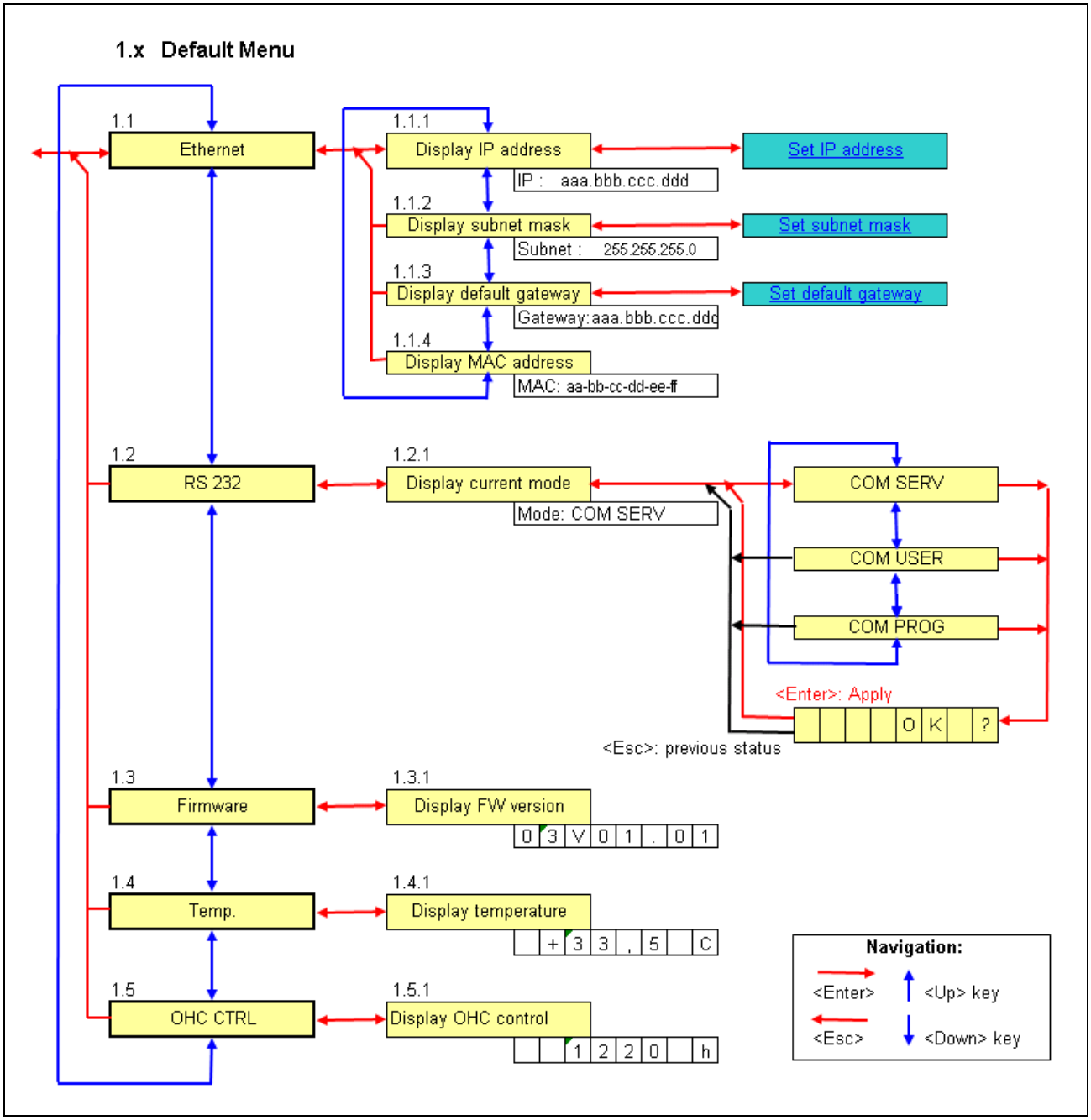


Fig. 7-5: Overview: default menu

Ethernet

Serves for indicating/setting the IP address, subnet mask and the default gateway. Moreover, you will find the MAC address here.

For the setting please consider Fig. 7-6.

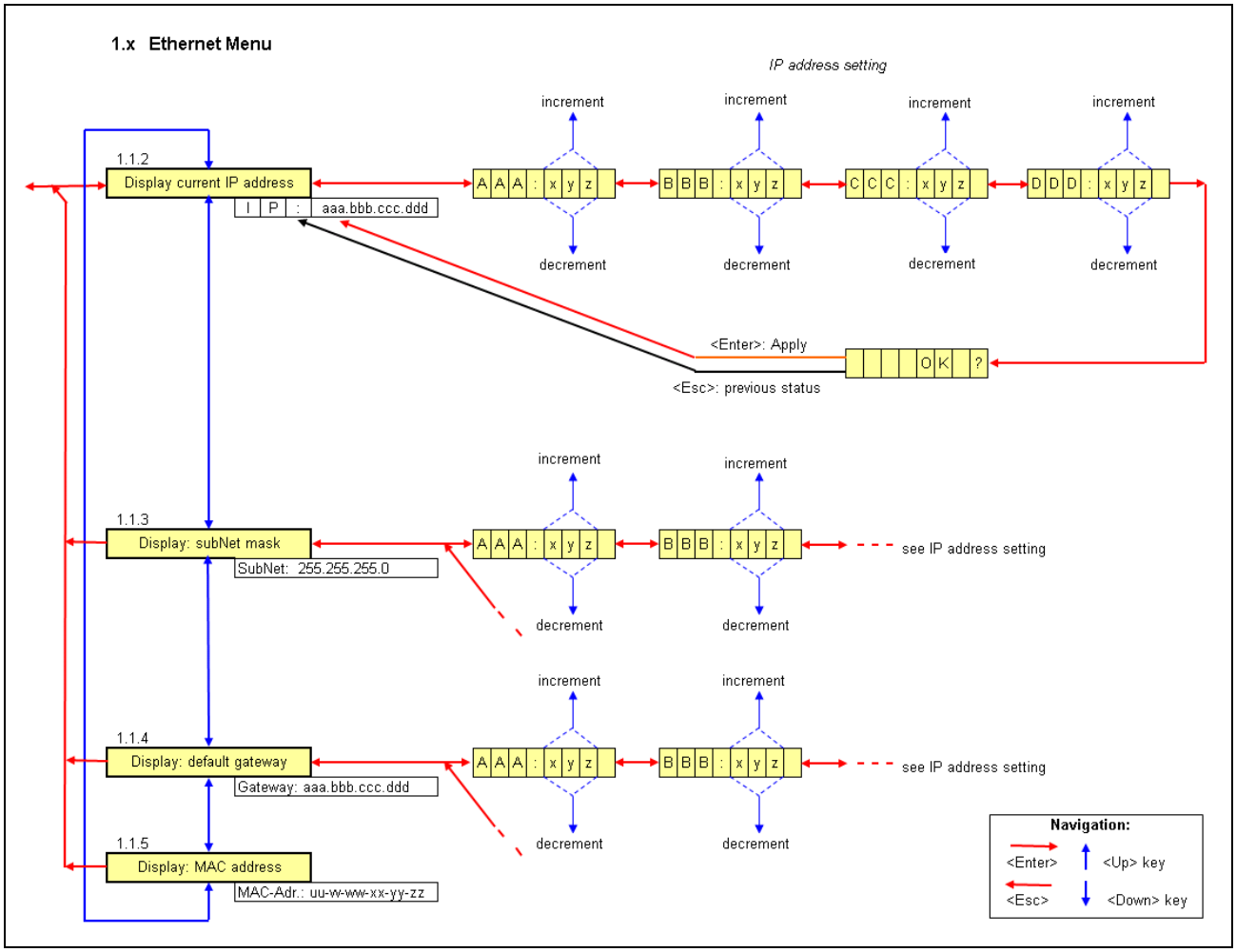


Fig. 7-6: Setting the IP address, subnet mask and default gateway

1. To change an address activate the <Enter> key while the current address is displayed.

   In the display the letter sequence "AAA:" appears to identify the first address byte followed by its current decimal value.
2. Set the desired value with the <Up> and <Down> key.
3. Acknowledge the setting with <Enter>.

   In the display the letter sequence "BBB:" appears to identify the second address byte followed by its current decimal value.
4. Set the remaining address bytes (BBB, CCC, DDD) as described above.

   After acknowledging the last setting the system requests by displaying "OK ?", if the new address value is to be applied.
5. Confirm with <Enter>, then the new address value is indicated, written in the die compact flash card and used **when starting** the control next time. If this is not desired, you can reject the change with <Esc>. Then, the previous address value remains active.

RS 232

Determines how the onboard RS232 interface (X3C) is used:

COM SERV: Communication with a small operator terminal. The interface is configured by the small operator terminal.

COM USER: Communication with a programming device. The interface configuration occurs either with the library function by the PLC program or keeps the default setting (38 400 bauds, no Parity, 8 data bits, 1 stop bit).

COM PROG: For communication with a programming device.

Note: After applying a modified setting the IndraLogic L20 has to be switched off and on again. Only then, the settings are applied.

Firmware

Indicates the version number of the installed firmware.

Temp.

Indicates the internal temperature of the IndraLogic L20.

Note: If the internal temperature reaches 80 °C, the control switches automatically in the stop mode, switches the outputs in the safe status and indicates the warning "Temp !!!" on the display. This mode can only be quit by switching the power supply off and on again.

Before you switch on the control again after an overheating you should find the cause of the problem. Ensure that the surrounding air temperature of the IndraLogic L20 is not higher than 45 °C.

Note: The IH_Temperature library function (see page 6-9) allows the program-controlled reading of the internal temperature of the IndraLogic L20. Thus, critical temperature rises can be detected at an early stage and measures can be taken to avoid an over-temperature.

OHC CTRL

Indicates the value of the IndraLogic L20's operating hours counter .

PLC Menu

After calling the PLC menu with the <Enter> key in menu "PLC status" the "OUTP ON" or "OUTP OFF" menu item is indicated.

Use the <Up>, <Down>, <Enter> or <Esc> keys to navigate between the single menu items and levels.

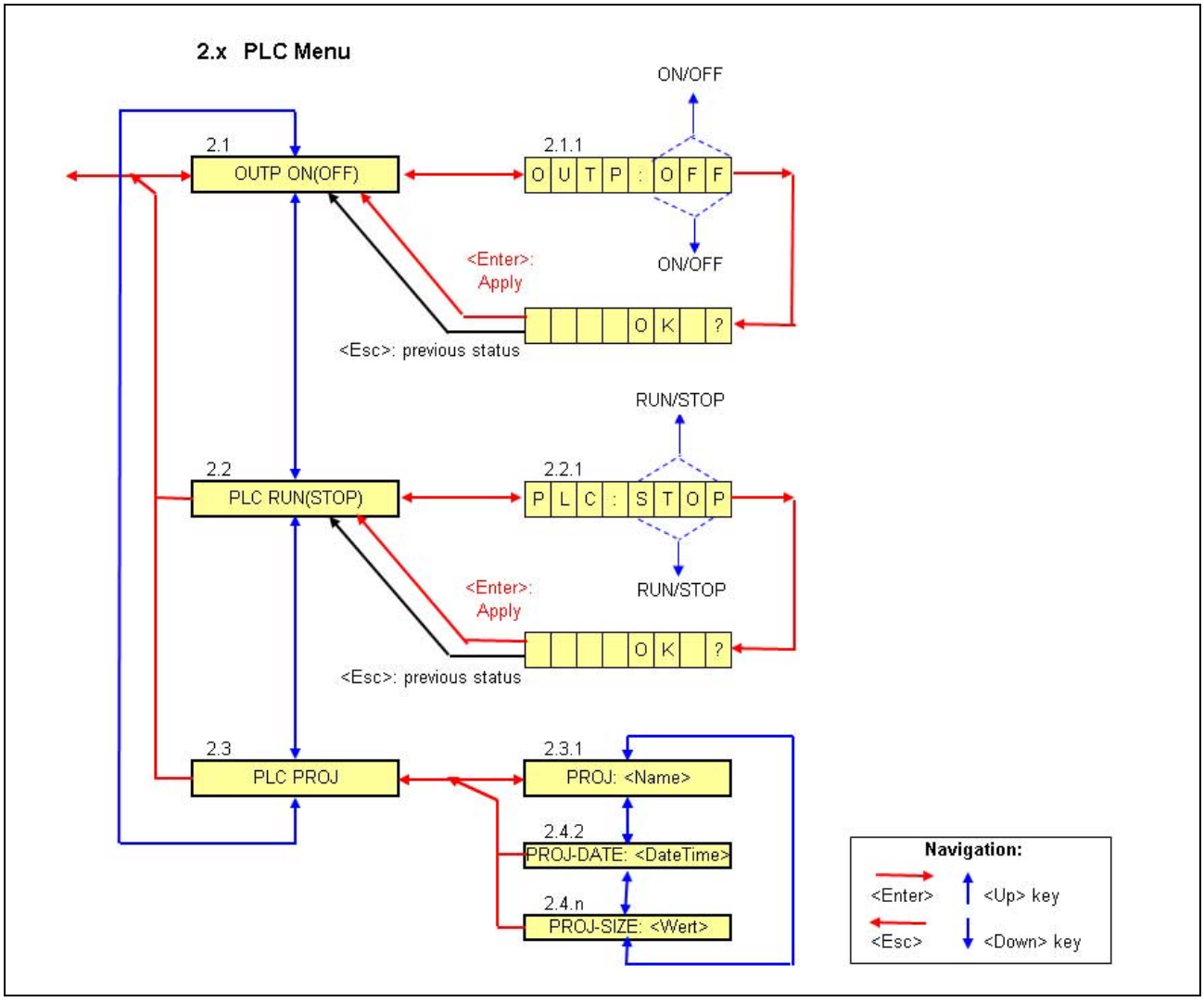


Fig. 7-7: Overview: PLC menu

OUTP ON / OUTP OFF

Affects all local digital outputs of the IndraLogic L20, all outputs of the connected Rexroth Inline modules as well as all outputs triggered via Profibus DP:

OUTP ON: Enables outputs

OUTP OFF: Resets outputs (safe status). The Stop LED flashes.

PLC RUN / PLC STOP

- Starts/stops the PLC program sequence and
- affects all local digital outputs of the IndraLogic L20, all outputs of the connected Rexroth Inline modules as well as all outputs triggered via Profibus DP.

PLC RUN: Enables outputs and starts the PLC program.

PLC STOP: Sets the outputs into the safe status and stops the PLC program.

Note: The same functions can be activated by "Start" and "Stop" at a connected programming device. A e. g. PLC STOP activated at the IndraLogic L20 can therefore also be canceled via the programming device!

PLC PROJ

Indicates name, creation date and size of the currently loaded PLC project.

Profibus DP Menu

After calling the Profibus DP menu with the <Enter> key in the "Onboard-DP status" menu detailed information on some Profibus DP settings can be indicated.

Use the <Up>, <Down>, <Enter> or <Esc> keys to navigate between the single menu items and levels.

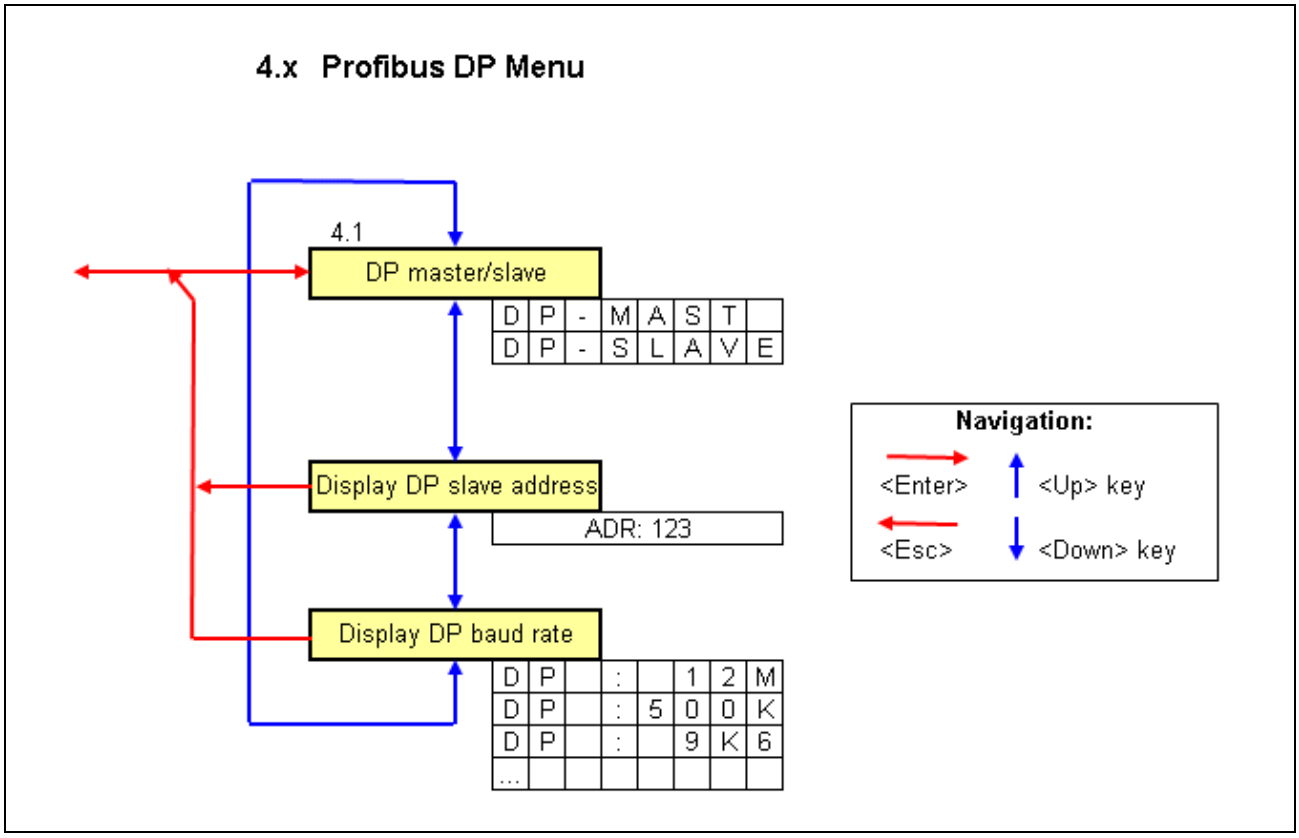


Fig. 7-8: Overview: Profibus DP menu

DP Master/Slave

Indicates, if the onboard DP interface of the IndraLogic L20 is parameterized as master ("DP-MAST") or as Slave (DP-SLAVE).

Indicating DP Slave Address

Indicates the active slave address of the IndraLogic L20. The display is only available, if the IndraLogic L20 has been parameterized as slave.

Indicating DP Baud Rate

Indicates the active baud rate of the onboard DP interface (X7P).

Examples: "12M" = 12 Mbauds; "500K" = 500 kbauds; "9K6" = 9600 bauds

Boot Menu: Deleting the Boot Project

If you press the operating key combination <Esc>+<Enter> during the control startup, the IndraLogic L20 branches into the boot menu.

In the boot menu you have access to the following function:

Bootproj

Deletes the boot project stored on the compact flash card (corresponds to the IndraLogic function "Reset (origin)").

Note: The function to delete the boot project should only be used in exceptional cases (e. g. if the control does not start because of a faulty boot project)!

Press <Esc> to exit the boot menu without deleting the boot project

- or -

press two times <Enter> to delete the boot project and then <Esc> to exit the boot menu.

After that, the default setting appears in the display of the IndraLogic L20.

Notes

8 Technical Data

8.1 Equipment

Processor	STMicroelectronics ST40 with 192 MHz
Working memory	At least 16 Mbytes DRAM and at least 64 kbytes NvRAM
Interfaces: **Interface to I/O modules** **Communication interfaces**	 • Rexroth Inline interface • 1 x Ethernet connection (RJ 45, 10/100 Base-T) • 1 x serial RS 232 interface • 1 x Profibus DP interface (master-/slave-compliant)
Inputs and outputs	• 8 electrically isolated digital inputs • 8 electrically isolated digital outputs

Fig. 8-1: Equipment

8.2 Power Data

User memory for program code	1 Mbyte
User memory for data	2 Mbytes
Remanent memory	32 kbytes
Task number	8
Profibus DP	According to IEC 61158-3

Fig. 8-2: Power data

Notes

9 List of Figures

10 Index

E

F

G

I

K

L

M

O

P

R

S

T

U

V

W

11 Service & Support

11.1 Helpdesk

Unser Kundendienst-Helpdesk im Hauptwerk Lohr am Main steht Ihnen mit Rat und Tat zur Seite. Sie erreichen uns

Our service helpdesk at our headquarters in Lohr am Main, Germany can assist you in all kinds of inquiries. Contact us

- telefonisch - by phone: **+49 (0) 9352 40 50 60**
  über Service Call Entry Center Mo-Fr 07:00-18:00 Central European Time
  - via Service Call Entry Center Mo-Fr 7:00 am - 6:00 pm CET
- per Fax - by fax: **+49 (0) 9352 40 49 41**
- per e-Mail - by e-mail: service.svc@boschrexroth.de

11.2 Service-Hotline

Außerhalb der Helpdesk-Zeiten ist der Service Deutschland direkt ansprechbar unter

After helpdesk hours, contact the German service experts directly at

+49 (0) 171 333 88 26

oder - or **+49 (0) 172 660 04 06**

Hotline-Rufnummern anderer Länder entnehmen Sie bitte den Adressen in den jeweiligen Regionen.

Hotline numbers of other countries to be seen in the addresses of each region.

11.3 Internet

Unter **www.boschrexroth.com** finden Sie ergänzende Hinweise zu Service, Reparatur und Training sowie die **aktuellen** Adressen *) unserer auf den folgenden Seiten aufgeführten Vertriebs- und Servicebüros.

Verkaufsniederlassungen
Niederlassungen mit Kundendienst

Außerhalb Deutschlands nehmen Sie bitte zuerst Kontakt mit unserem für Sie nächstgelegenen Ansprechpartner auf.

*) Die Angaben in der vorliegenden Dokumentation können seit Drucklegung überholt sein.

At **www.boschrexroth.com** you can find additional notes about service, repairs and training. The current addresses *) for our sales and service facilities locations around the world are on the following pages.

sales agencies
sales agencies providing service

Please contact our sales / service office in your area first.

*) Data in the present documentation may have become obsolete since printing.

11.4 Vor der Kontaktaufnahme... - Before contacting us...

Wir können Ihnen schnell und effizient helfen wenn Sie folgende Informationen bereithalten:

1. detaillierte Beschreibung der Störung und der Umstände.
2. Angaben auf dem Typenschild der betreffenden Produkte, insbesondere Typenschlüssel und Seriennummern.
3. Tel.-/Faxnummern und e-Mail-Adresse, unter denen Sie für Rückfragen zu erreichen sind.

For quick and efficient help, please have the following information ready:

1. Detailed description of the failure and circumstances.
2. Information on the type plate of the affected products, especially type codes and serial numbers.
3. Your phone/fax numbers and e-mail address, so we can contact you in case of questions.

11.5 Kundenbetreuungsstellen - Sales & Service Facilities

Deutschland – Germany

vom Ausland: (0) nach Landeskennziffer weglassen!
from abroad: don't dial (0) after country code!

Vertriebsgebiet Mitte Germany Centre Bosch Rexroth Electrice Drives and Controls GmbH Bgm.-Dr.-Nebel-Str. 2 / Postf. 1357 97816 Lohr am Main / 97803 Lohr **Kompetenz-Zentrum Europa** Tel.: +49 (0)9352 40-0 Fax: +49 (0)9352 40-4885	**SERVICE AUTOMATION** **CALL ENTRY CENTER** **Helpdesk** **MO – FR** **von 07:00 - 18:00 Uhr** **from 7 am – 6 pm** **Tel. +49 (0) 9352 40 50 60** **Fax +49 (0) 9352 40 49 41** service.svc@boschrexroth.de	**SERVICE AUTOMATION** **HOTLINE 24 / 7 / 365** **außerhalb der Helpdesk-Zeit** **out of helpdesk hours** **Tel.: +49 (0)172 660 04 06** oder / or **Tel.: +49 (0)171 333 88 26**	**SERVICE AUTOMATION** **ERSATZTEILE / SPARES** verlängerte Ansprechzeit - extended office time - ♦ nur an Werktagen - only on working days - ♦ von 07:00 - 18:00 Uhr - from 7 am - 6 pm - **Tel. +49 (0) 9352 40 42 22**
Vertriebsgebiet Süd Germany South Bosch Rexroth AG Landshuter Allee 8-10 80637 München Tel.: +49 (0)89 127 14-0 Fax: +49 (0)89 127 14-490	Vertriebsgebiet West Germany West Bosch Rexroth AG Regionalzentrum West Borsigstrasse 15 40880 Ratingen Tel.: +49 (0)2102 409-0 Fax: +49 (0)2102 409-406 +49 (0)2102 409-430	Gebiet Südwest Germany South-West Bosch Rexroth AG Service-Regionalzentrum Süd-West Siemensstr. 1 70736 Fellbach Tel.: +49 (0)711 51046–0 Fax: +49 (0)711 51046–248	
Vertriebsgebiet Nord Germany North Bosch Rexroth AG Walsroder Str. 93 30853 Langenhagen Tel.: +49 (0) 511 72 66 57-0 Service: +49 (0) 511 72 66 57-256 Fax: +49 (0) 511 72 66 57-93 Service: +49 (0) 511 72 66 57-783	Vertriebsgebiet Mitte Germany Centre Bosch Rexroth AG Regionalzentrum Mitte Waldecker Straße 13 64546 Mörfelden-Walldorf Tel.: +49 (0) 61 05 702-3 Fax: +49 (0) 61 05 702-444	Vertriebsgebiet Ost Germany East Bosch Rexroth AG Beckerstraße 31 09120 Chemnitz Tel.: +49 (0)371 35 55-0 Fax: +49 (0)371 35 55-333	Vertriebsgebiet Ost Germany East Bosch Rexroth AG Regionalzentrum Ost Walter-Köhn-Str. 4d 04356 Leipzig Tel.: +49 (0)341 25 61-0 Fax: +49 (0)341 25 61-111

Europa (West) - Europe (West)

vom Ausland: (0) nach Landeskennziffer weglassen, Italien: 0 nach Landeskennziffer mitwählen
from abroad: don't dial (0) after country code, Italy: dial 0 after country code

Austria - Österreich

Bosch Rexroth GmbH
Electric Drives & Controls
Stachegasse 13
1120 Wien
Tel.: +43 (0) 1 985 25 40
Fax: +43 (0) 1 985 25 40-1459

Austria – Österreich

Bosch Rexroth GmbH
Electric Drives & Controls
Industriepark 18
4061 Pasching
Tel.: +43 (0)7221 605-0
Fax: +43 (0)7221 605-1220

Belgium - Belgien

Bosch Rexroth NV/SA
Henri Genessestraat 1
1070 Bruxelles

Tel: +32 (0) 2 451 26 08
Fax: +32 (0) 2 451 27 90
info@boschrexroth.be
service@boschrexroth.be

Denmark - Dänemark

BEC A/S
Zinkvej 6
8900 Randers

Tel.: +45 87 11 90 60
Fax: +45 87 11 90 61

Denmark - Dänemark

Bosch Rexroth A/S
Engelsholmvej 26
8900 Randers

Tel.: +45 36 77 44 66
Fax: +45 70 10 03 20
thomas.jansen@boschrexroth.dk

Great Britain – Großbritannien

Bosch Rexroth Ltd.
Electric Drives & Controls
Broadway Lane, South Cerney
Cirencester, Glos GL7 5UH

Tel.: +44 (0)1285 863-000
Fax: +44 (0)1285 863-030
sales@boschrexroth.co.uk
service@boschrexroth.co.uk

Finland - Finnland

Bosch Rexroth Oy
Electric Drives & Controls
Ansatie 6
01740 Vantaa

Tel.: +358 10 3441 000
Fax: +358 10 3441 506

France - Frankreich

Bosch Rexroth SAS
Electric Drives & Controls
Avenue de la Trentaine
(BP. 74)
77503 Chelles Cedex
Tel.: +33 (0)164 72-63 22
Fax: +33 (0)164 72-63 20
Hotline: +33 (0)608 33 43 28

France - Frankreich

Bosch Rexroth SAS
Electric Drives & Controls
ZI de Thibaud, 20 bd. Thibaud
(BP. 1751)
31084 Toulouse
Tel.: +33 (0)5 61 43 61 87
Fax: +33 (0)5 61 43 94 12

France – Frankreich

Bosch Rexroth SAS
Electric Drives & Controls
91, Bd. Irène Joliot-Curie
69634 Vénissieux – Cedex
Tel.: +33 (0)4 78 78 53 65
Fax: +33 (0)4 78 78 53 62

France – Frankreich

Tightening & Press-fit:
Globe Techniques Nouvelles
143, Av. du Général de Gaulle
92252 La Garenne Colombes

Tel.: +33 (0)1 41 19 33 33

Italy - Italien

Bosch Rexroth S.p.A.
Strada Statale Padana
Superiore 11, no. 41
20063 Cernusco S/N.MI
Tel.: +39 02 92 365 1
Service: +39 02 92 365 300
Fax: +39 02 92 365 500
Service: +39 02 92 365 516

Italy - Italien

Bosch Rexroth S.p.A.
Via Paolo Veronesi, 250
10148 Torino

Tel.: +39 011 224 88 11
Fax: +39 011 224 88 30

Italy - Italien

Bosch Rexroth S.p.A.
Via Mascia, 1
80053 Castellamare di Stabia NA

Tel.: +39 081 8 71 57 00
Fax: +39 081 8 71 68 85

Italy - Italien

Bosch Rexroth S.p.A.
Via del Progresso, 16 (Zona Ind.)
35020 Padova

Tel.: +39 049 8 70 13 70
Fax: +39 049 8 70 13 77

Italy - Italien

Bosch Rexroth S.p.A.
Via Isonzo, 61
40033 Casalecchio di Reno (Bo)

Tel.: +39 051 29 86 430
Fax: +39 051 29 86 490

Italy - Italien

Tightening & Press-fit:
TEMA S.p.A. Automazione
Via Juker, 28
20025 Legnano

Tel.: +39 0 331 4671

Netherlands – Niederlande/Holland

Bosch Rexroth B.V.
Kruisbroeksestraat 1
(P.O. Box 32)
5281 RV Boxtel

Tel.: +31 (0) 411 65 16 40
Fax: +31 (0) 411 65 14 83
www.boschrexroth.nl

Netherlands - Niederlande/Holland

Bosch Rexroth Services B.V.
Technical Services
Kruisbroeksestraat 1
(P.O. Box 32)
5281 RV Boxtel
Tel.: +31 (0) 411 65 19 51
Fax: +31 (0) 411 67 78 14
Hotline: +31 (0) 411 65 19 51
services@boschrexroth.nl

Norway - Norwegen

Bosch Rexroth AS
Electric Drives & Controls
Berghagan 1 or: Box 3007
1405 Ski-Langhus 1402 Ski

Tel.: +47 64 86 41 00
Fax: +47 64 86 90 62
Hotline: +47 64 86 94 82
arnt.kristian.barsten @boschrexroth.no

Spain – Spanien

Goimendi Automation S.L.
Parque Empresarial Zuatzu
C/ Francisco Grandmontagne no.2
20018 San Sebastian

Tel.: +34 9 43 31 84 21
- service: +34 9 43 31 84 56
Fax: +34 9 43 31 84 27
- service: +34 9 43 31 84 60
sat.indramat@goimendi.es

Spain - Spanien

Bosch Rexroth S.A.
Electric Drives & Controls
Centro Industrial Santiga
Obradors 14-16
08130 Santa Perpetua de Mogoda
Barcelona
Tel.: +34 9 37 47 94-00
Fax: +34 9 37 47 94-01

Spain - Spanien

Bosch Rexroth S.A.
Electric Drives & Controls
c/ Almazara, 9
28760 Tres Cantos (Madrid)

Tel.: +34 91 806 24 79
Fax: +34 91 806 24 72
fernando.bariego@boschrexroth.es

Sweden - Schweden

Bosch Rexroth AB
Electric Drives & Controls
- Varuvägen 7
(Service: Konsumentvägen 4, Älfsjö)
125 81 Stockholm

Tel.: +46 (0) 8 727 92 00
Fax: +46 (0) 8 647 32 77

Sweden - Schweden

Bosch Rexroth AB
Electric Drives & Controls
Ekvändan 7
254 67 Helsingborg
Tel.: +46 (0) 4 238 88 -50
Fax: +46 (0) 4 238 88 -74

Switzerland East - Schweiz Ost

Bosch Rexroth Schweiz AG
Electric Drives & Controls
Hemrietstrasse 2
8863 Buttikon
Tel. +41 (0) 55 46 46 111
Fax +41 (0) 55 46 46 222

Switzerland West - Schweiz West

Bosch Rexroth Suisse SA
Av. Général Guisan 26
1800 Vevey 1

Tel.: +41 (0)21 632 84 20
Fax: +41 (0)21 632 84 21

Europa (Ost) - Europe (East)

vom Ausland: (0) nach Landeskennziffer weglassen
from abroad: don't dial (0) after country code

Czech Republic - Tschechien	Czech Republic - Tschechien	Hungary - Ungarn	Poland – Polen
Bosch -Rexroth, spol.s.r.o. Hviezdoslavova 5 627 00 Brno Tel.: +420 (0)5 48 126 358 Fax: +420 (0)5 48 126 112	**Tightening & Press-fit:** Bosch -Rexroth, spol.s.r.o. Stetkova 18 140 68 Praha 4 Tel.: +420 (0)241 406 675	Bosch Rexroth Kft. Angol utca 34 1149 Budapest Tel.: +36 (1) 422 3200 Fax: +36 (1) 422 3201	Bosch Rexroth Sp.zo.o. ul. Jutrzenki 102/104 02-230 Warszawa Tel.: +48 (0) 22 738 18 46 Fax: +48 (0) 22 738 18 64
Poland – Polen	**Romania - Rumänien**	**Romania - Rumänien**	**Russia - Russland**
Bosch Rexroth Sp.zo.o. Biuro Poznan ul. Dabrowskiego 81/85 60-529 Poznan Tel.: +48 061 847 64 62 /-63 Fax: +48 061 847 64 02	East Electric S.R.L. Bdul Basarabia no.250, sector 3 73429 Bucuresti Tel./Fax:: +40 (0)21 255 35 07 +40 (0)21 255 77 13 Fax: +40 (0)21 725 61 21 eastel@rdsnet.ro	Bosch Rexroth Sp.zo.o. Str. Drobety nr. 4-10, app. 14 70258 Bucuresti, Sector 2 Tel.: +40 (0)1 210 48 25 +40 (0)1 210 29 50 Fax: +40 (0)1 210 29 52	Bosch Rexroth OOO Tschschjolkowskoje Chaussee 100, Etage 11 105523 Moskau Tel.: +7-495-783 30 60 Fax: +7-495 783 30 69 brcschrexroth.ru
Turkey - Türkei	**Turkey - Türkei**	**Slowakia - Slowakei**	**Slowenia - Slowenien**
Bosch Rexroth Otomasyon San & Tic. A..S. Fevzi Cakmak Cad No. 3 34295 Sefaköy Istanbul Tel.: +90 212 411-13 00 Fax: +90 212 411-13 17 www.boschrexroth.com.tr	Servo Kontrol Ltd. Sti. Perpa Ticaret Merkezi B Blok Kat: 11 No: 1609 80270 Okmeydani-Istanbul Tel: +90 212 320 30 80 Fax: +90 212 320 30 81 remzi.sali@servokontrol.com www.servokontrol.com	**Tightening & Press-fit:** MTS, spol .s.r.o. 02755 Kriva 53 Tel.: +421 43 5819 161	DOMEL Otoki 21 64 228 Zelezniki Tel.: +386 4 5117 100 Fax: +386 4 5117 357 brane.ozebek@domel.si

Australien, Süd-Afrika - Australia, South Africa

Australia - Australien	Australia - Australien	South Africa - Südafrika	South Africa - Südafrika
AIMS - Australian Industrial Machinery Services Pty. Ltd. 28 Westside Drive Laverton North Vic 3026 Melbourne Tel.: +61 3 93 14 3321 Fax: +61 3 93 14 3329 **Hotlines: +61 3 93 14 3321** **+61 4 19 369 195** enquires@aimservices.com.au	Bosch Rexroth Pty. Ltd. No. 7, Endeavour Way Braeside Victoria, 31 95 Melbourne Tel.: +61 3 95 80 39 33 Fax: +61 3 95 80 17 33 mel@rexroth.com.au	TECTRA Automation (Pty) Ltd. 100 Newton Road, Meadowdale Edenvale 1609 Tel.: +27 11 971 94 00 Fax: +27 11 971 94 40 **Hotline: +27 82 903 29 23** georgv@tectra.co.za	**Tightening & Press-fit:** Jendamark Automation 76A York Road, North End 6000 Port Elizabeth Tel.: +27 41 391 4735

Asien - Asia (incl. Pacific Rim)

China

Shanghai Bosch Rexroth
Hydraulics & Automation Ltd.
BRC Service Department
No.55 Jinwan Road
Jinqiao Industry Zone
Shanghai 201206 - P.R.China
Tel.: +86 21 38824688-749
Fax: +86 21 50555447
brcservice@boschrexroth.com.cn

China

Shanghai Bosch Rexroth
Hydraulics & Automation Ltd.
4/f, Marine Tower
No.1, Pudong Avenue
Shanghai 200120 - P.R.China
Tel: +86 21 68 86 15 88
Fax: +86 21 68 86 05 99
+86 21 58 40 65 77
y.wu@boschrexroth.com.cn

China

Bosch Rexroth (China) Ltd.
Satellite Service Office Changchun
Rm. 1910, Guangming Building
No.336 Xi'an Rd., Chao Yang Distr.
Changchun 130061 - P.R.China
Tel.+Fax: +86 431 898 1129
Mobile: +86 139 431 92 659
shuhong.wang@boschrexroth.com.cn

China

Bosch Rexroth (China) Ltd.
Satellite Service Office Wuhan
No. 22, Pinglanju, Milanyuan, Golden Harbor
No. 236 Longyang Avenue
Economic & Technology Development Zone
Wuhan 430056 - P.R.China
Tel.+Fax: +86 27 84 23 23 92
Mobile: +86 139 71 58 89 67
ym.zhu@boschrexroth.com.cn

China

Bosch Rexroth (China) Ltd.
Beijing Representative Office
Xi San Qi Dong, De Sheng Mei Wai
Hai Dian District
Beijing 100096, P.R.China
Tel.: +86 10 82 91 22 29
Fax: +86 10 82 91 81 09
yan.zhang@boschrexroth.com.cn

China

Bosch Rexroth (China) Ltd.
Guangzhou Repres. Office
Room 3710-3716, Metro Plaza,
Tian He District, 183 Tian He Bei Rd
Guangzhou 510075, P.R.China
Tel.: +86 20 87 55 00 30
+86 20 87 55 00 11
Fax: +86 20 87 55 23 87
andrew.wang@boschrexroth.com.cn

China

Bosch Rexroth (China) Ltd.
Dalian Representative Office
Room 2005,Pearl River Int. Building
No.99 Xin Kai Rd., Xi Gang District
Dalian, 116011, P.R.China
Tel.: +86 411 83 68 26 02
Fax: +86 411 83 68 27 02
jason.tan@boschrexroth.com.cn

China

Tightening & Press-fit:
C. Melchers GmbH & Co
Shanghai Representation
13 Floor Est Ocean Centre
No.588 Yanan Rd. East
65 Yanan Rd. West
Shanghai 200001
Tel.: +86 21 63 52 88 48
Fax: +86 21 63 51 31 38
shermanxia@sh.melchers.com.cn

Hongkong

Bosch Rexroth (China) Ltd.
6th Floor,
Yeung Yiu Chung No.6 Ind Bldg.
19 Cheung Shun Street
Cheung Sha Wan,
Kowloon, Hongkong
Tel.: +852 27 86 46 32
Fax: +852 27 42 60 57
Paul.li@boschrexroth.com.cn

India - Indien

Bosch Rexroth (India) Ltd.
Electric Drives & Controls
Plot. No.96, Phase III
Peenya Industrial Area
Bangalore – 560058
Tel.: +91 80 41 17 0211
Fax: +91 80 83 97 374
pradeep.rs@boschrexroth.co.in

India - Indien

Bosch Rexroth (India) Ltd.
Electric Drives & Controls
Advance House, II Floor
Ark Industrial Compound
Narol Naka, Makwana Road
Andheri (East), Mumbai - 400 059
Tel.: +91 22 28 56 32 90
+91 22 28 56 33 18
Fax: +91 22 28 56 32 93
singh.op@boschrexroth.co.in

India - Indien

Tightening & Press-fit:

MICO
Hosur Road Adugodi
560 030 Bangalore Karnataki
Tel.: +91 80 22 99 28 86

India - Indien

Bosch Rexroth (India) Ltd.
S-10, Green Park Extension
New Delhi – 110016
Tel.: +91 11 26 56 65 25
+91 11 26 56 65 27
Fax: +91 11 26 56 68 87
koul.rp@boschrexroth.co.in

Indonesia - Indonesien

PT. Bosch Rexroth
Building # 202, Cilandak Commercial Estate
Jl. Cilandak KKO, Jakarta 12560
Tel.: +62 21 7891169 (5 lines)
Fax: +62 21 7891170 - 71
rudy.karimun@boschrexroth.co.id

Japan

Bosch Rexroth Corporation
Service Center Japan
2125-1 atsukawado-cho
Kasugai-shi Aichi-ken
486-0932, Japan
Tel.: +81 568 35 7701
Fax: +81 568 35 7705

Japan

Bosch Rexroth Corporation
Electric Drives & Controls
BOSCH Bldg. 4F, 3-6-7 Shibuya
Shibuya-ku, Tokyo
150-0002, Japan
Tel : +81 354 85 7240
Fax: +81 354 85 7241

Korea

Bosch Rexroth-Korea Ltd.
Electric Drives & Controls
1515-14 Dadae-Dong, Saha-gu
Pusan Metropolitan City, 604-050
Tel.: +82 51 26 00 741
Fax: +82 51 26 00 747
eunkyong.kim@boschrexroth.co.kr

Korea

Bosch Rexroth-Korea Ltd.
Electric Drives and Controls
Bongwoo Bldg. 7FL, 31-7, 1Ga
Jangchoong-dong, Jung-gu
Seoul, 100-391
Tel.: +82 234 061 813
Fax: +82 222 641 295

Korea

Bosch Rexroth-Korea Ltd.
Electric Drives & Controls
1515-14 Dadae-Dong, Saha-gu
Ulsan, 680-010
Tel.: +82 52 256-0734
Fax: +82 52 256-0738
keonhyun.jeong@boschrexroth.co.kr

Korea

Tightening & Press-fit:
KVT Co., Ltd.
901, Daeryung Techno Town 8
481-11 Gasan-Dong
Geumcheon-Gu
Seoul, 153-775
Tel.: +82 2 2163 0231 9

Malaysia

Bosch Rexroth Sdn.Bhd.
11, Jalan U8/82, Seksyen U8
40150 Shah Alam
Selangor, Malaysia
Tel.: +60 3 78 44 80 00
Fax: +60 3 78 45 48 00
hhlim@boschrexroth.com.my

Singapore - Singapur

Bosch Rexroth Pte Ltd
15D Tuas Road
Singapore 638520
Tel.: +65 68 61 87 33
Fax: +65 68 61 18 25
raymond.toh@boschrexroth.com.sg

Taiwan

Bosch Rexroth Co., Ltd.
Taichung Industrial Area
No.19, 38 Road
Taichung, Taiwan 407, R.O.C.
Tel : +886 - 4 -235 08 383
Fax: +886 - 4 -235 08 586
jim.lin@boschrexroth.com.tw
david.lai@boschrexroth.com.tw

Taiwan

Bosch Rexroth Co., Ltd.
Tainan Branch
No. 17, Alley 24, Lane 737
Chung Cheng N.Rd. Yungkang
Tainan Hsien, Taiwan, R.O.C.
Tel : +886 - 6 –253 6565
Fax: +886 - 6 –253 4754
charlie.chen@boschrexroth.com.tw

Thailand

NC Advance Technology Co. Ltd.
59/76 Moo 9
Ramintra road 34
Tharang, Bangkhen,
Bangkok 10230
Tel.: +66 2 943 70 62
+66 2 943 71 21
Fax: +66 2 509 23 62
Hotline +66 1 984 61 52
sonkawin@hotmail.com

Nordamerika – North America

USA Headquarters - Hauptniederlassung Bosch Rexroth Corporation Electric Drives & Controls 5150 Prairie Stone Parkway Hoffman Estates, IL 60192-3707 Tel.: +1 847 645-3600 Fax: +1 847 645-6201 servicebrc@boschrexroth-us.com repairbrc@boschrexroth-us.com	USA Central Region - Mitte Bosch Rexroth Corporation Electric Drives & Controls 1701 Harmon Road Auburn Hills, MI 48326 Tel.: +1 248 393-3330 Fax: +1 248 393-2906	USA Southeast Region - Südost Bosch Rexroth Corporation Electric Drives & Controls 2810 Premiere Parkway, Suite 500 Duluth, GA 30097 Tel.: +1 678 957-4050 Fax: +1 678 417-6637	**USA SERVICE-HOTLINE** - 7 days week/ 24 hrs day - **+1-800-REXROTH** **+1 800 739 7684**
USA Northeast Region – Nordost Bosch Rexroth Corporation Electric Drives & Controls 99 Rainbow Road East Granby, CT 06026 Tel.: +1 860 844-8377 Fax: +1 860 844-8595	USA West Region – West Bosch Rexroth Corporation Electric Drives & Controls 7901 Stoneridge Drive, Suite 220 Pleasanton, CA 94588 Tel.: +1 925 227-1084 Fax: +1 925 227-1081	Canada East - Kanada Ost Bosch Rexroth Canada Corporation 5345 Outer Drive unit 5 Windsor, Ontario Canada N9A 6J3 Tel.: +1 519 737 7393 Fax.: +1 519 737 9999	Canada East - Kanada Ost Bosch Rexroth Canada Corporation Automation Division 3426 Mainway Drive Burlington, Ontario Canada L7M 1A8 Tel.: +1 905 335 5511 Fax: +1 905 335 4184 (Main) +1 905 335 9803 (Serv.) automation.service@boschrexroth.ca automation.repair@boschrexroth.ca
Canada West - Kanada West Bosch Rexroth Canada Corporation 5345 Goring St. Burnaby, British Columbia Canada V7J 1R1 Tel. +1 604 205 5777 Fax +1 604 205 6944 automation.service@boschrexroth.ca automation.repair@boschrexroth.ca	CANADA SERVICE HOTLINE - 7 days week/ 24 hrs day - **+1 905 335 5511**	Mexico Bosch Rexroth Mexico S.A. de C.V. Calle Neptuno 72 Unidad Ind. Vallejo 07700 Mexico, D.F. Tel.: +52 55 57 54 17 11 Fax: +52 55 57 54 50 73 mario.hernandez@boschrexroth.com.mx	Mexico Bosch Rexroth S.A. de C.V. Calle Argentina No 3913 Fracc. las Torres 64930 Monterrey, N.L. Tel.: +52 81 83 49 80 91 +52 81 83 49 80 92 +52 81 83 49 80 93 Fax: +52 81 83 65 52 80

Südamerika – South America

Argentina - Argentinien Bosch Rexroth S.A.I.C. "The Drive & Control Company" Rosario 2302 B1606DLD Carapachay Provincia de Buenos Aires Tel.: +54 11 4756 01 40 +54 11 4756 02 40 +54 11 4756 03 40 +54 11 4756 04 40 Fax: +54 11 4756 01 36 +54 11 4721 91 53 victor.jabif@boschrexroth.com.ar	Argentina - Argentinien NAKASE SRL Servicio Tecnico CNC Calle 49, No. 5764/66 B1653AOX Villa Balester Provincia de Buenos Aires Tel.: +54 11 4768 42 42 Fax: +54 11 4768 42 42 111 **Hotline: +54 11 155 307 6781** nakase@nakase.com	Brazil - Brasilien Bosch Rexroth Ltda. Av. Tégula, 888 Ponte Alta, Atibaia SP CEP 12942-440 Tel.: +55 11 4414 -56 92 +55 11 4414 -56 84 Fax sales: +55 11 4414 -57 07 Fax serv.: +55 11 4414 -56 86 alexandre.wittwer@rexroth.com.br	Brazil - Brasilien Bosch Rexroth Ltda. R. Dr.Humberto Pinheiro Vieira, 100 Distrito Industrial [Caixa Postal 1273] 89220-390 Joinville - SC Tel./Fax: +55 47 473 58 33 Mobil: +55 47 9974 6645 sergio.prochnow@boschrexroth.com.br
Columbia - Kolumbien Reflutec de Colombia Ltda. Calle 37 No. 22-31 Santafé de Bogotá, D.C. Colombia Tel.: +57 1 208 65 55 Fax: +57 1 269 97 37 reflutec@etb.net.co			

Rexroth
Bosch Group

Bosch Rexroth AG
Electric Drives and Controls
P.O. Box 13 57
97803 Lohr, Germany
Bgm.-Dr.-Nebel-Str. 2
97816 Lohr, Germany
Phone +49 (0)93 52-40-50 60
Fax +49 (0)93 52-40-49 41
service.svc@boschrexroth.de
www.boschrexroth.com

R911312329

Printed in Germany
DOK-CONTRL-IC*L20*****-AW02-EN-P