AC500 PLC

AC500 V3 KNX IP Integration
PS5604-KNX

Example Project Description f. AB2.x.y

AC500 PLC

AC500 V3 KNX IP Integration PS5604-KNX

Example Project Description f. AB2.x

NOTICE

This document contains information about one or more ABB products and may include a description of or a reference to one or more standards that may be generally relevant to the ABB products. The presence of any such description of a standard or reference to a standard is not a representation that all of the ABB products referenced in this document support all of the features of the described or referenced standard. In order to determine the specific features supported by a particular ABB product, the reader should consult the product specifications for the particular ABB product.

ABB may have one or more patents or pending patent applications protecting the intellectual property in the ABB products described in this document.

The information in this document is subject to change without notice and should not be construed as a commitment by ABB. ABB assumes no responsibility for any errors that may appear in this document.

Products described or referenced in this document are designed to be connected and to communicate information and data through network interfaces, which should be connected to a secure network. It is the sole responsibility of the system/product owner to provide and continuously ensure a secure connection between the product and the system network and/or any other networks that may be connected.

The system/product owners must establish and maintain appropriate measures, including, but not limited to, the installation of firewalls, application of authentication measures, encryption of data, installation of antivirus programs, and so on, to protect these products, the network, its system, and interfaces against security breaches, unauthorized access, interference, intrusion, leakage, and/or theft of data or information.

ABB performs functionality testing on the products and updates that we release. However, system/product owners are ultimately responsible for ensuring that any product updates or other major system updates (to include but not limited to code changes, configuration file changes, third-party software updates or patches, hardware change out, and so on) are compatible with the security measures implemented. The system/ product owners must verify that the system and associated products function as expected in the environment in which they are deployed.

In no event shall ABB be liable for direct, indirect, special, incidental or consequential damages of any nature or kind arising from the use of this document, nor shall ABB be liable for incidental or consequential damages arising from use of any software or hardware described in this document.

The software or hardware described in this document is furnished under a license and may be used, copied, or disclosed only in accordance with the terms of such license. This product meets the requirements specified in EMC Directive 2014/30/EU and in Low Voltage Directive 2014/35/EU.

A. For customers domiciled outside Germany /

Für Kunden mit Sitz außerhalb Deutschlands

„Warranty, Liability:

The user shall be solely responsible for the use of this products described within this file. ABB shall be under no warranty whatsoever. ABB's liability in connection with application of the products or examples provided or the files included within this products, irrespective of the legal ground, shall be excluded. The exclusion of liability shall not apply in the case of intention or gross negligence. The present declaration shall be governed by and construed in accordance with the laws of Switzerland under exclusion of its conflict of laws rules and of the Vienna Convention on the International Sale of Goods (CISG)."

„Gewährleistung und Haftung:

Der Nutzer ist allein für die Verwendung des in diesem Dokument beschriebenen Produkte und beschriebenen Anwendungsbeispiele verantwortlich.

ABB unterliegt keiner Gewährleistung. Die Haftung von ABB im Zusammenhang mit diesem Anwendungsbeispiel oder den in dieser Datei enthaltenen Dateien - gleich aus welchem Rechtsgrund - ist ausgeschlossen. Dieser Ausschluß gilt nicht im Falle von Vorsatz oder grober Fahrlässigkeit. Diese Erklärung unterliegt Schweizer Recht unter Ausschluß der Verweisungsnormen und des UN-Kaufrechts (CISG)."

B. Nur für Kunden mit Sitz in Deutschland

„Gewährleistung und Haftung:

Die in diesem dokument beschriebenen Anwendungsbeispiele oder enthaltenen Dateien beschreiben eine mögliche Anwendung der AC500 bzw. zeigen eine mögliche Einsatzart. Sie stellen nur Beispiele für Programmierungen dar, sind aber keine fertigen Lösungen. Eine Gewähr kann nicht übernommen werden.

Der Nutzer ist für die ordnungsgemäße, insbesondere vollständige und fehlerfreie Programmierung der Steuerungen selbst verantwortlich. Im Falle der teilweisen oder ganzen Übernahme der Programmierbeispiele können gegen ABB keine Ansprüche geltend gemacht werden.

Die Haftung von ABB, gleich aus welchem Rechtsgrund, im Zusammenhang mit den Anwendungsbeispielen oder den in dieser Datei enthaltenen Beschreibung wird ausgeschlossen. Der Haftungsausschluß gilt jedoch nicht in Fällen des Vorsatzes, der groben Fahrlässigkeit, bei Ansprüchen nach dem Produkthaftungsgesetz, im Falle der Verletzung des Lebens, des Körpers oder der Gesundheit oder bei schuldhafter Verletzung einer wesentlichen Vertragspflicht. Im Falle der Verletzung einer wesentlichen Vertragspflicht ist die Haftung jedoch auf den vertragstypischen, vorhersehbaren Schaden begrenzt, soweit nicht zugleich ein anderer der in Satz 2 dieses Unterabsatzes erwähnten Fälle gegeben ist. Eine Änderung der Beweislast zum Nachteil des Nutzers ist hiermit nicht verbunden.

Es gilt materielles deutsches Recht unter Ausschluß des UN-Kaufrechts."

TRADEMARKS

Release: August 2017

Document number: 3ADR010265_1_en_US

TABLE OF CONTENTS

LIST OF FIGURES

1. INTRODUCTION

1.1 Scope of the document

This document describes the use of the AC500 V3 PLCs interface to KNX and is intended for system integrators who program functionality in IEC61131 type languages. In addition, the information contained also supports the planning of KNX systems in which this controller is used.

Prerequisites for understanding this document are basic knowledge of KNX / ETS and IEC 61131-3 / programming. For more information see chapter 4.1 Support and training.

This document describes an example project and all relevant steps and menus of the Automation builder and ETS engineering Tools to result in a running demo system.

1.2 Overview of the System

The AC500 V3 PLC with its well-known and standardized IEC61131 programming can now via KNX be seamlessly integrated into room automation. The tight integration of the Automation Builder and KNX Engineering reduces commissioning and maintenance efforts.

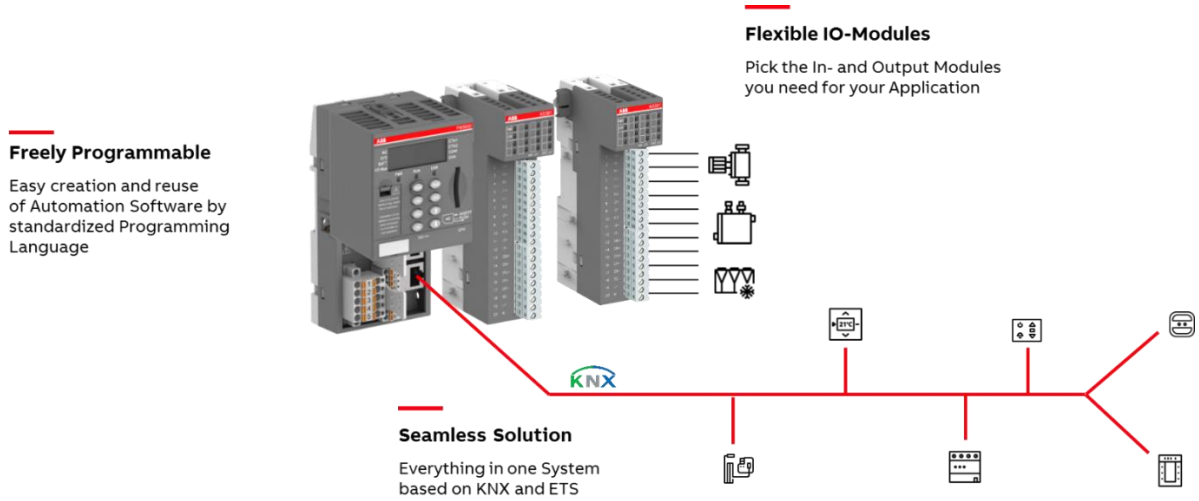


Figure 1 Schematic product overview

The AC500 as a modular and highly flexible PLC with scalable inputs and outputs and its general communication options is designed for the automation of complex logic, control and multiple parallel communications as required in central systems with many signals, actuators or powerful and flexible visualization capabilities on its webserver or via the CP600 modular panel series.

For example: Automation of HVAC type applications like Air Handling Units (AHU), central heating and cooling or general energy management and monitoring applications. Its inbuild OPC UA Server and MQTT secure communication options enable also remote and cloud connectivity.
ABB as a leading supplier in KNX offers a vast and specially designed complete KNX product portfolio for distributed room automation with KNX sensors, actuators and gateways.
(see http://www.abb.com/knx) These room devices are via KNX IP seamlessly connectable now to the AC500 e.g. in the central automation level in Building Automation.

- Fully integrated engineering in KNX and ETS. The AC500 controller is a certified KNX device with group objects in ETS and a physical KNX address
- Direct data exchange between the Engineering Software ABB Automation Builder and ETS

1.3 Safety Instructions and Preconditions

The user has to read the following instructions and documents before using the libraries:

- All pertinent state, regional, and local safety regulations must be observed when installing and using this product. When functions or devices are used for applications with technical safety requirements, the relevant instructions must be followed.
- Read the complete safety instructions of the user's manuals for the devices you are using, before installation and commissioning.
- Read all safety instructions of the AC500 PLC. See System description AC500 in the online help in Automation Builder
- Read the user Information of the devices and functions you are using, see online help in Automation Builder.

The …Library package has been released for the software and firmware versions listed in the Readme file of the package only.

In no event will ABB or its representatives be liable for loss of data, profits, revenue or consequential, incidental or other damage that may result from the use of other versions of product, software or firmware versions. The error-free operation of the High Availability Modbus TCP Library with other devices, software or firmware versions should be possible but cannot be guaranteed and may need adaptations e.g. of example programs.

The user must follow all applicable safety instructions and the guidelines mentioned in the user documents of the ABB products.

Read the complete safety instructions for the AC500 before installation and commissioning.

	CAUTION! Generally, the user in all applications is fully and alone responsible for checking all functions carefully, especially for safe and reliable operation.
	Note: The Function Blocks contained in some libraries may only be executed in RUN mode of the PLC, but not in simulation mode.

2. BASICS

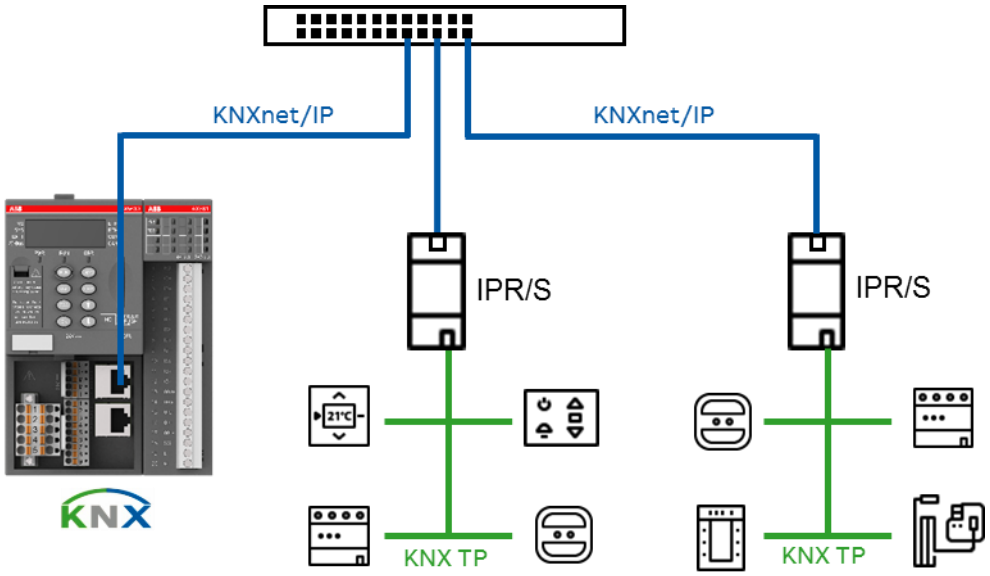


Figure 2: System Topology

The V3 AC500 CPU is after the Automation Builder engineering step a standard KNX device, in which KNX communication is done via the IP network interface. It is arranged topologically on the area / main line of the KNX IP routers and communicates with them via the KNXnet/IP protocol.

The described Example Project uses the network interface Ethernet 1 (upper connection) as activated Port for the KNX com-unication and the web interface.

2.1 Overview of Engineering

Programming and commissioning of the Building Automation Controller is done with the engineering software ABB Automation Builder (abbreviated as AB). This includes, among others

- Configuration of the AC500 and its communications and input and output modules and all necessary parameters like e.g. Set the IP network settings
- Creation of the automation program
- Creation of the KNX group object
- Firmware update of the Building Automation Controller
- Download of the above into the AC500 Controller

The subsequent linking of the AC500 and the other KNX devices takes place with the vendor independent KNX commissioning software ETS. This includes

- Programming the physical KNX address of the AC500
- Parameterizing the KNX send conditions of the AC500
- Linking the KNX group objects of the AC500 with KNX group addresses
- Download this KNX configuration listed above to the AC500

The programming of the AC500 and the KNX commissioning can be done also by different people at different times. Both programs carry out their own download parts of their respective configurations to the AC500. The only data exchange between the two Engineering programs are the details about the KNX group objects defined in the ABB Automation Builder. This is done flexibly via a configuration file.

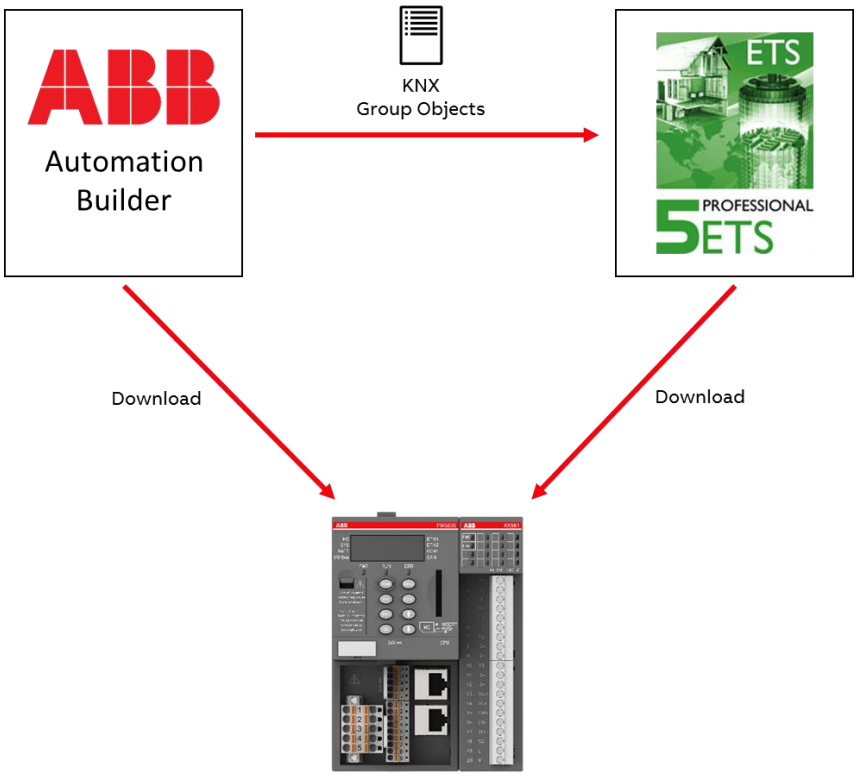


Figure 3: Data exchange

This has the further advantage that changes are possible at any time without the respective other program. For example, the KNX group objects associated with the AC500 can be changed without the ABB Automation Builder being necessary. On the other hand, the automation program can also be changed without the ETS. As long as no changes have been made to the KNX group objects.

The following chapter describes in detail the individual steps of commissioning the AC500.

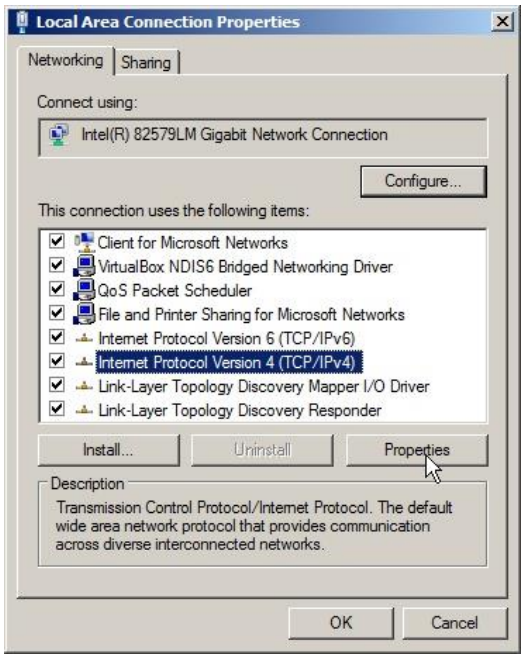
Both programs are fee-based. There are demo versions offered.

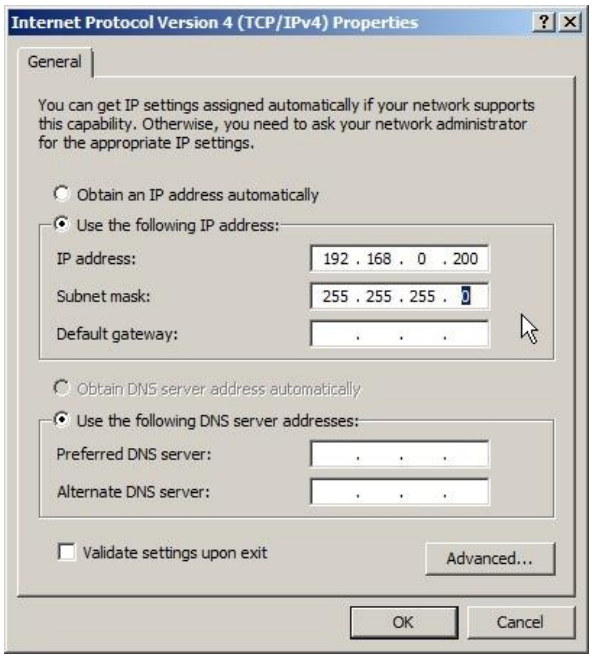
2.2 Prerequisites

- PC(s) with Windows 7 or higher with Administrator right(s)
- At least temporary network access to the internet - for downloading and installing of
    - ABB Automation Builder, (and example .project)
    - ETS and the necessary additions (DCA .etsapp, .knxprod, example .knxproj).
- Network access to the local network, were the AC500 and KNX devices are connected.

The current IP address of the engineering PC(s) where Automation Builder and ETS are located in same Network / masked IP range, this can be determined/set:

- Start > Control Panel > Network and Sharing Center, select the used Local Area Connection and enter the button Properties.
- Select the "Internet Protocol Version 4 (TCP/IP)" and enter the button Properties.
- Select "Use the following IP address"





The later being prepared IP address of the AC500 must be in the same address area as those, of the connected PC(s).

- Components as used in the example projects (please check matching AB Version 2.x.y
  (AB: PS5604-KNX_V2-x-y_Example_project.project;
  ETS: PS5604-KNX_V2-x-y_Exampleproject.knxproj;)
  - AC500 V3 CPU, e.g. PM5630-2ETH with Terminal Base, e.g. TB5600-2ETH or higher
  - PS5604-KNX run time license (1SAP195800R0101)
  - Power supply unit 24 DC for supply of the AC500 and I/O module, e.g. CP-D 24/4.2
  - IPR/S3.1.1 IP Router (connects the KNX bus with the Ethernet network)
  - KNX Switch Actor, e.g SA/S4.6.2.1 (or SA/S410.2.1) Switch Act, 4F, 10A
  - KNX PowerSupply, e.g. SV/S30.320.2.1 PowerSupply, diagn. 320mA
  - ETS license Lite or Professional (DCA use not possible w. Demo)

3. COMMISSIONING

3.1 Installation of the Engineering Software

3.1.1 Automation Builder

Download the latest ABB Automation Builder from the product website of the AC500Controller. You will find these at

www.abb.com/plc choosing the Links “Automation Builder” and then “Download”.

In the Installation Manager Menu “PLC-AC500 V3” and “KNX” should be selected as minimal options to install. Other optional program packages as shown in Figure 4 can be selected as needed and also updated later by starting the Installation Manager (from inside AB under tools or as a separate installed Windows program under “ABB”/”Automation Builder”)).

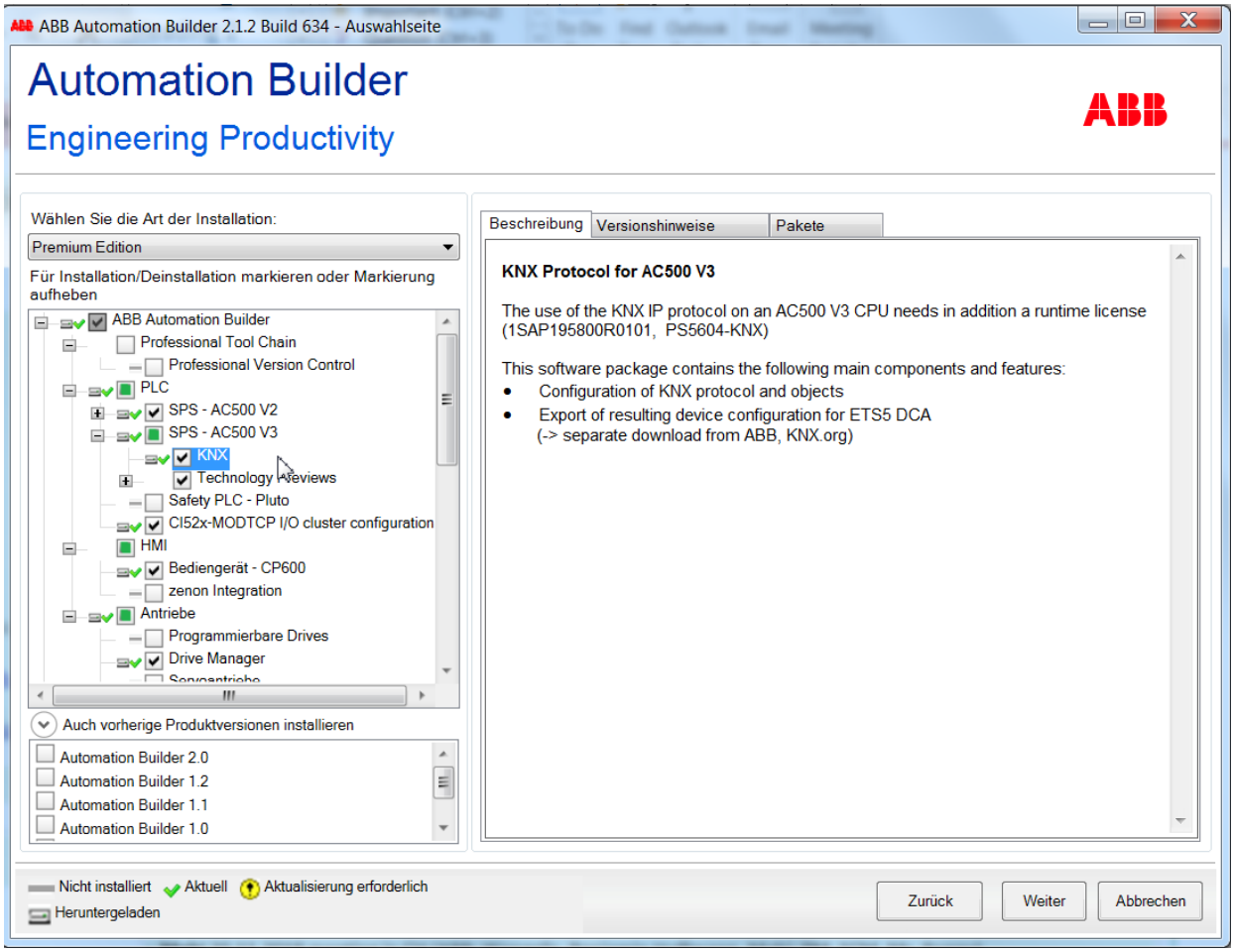


Figure 1: Installation of Automation Builder for Building Automation

Please use Automation Builder version 2.2.0 or later (if already available). As of version 2.1.2, KNX is supported in the Automation Builder. However, this 2.1.2 release contains an outdated implementation of the status flags and ValueRead function, which is not compatible with the following versions. The difference is described in the following chapters.

The ABB Automation Builder requires currently the Standard license, but also offers a demo mode in which the functionality can be fully tested for 30 days.

3.1.2 ETS

- Download the latest ETS version from www.knx.org (ETS 5.7.1 or newer).
- The integration of the ABB Automation Builder into the ETS is done by the Device Configuration App (DCA) “ABB Automation Builder” (see below). Please download this DCA from
  - the ETS app shop at www.knx.org (or in ABB Library if you have access)
- Install it as shown in Figure 5 by clicking on Apps in the lower right corner of the ETS and clicking on the + icon to add.

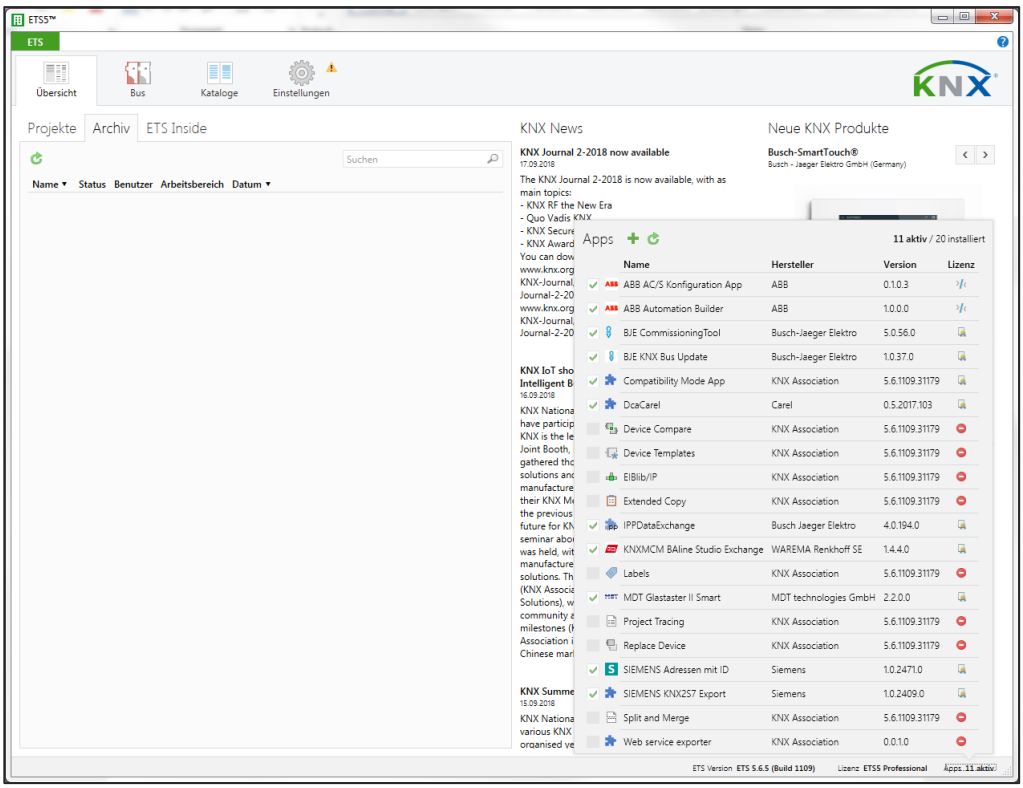


Figure 4: Installation of the DCA in ETS

To install and run DCAs, you need the ETS license Lite or Professional. DCAs are not available in the ETS5 demo version.

- Import the ETS application of the AC500 into the ETS catalog, either via the online function of the ETS catalog or as a download of the .knxprod file from www.abb.de/knx
  - “PS5604-KNX AC500 KNX runtime license.knxprod”,

3.2 Programming the Controller

3.2.1 Automation Builder Project

This example description assumes that you have the example project: “*PS5604-KNX_V2-x-y_Example.project*” (x.y. denot to the Automation builder Version AB2.x.y. …there are several adapted projects) either

- From the Automation Builder Example Project folder on your PC (select the Example matching the Automation Builder version if there are several) under ….Documents/Public Documents/Automation Builder/Examples/PS5604-KNX.
- Or download it from (https://new.abb.com/products/1SAP195800R0101/ps5604-knx-ac500-v3-ip-runtime-license under “Downloads” tab.)

In this example project the AC500 is preconfigured as PM5630 which can be changed to any other V3 CPU type.

If you start with your own empty Automation Builder Project, you must select a correct V3 CPU and insert the KNX communication interface by right click add object, either to the network interface 1 (ETH1) or network interface 2 (ETH2).

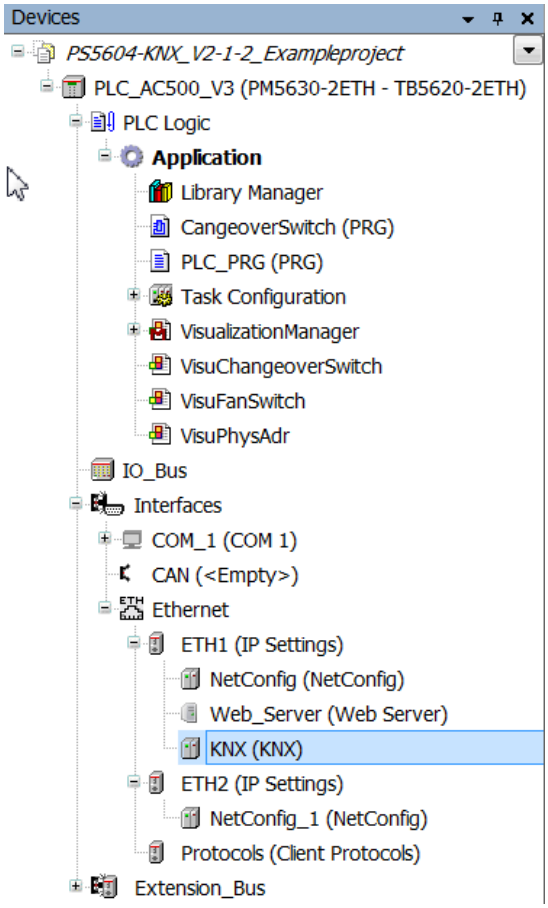


Figure 5: KNX at network interface ETH1

3.2.2 Update FW and verify IP Address of the CPU

In the delivery state, the V3 CPU needs an actual Firmware which has to be downloaded first.

Then it has the standard IP network settings at the network interface 1

IP Address: 192.168.0.10

Subnet Mask: 255.255.255.0

You can change the IP network settings via the display of the CPU or via the ABB Automation Builder:

Automation Builder

You can also change the network settings in the Automation Builder using the menu item Tools “IP-Configuration”. Perform a scan on the network, and then select the CPU. Using Send Configuration, the network settings can be sent to the controller. To do this, the CPU must be stopped by pressing the RUN button on the controller until the display prompt STOP.

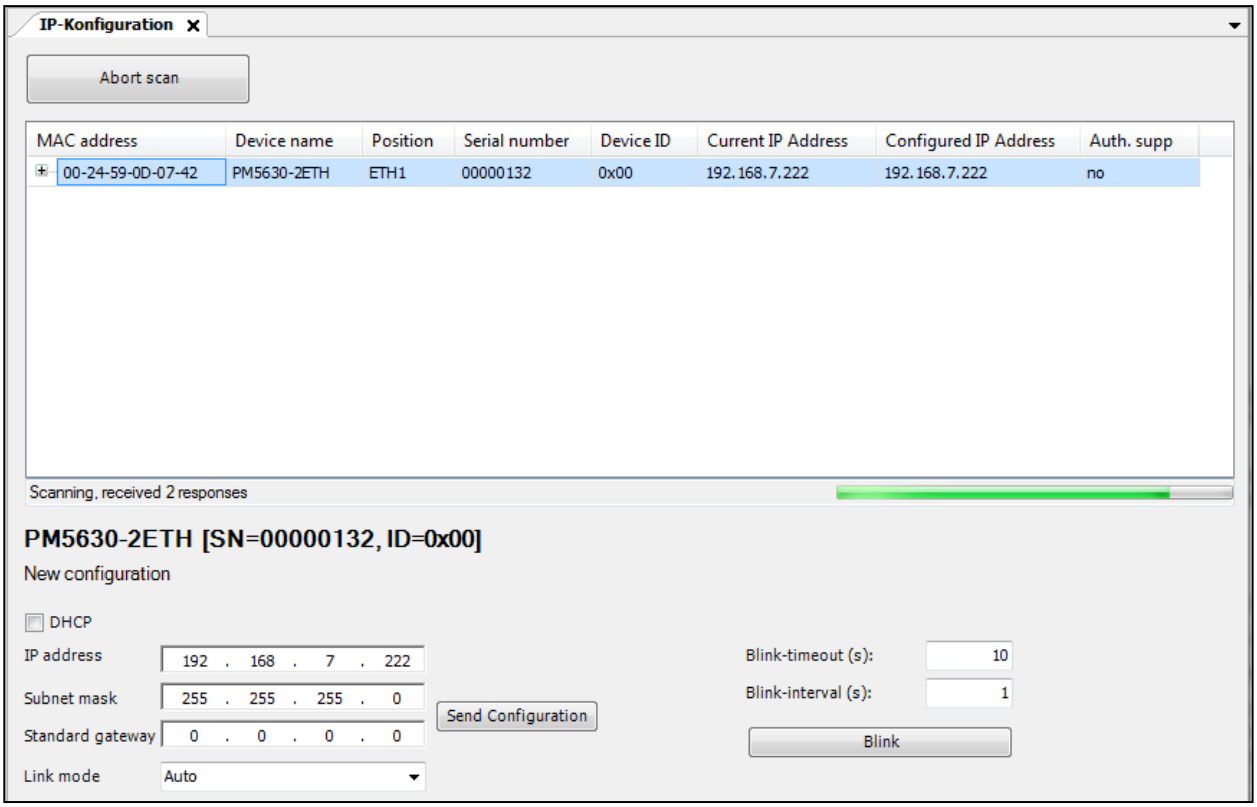


Figure 6: IP device scan and configuration

Display

1. Press the CFG button
2. Confirm the display prompt 1PEth1 with the OK button
3. Press the CFG button to select whether to set a static IP address (static), reset (reset) or automatic (DHCP) IP configuration. Confirm with the OK button.
4. For a static IP address, the IP address and the subnet mask are displayed one after the other by pressing the CFG button. The addresses are displayed block by block in the display as follows

   IP Address: A1.A2.A3.A4

   Subnet mask: n1.n2.n3.n4

   You can change these with the arrow keys and save them with the OK button.

With the ESC button you can leave the menu at any time without saving.

For detailed instructions, see the ABB Automation Builder Help.

Once the desired IP network setting has been set for the AC500 CPU, this network address must also be made known to the Automation Builder Project. To do this, right click on the CPU and select “Communication Settings”.

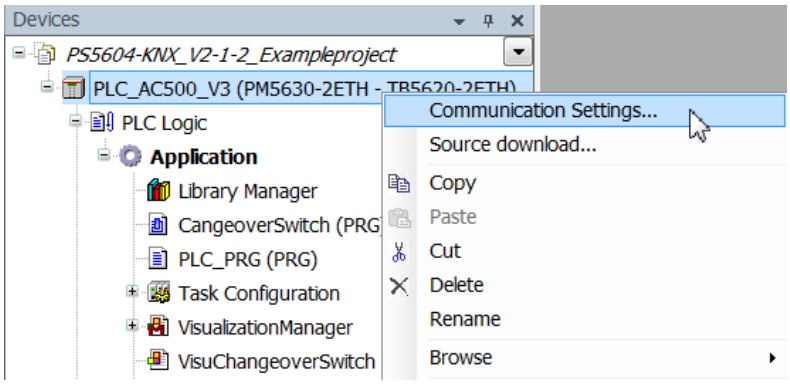


Figure 7: Communication Settings

Enter the IP address there or select the Controller found in the network by clicking on the "..." button on the right side of the address field.

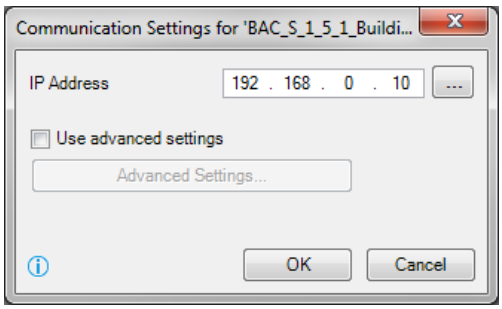


Figure 8: Set network settings

3.2.3 Configure input and output modules

If you want to add input and output modules to the CPU, click on the *Add object* function of the *IO_Bus* in the device tree under the CPU.

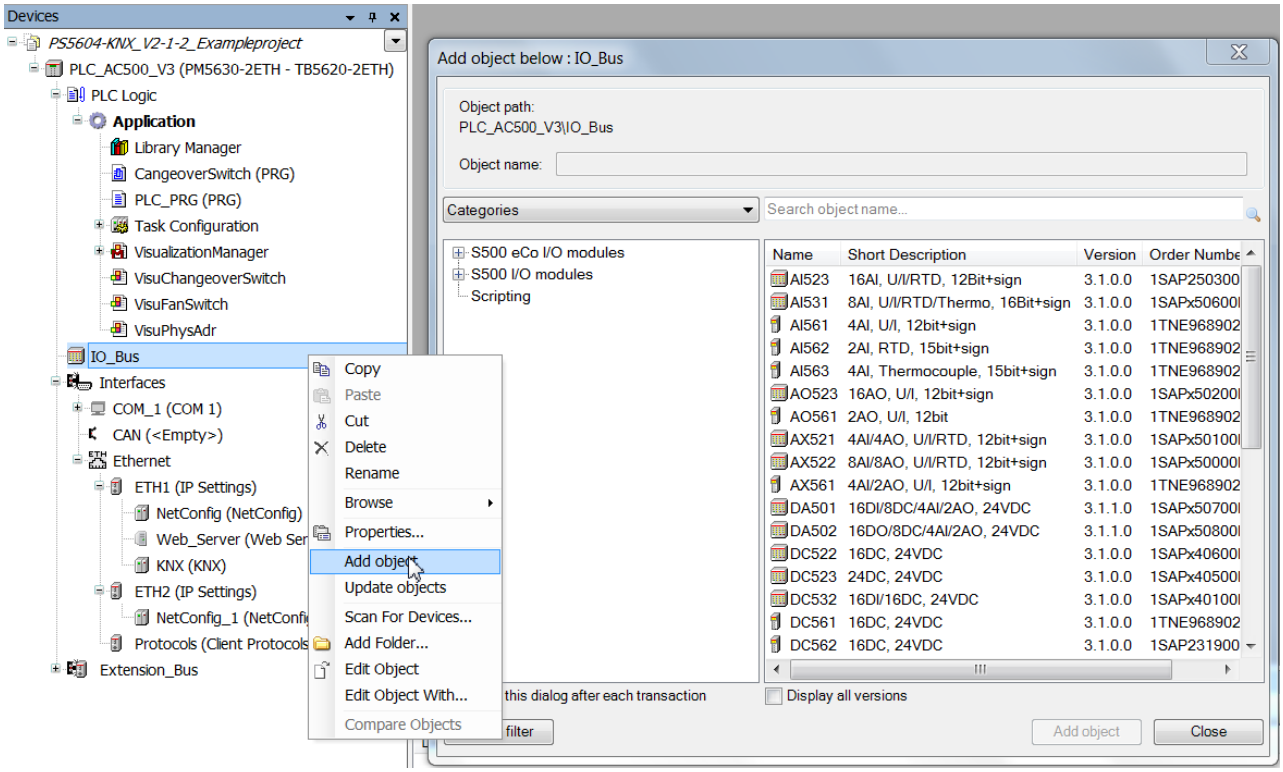


Figure 9: Add input / output module

The order of the I/O modules in the tree view of the Automation Builder must be in the order in which they are mechanically connected to the Controller.

3.2.4 Create KNX Group Objects

The data exchange with the KNX bus is done via KNX Group Objects.

Check existing ones in your example project or create new KNX Group Objects required for your application in the parameters of the *KNX* interface below the network interface 1 (*ETH1*) on the KNX parameter page *General*.

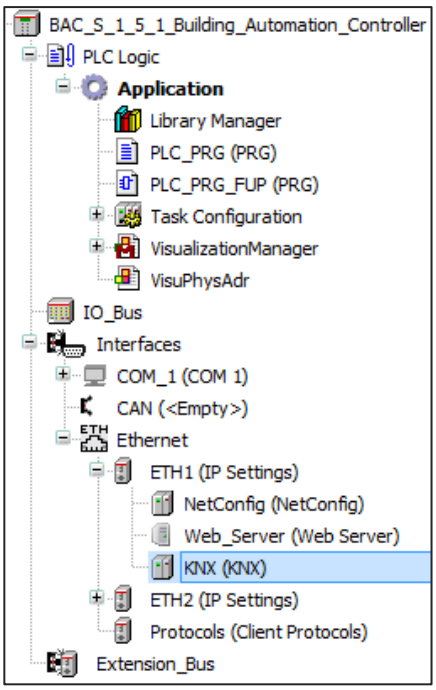


Figure 10: Open the KNX settings

With the Add button you can create a new KNX Group Object with the following properties as shown in Figure 13:

- **Channel Number**: The number of the KNX Group Object. This must be unique within the Controller. It is dis-played in the ETS and influences the display order in the ETS and the Automation Builder.
- **Group Object Name**: The name of the KNX Group Object. It is freely selectable and is displayed in the ETS under the field name.
- **Type**. Specification of the KNX data point type (DPT) of the Group Object. This determines the memory size, scaling and unit. For further information see the KNX Standard.
- **Group Object Function**: The name of the function of the Group Object. It is freely selectable and is displayed in the ETS under the field Function.
- **Input / Output**: Selection of the communication direction.
  - Input means that the controller receives values from the KNX bus.
  - Output means that the controller sends values to the KNX bus.

Based on this selection, the flags of the KNX Group Object are set accordingly in the ETS.

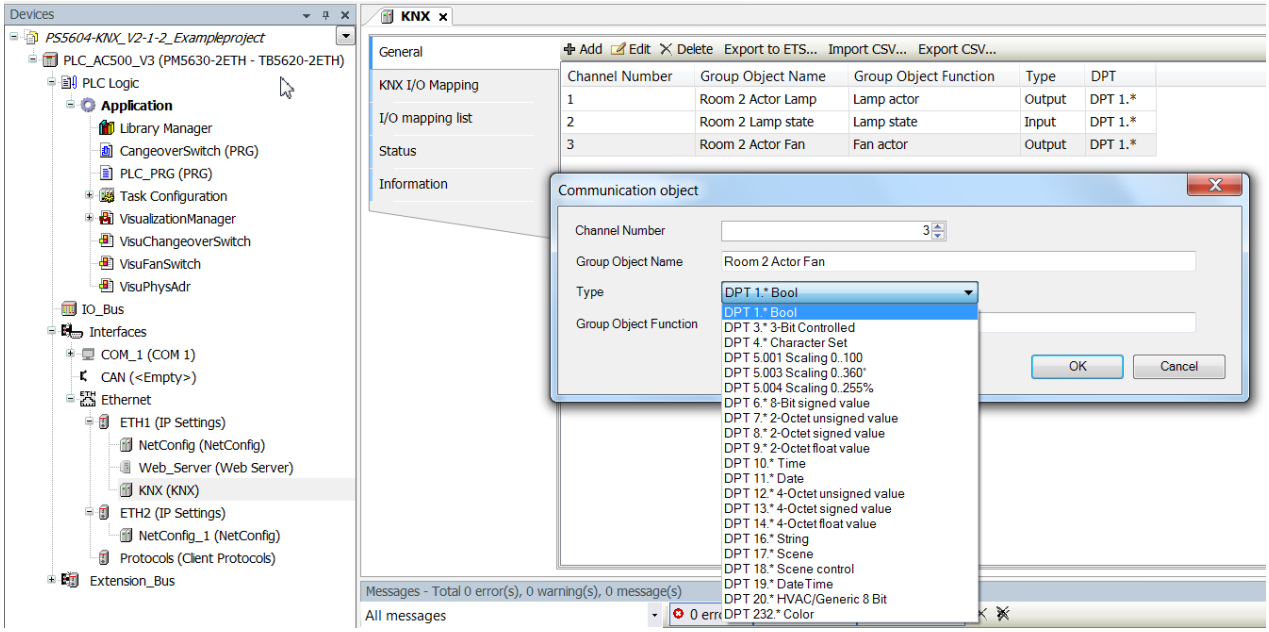


Figure 11: Creation of the KNX Group Objects

You can use the Export CSV... button in the menu bar to display the list of KNX Group Objects in a spreadsheet program such as Excel and edit and extend it flexibly. Then you can import them again via Import CSV.

After you have created all the required KNX Group Objects, export them using the Export to ETS button. This exported file contains the configuration of the KNX Group Objects of the AC500Controller and is imported by ETS for linking to other KNX devices. If you have not yet created project information under Project / Project Information, the default values will be used during the export.

To use these KNX Group Objects in your automation program, you must assign them with IEC61131-3 variables. This additional abstraction layer of an additional variable allows you to create modular automation programs that are independent of the used bus system or input / output modules.

The assignment is possible either via the parameter page KNX I/O Mapping or I/O mapping list. Both pages offer the same function in different representation.

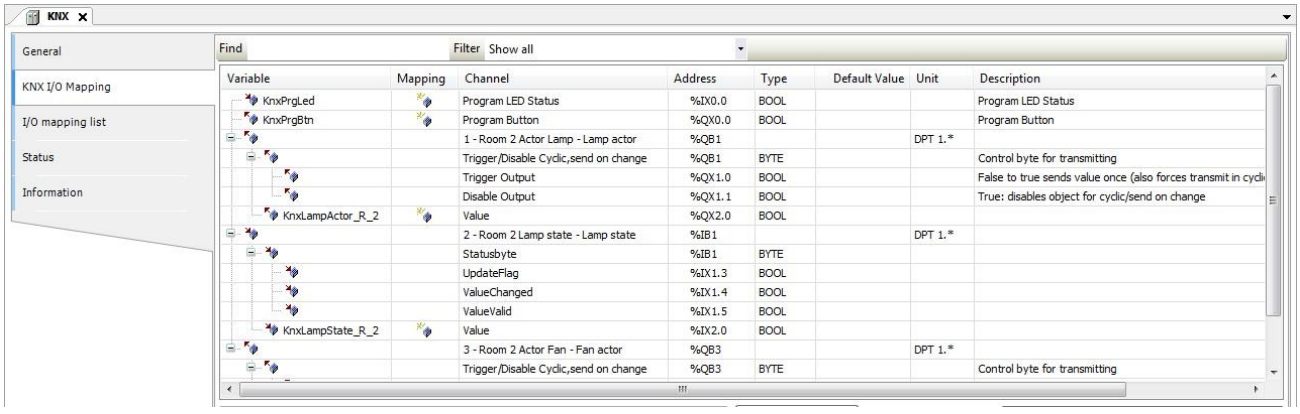


Figure 12: Assignment of the IEC 61131-3 variables to the KNX Group Objects

On the KNX I/O Mapping page, the KNX variables are shown hierarchically. Each KNX Group Object consists of several channels with additional information. These differ depending on whether it is an input or an output.

The table is structured as follows:

- **Variable**: Enter the name of the IEC 61131-3 variable that you want to assign to this channel (KNX Group Object).

- **Mapping**: Shows if the channel is already linked
- Channel: Name of the Channel
- **Address**: The memory address under which the information is stored in the memory of the Control-ler. Inputs start with %I and outputs start with %Q.
- **Typ**: Specification of the IEC 61131-3 variable type
- **Default Value**: The value used after starting the controller.
  - At a KNX Group Object input, this value is used by the automation program until a value has been received from the KNX bus.
  - At a KNX Group Object output, this value is sent to the bus when the controller is started.
- **Unit**: Specification of the KNX data point type (DPT)
- **Description**: Note text

A **KNX Group Object input** consists of the following channels:

Channel name: Object Number + Object Name + Object Function

- **UpdateFlag**: This status flag is set to the value true as soon as a new KNX telegram has been received. Even if the value of the telegram does not differ from the previous one.
- V**alueChanged**: This status flag is set to the value true as soon as a new KNX telegram has been received and the value differs from the previous one.
- **ValueValid:** This status flag is set to the value true as soon as a KNX telegram has been re-ceived for the first time after the controller has been started.
- Value: The current value of the KNX Group Object received from the KNX bus.

Channel name: “Control” + Object Number + Object Name + Object Function

- **Reset status flags**: When this flag is set from false to true by the automation program then the above-mentioned status flags of the KNX Group Object are reset to the value false.
- **Set read flag**: When this flag is set from false to true by the automation program, a *ValueRead* telegram is sent to the KNX bus. This causes the KNX remote device to send back its current value.

Attention: This information refers to Automation Builder version 2.2.0 or later. In version 2.1.2 there are no "Control" channels. Reset Status flags and Set read flags must be called there by a function from the automation program referring the KNX Group Object number.

A **KNX Group Object output** consists of the following channels:

Channel name: Object Number + Object Name + Object Function

Trigger Output: When this flag is set from false to true by the automation program, the cur-rent value is sent immediately to the KNX bus.

The sending conditions that are may been activated in the ETS (send on change and cyclic sending) will be restarted

Disable Output: As long this flag is set to true by the automation program, the sending conditions send on change and cyclic sending in the ETS are deactivated.

Value: The current value of the KNX Group Object that is sent to the KNX bus.

The permanently defined channels Program Button and Program LED Status represent the functions to switch the controller into the KNX programming mode with which the physical KNX address can

be programmed. In the Automation Builder *example* and *template* project, these are already linked to the corresponding function as well as to the variables of the *VisuPhysAdr* web page and the display of the CPU (starting with AB2.2 and newest FW only!).

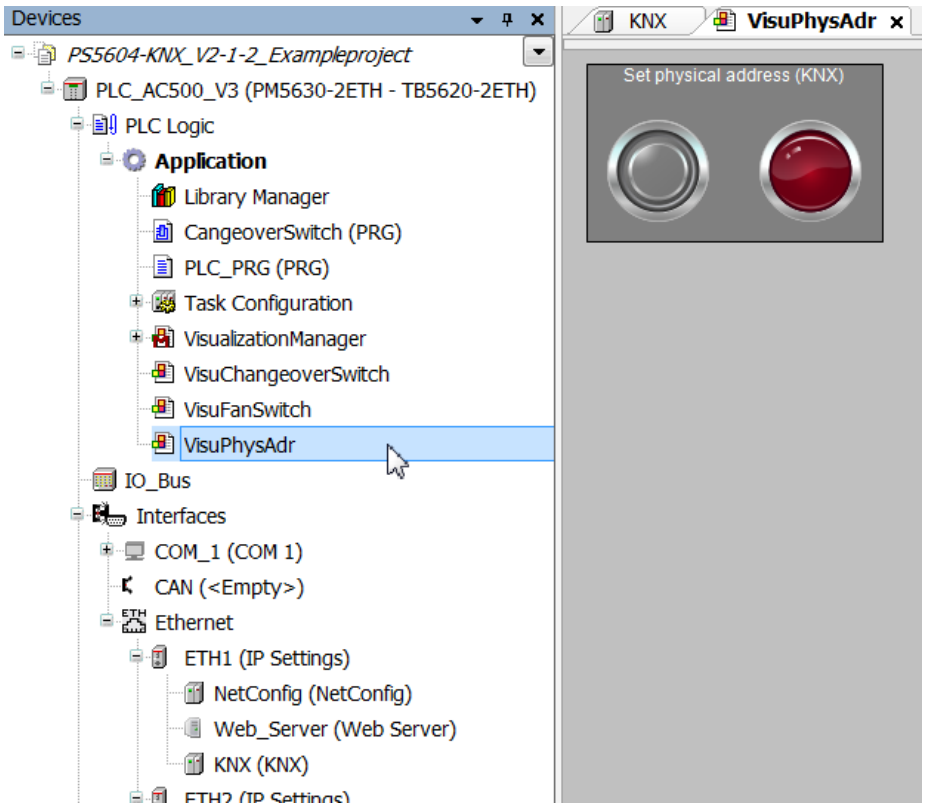


Figure 13: Webpage for the KNX programming button

Attention: If you want to control the KNX programming mode via your automation program or web user interface then use from ABB Automation Builder Version AB 2.2.0 or later the function call instead of the IEC 61131-3 variables *Program Button* and *Program LED Status*. Otherwise these variables are continuously overridden by the display of the Controller.

3.2.5 Create AC500 Automation Program

The KNX variables defined on the KNX I/O Mapping page are available program-wide under *Io-Config_Globals_Mapping*. These you can see if you click in to the programming window and either via right-mouse click select “input assistant” or press F2

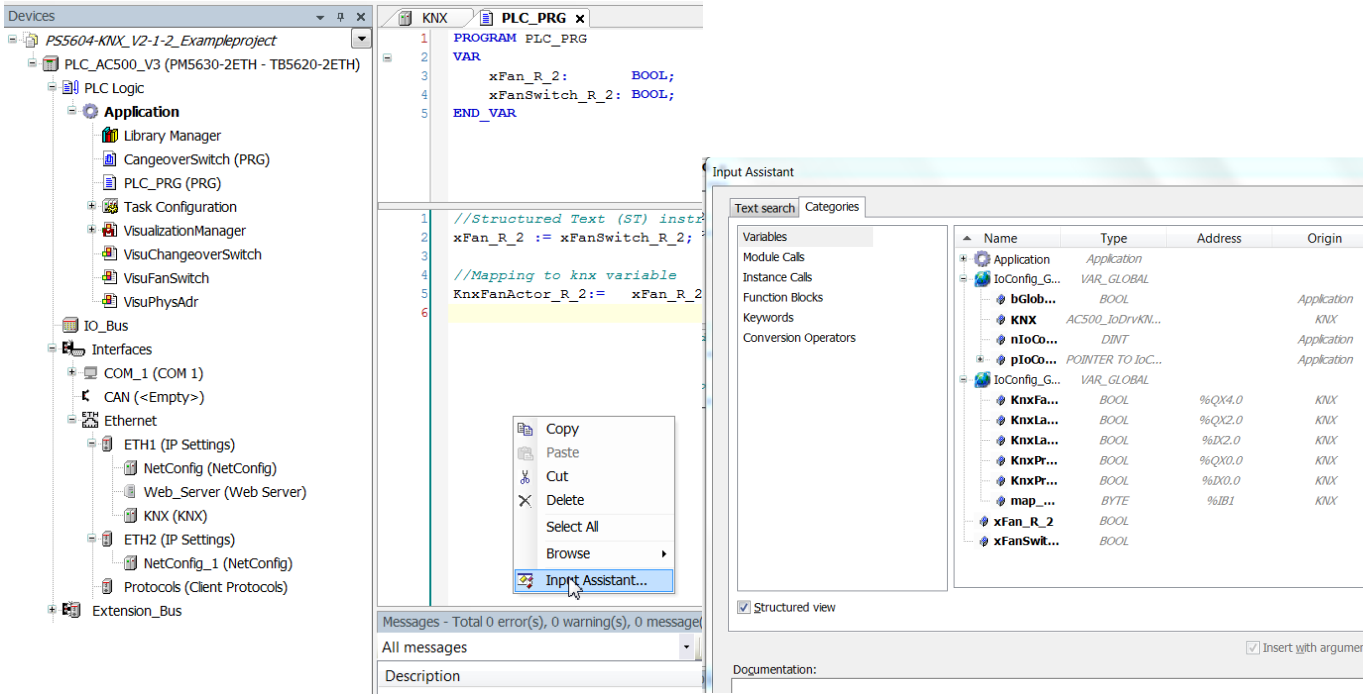


Figure 14a: Variable listing

Various programming languages are available for creating your automation program. A very simple program PLC_PRG (PRG) for the structured text programming language has already been created in the Automation Builder template project.

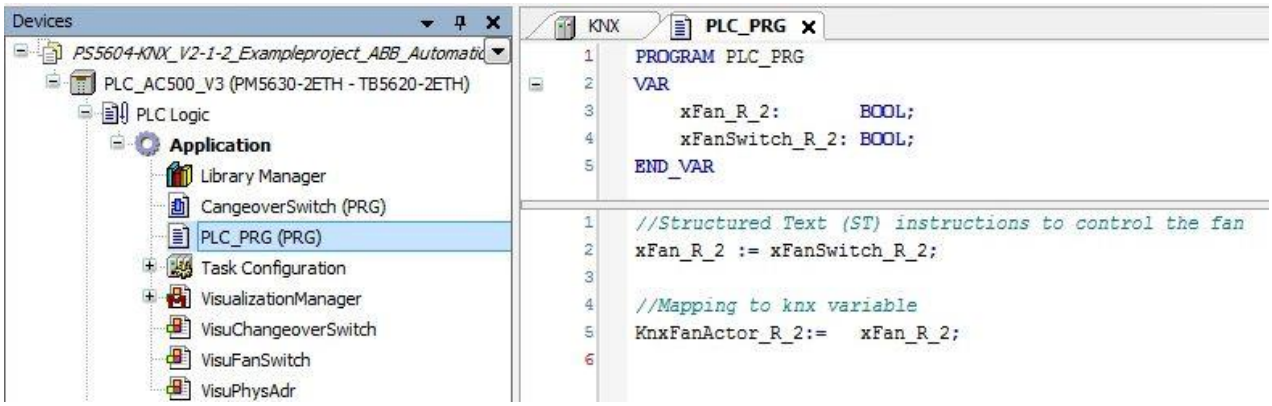


Figure 15b: Example program in "structured text" programming language

By right-clicking on the entry *Application* in the tree with Add object, you can create new programs and software functions. To do this, select the object POU in the window that opens in Figure 16 and select afterwards the desired programming language.

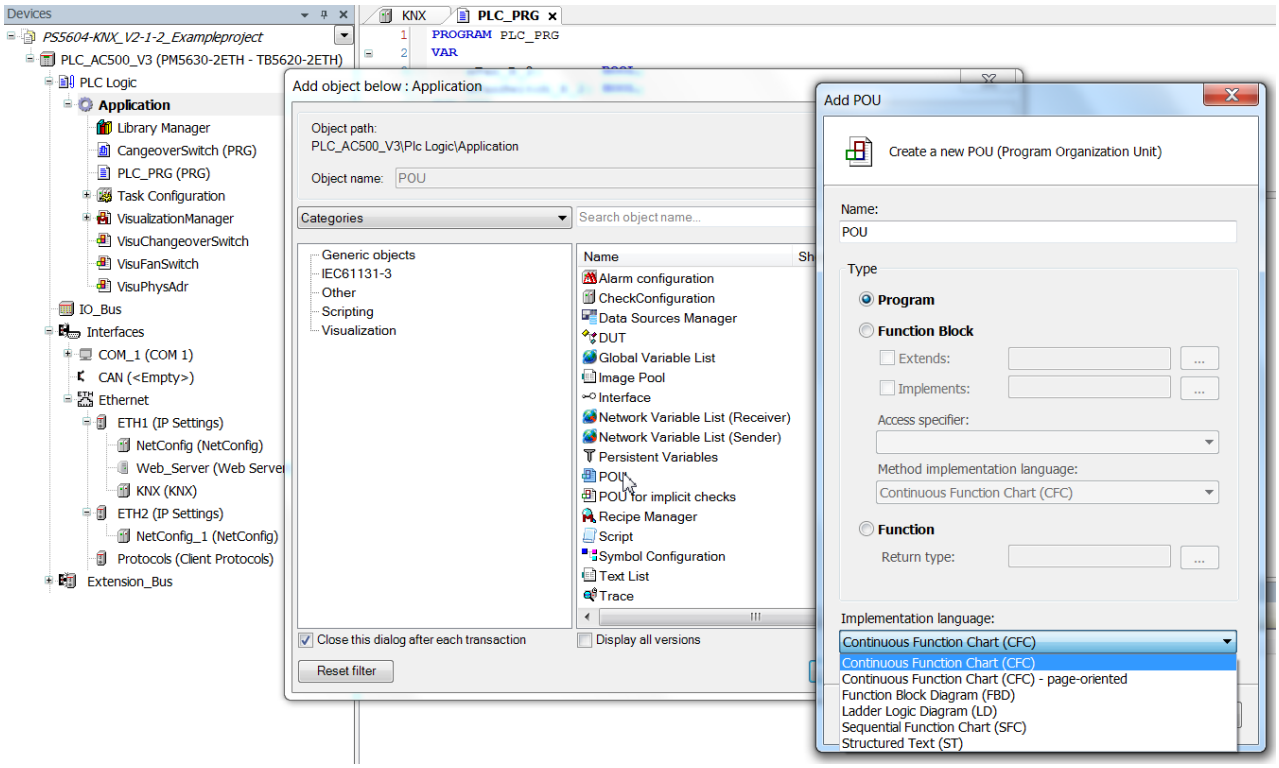


Figure 16: Add new programm

For example, also graphical programming languages are available such as Continuous Function Chart (CFC).

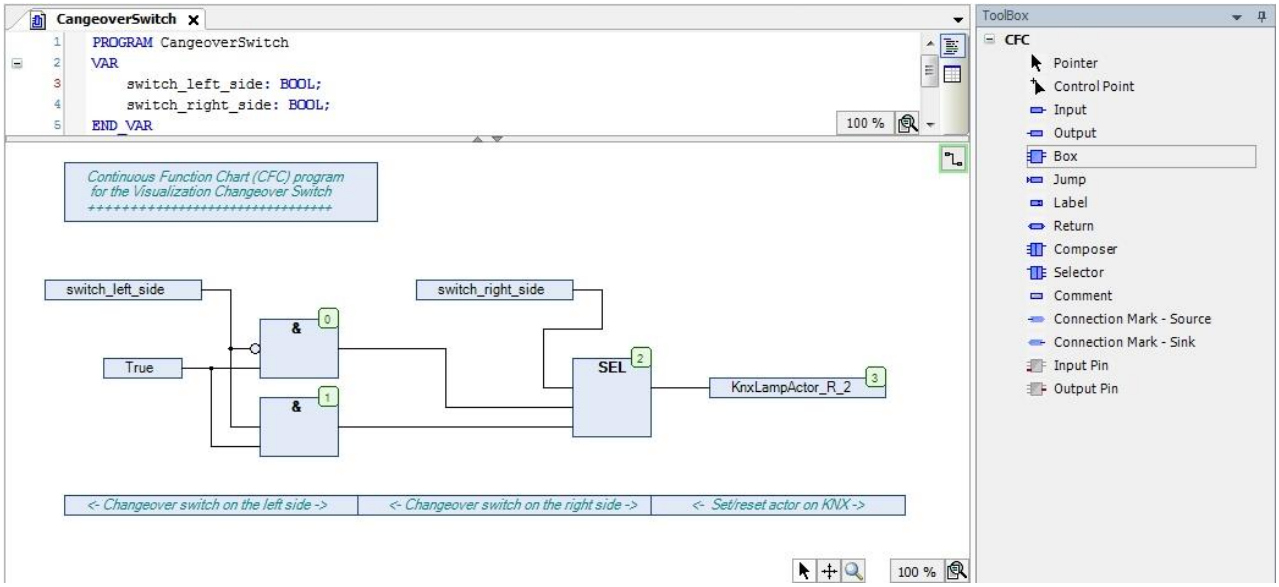


Figure 17: Sample program in a graphical programming language

To prepare the download of the configuration to the AC500 Controller, the program code must first be checked and compiled. You can find this function in the menu bar under **Build → Build**. Possibly errors and warnings are displayed in the message window at the bottom.

3.2.6 Download the Automation Program

Connect to the AC500 Controller via the function in the menu bar **Online → Login**. This requires an IP connection between the computer and the Controller.

The ABB Automation Builder recognizes that your Automation Builder project differs from the configuration on the controller and asks if you want to download your current project to the controller. Confirm this with **yes**. Depending on the extent of the change, this can optionally be done without stopping the automation program.

So that the Controller can use this configuration also after a restart it must be saved in the controller's permanent memory. To do this, execute the function **Online → Create boot application**. If this step is not performed, the last configuration saved by this function will be used when the controller is restarted.

Start the controller in run mode. Either via the menu bar **Debug → Start** or the **RUN** button on the controller.

While you are connected to the running controller, the ABB Automation Builder displays the cur-rent values of the variables and you can override them for testing purposes. Changes to the pro-gram code are not possible.

KNX communication is only working after also the matching ETS configuration was downloaded to the AC500 Controller. Until then, the Controller's KNX communication is deactivated and marked with a warning symbol.

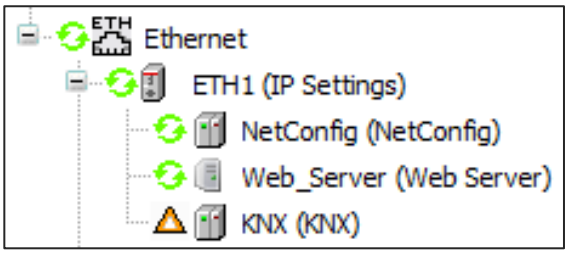


Figure 18: KNX interface not ready

However in this state the Controller can still be switched to the KNX programming mode and the physical KNX address can be programmed. Also the device info can be read by ETS.

You can use the **Online → Logout** function to end the connection to the Controller. The AC500 Controller remains in run mode.

3.3 Integration of the Controller in the KNX system

3.3.1 Insert Controller

Start the ETS and insert the *PS5604-KNX AC500 runtime license* as controller from the ETS device catalog into your ETS project. Give the controller a physical KNX address, which places the controller topologically on the IP main line or IP area.

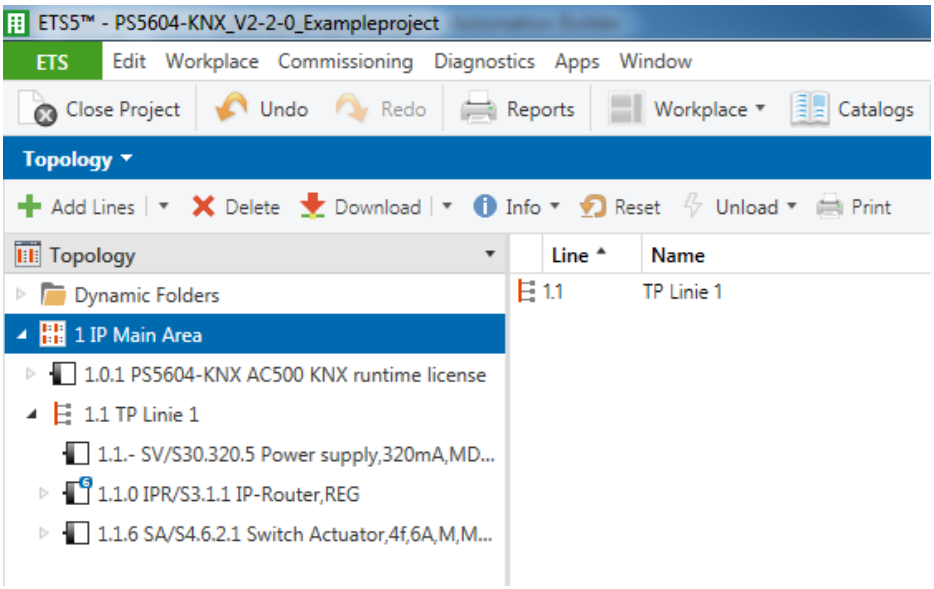


Figure 19: Topology in the ETS

3.3.2 Import configuration

Select the “PS5604-KNX AC500 runtime license” as device and then change to the DCA tab.

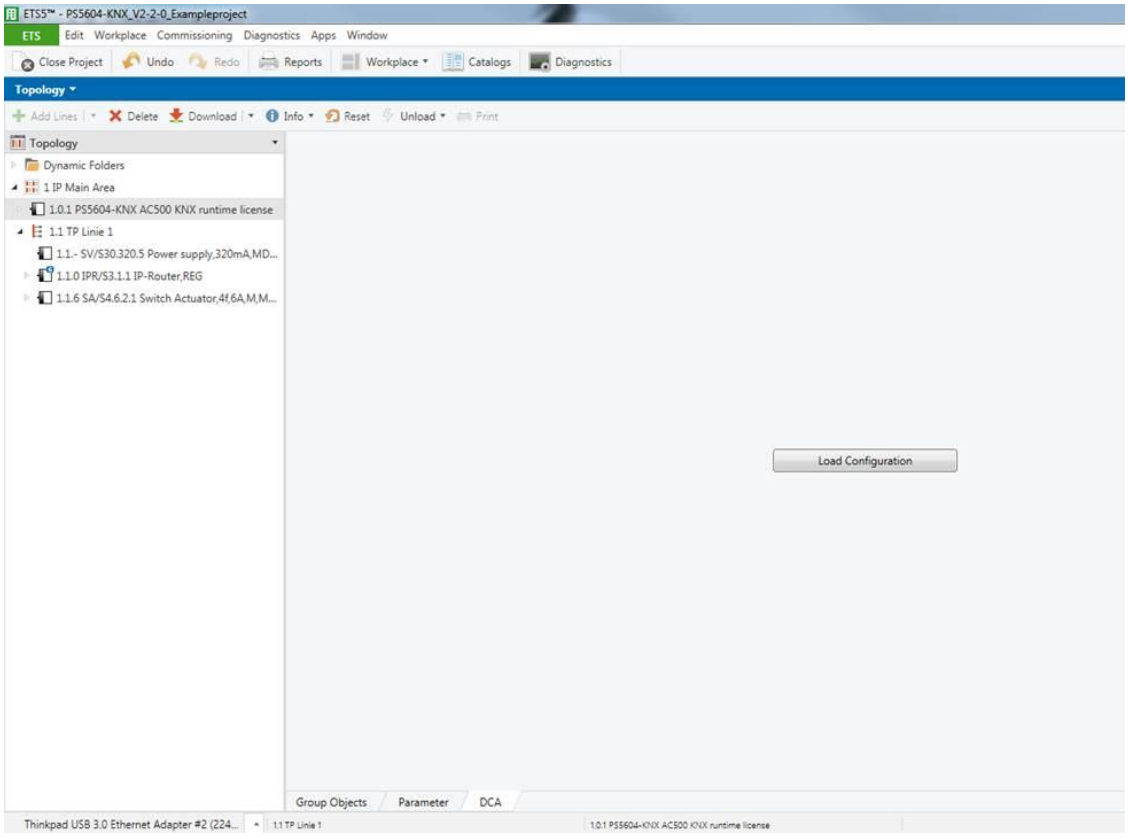


Figure 20: Open the DCA

Click on Load Configuration and select the configuration .xml file created in chapter 3.2.4.

The KNX Group Objects defined in the ABB Automation Builder of the “PS5604-KNX AC500 runtime license” as device are then displayed in the ETS.

Number	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
1	Room 2 Actor Lamp	Lamp actor	Aktor A Input	1/1/2	1 bit	C	R	-	T	-	1-bit	Low
2	Room 2 Lamp state	Lamp state	Aktor A Status	1/1/1	1 bit	C	-	W	T	U	1-bit	Low
3	Room 2 Actor Fan	Fan actor	Actor B Input	1/1/3	1 bit	C	R	-	T	-	1-bit	Low

Figure 21: KNX Group Objects in the ETS

3.3.3 Connect Controller with other KNX devices

You can now connect these group objects of the Controller with right-click Link with to a KNX Group Address.

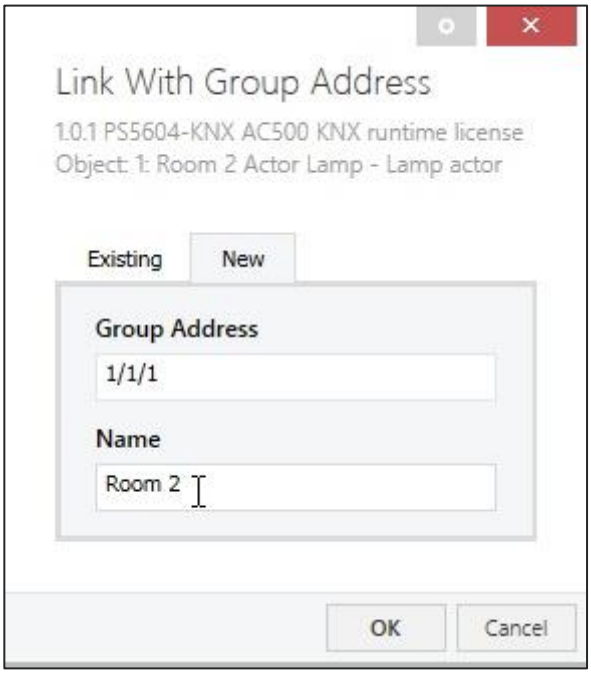


Figure 22: Create and link Group Address

This connected KNX Group Address must then be connected in the same way to the correspond-ing Group Object of the other KNX device.

3.3.4 Parameters of the device

The following settings are possible in the ETS parameters of the PS5604-KNX AC500 runtime license controller.

- **Default Gateway**: The used KNXnet/IP broadcast address. This must match the KNX system (KNX IP router). It is the default setting that is usually not changed.
- **Telegram rate**: The maximum transmission rate of the AC500 Controller can be limited in order to prevent an excessive bus load and thus to avoid malfunction of the KNX system.

  The KNX telegrams are buffered until they have been sent. New values which have been calculated by the automation program in the meantime are updating the cached values. The old cached value is discarded and not sent.
- **Project Information**: At this point, the project information of the ABB Automation Builder project is displayed.

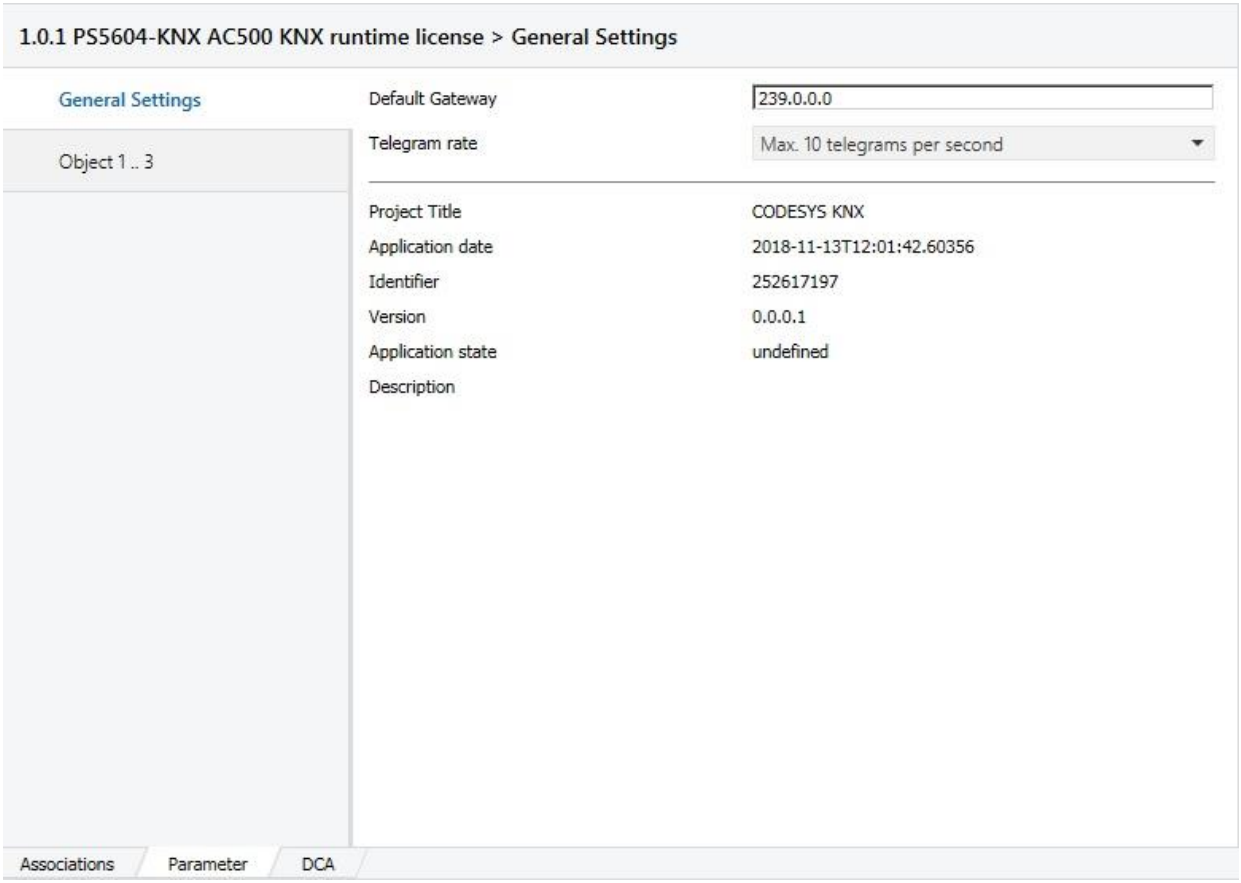


Figure 23: ETS Parameters of the Controller

For each KNX Group Object of the Controller an Object entry is displayed in the device parameters. This is named after the number of the KNX Group Object.

For outputs (Controller sends to the KNX bus) the KNX transmission conditions can be set:

- **Communication direction**: Setting of the transmission direction of the object
  - Input (KNX to PLC): The Controller receives values from the KNX bus
  - Output (PLC to KNX): The Controller sends values to the KNX bus
- **Send condition** (only for outputs): Setting whether the Controller sends a telegram to the KNX bus automatically when the object value is changed. The following options are available
  - **No automatic sending**: No automatic sending to the KNX bus. This must be done via the program code by the Trigger Output flag.
  - **Send on change**: Every time the object value changes, a telegram is sent to the KNX bus. No matter how minor this change is.
  - **Send on difference** (only for Group Objects which are not DPT 1.* Boolean) Every time the object value changes, this value is only sent to the KNX bus if it differs from the last sent value at least by the settable difference.
- **Sending difference** (only if Send on difference is active): Input of the difference by which the object value must change to be send. You can enter numbers with decimal places.
- **Cyclic sending** (only for outputs): Setting whether in addition the object value is sent cyclically repeatedly to the bus. This also happens if this object value has not changed. Two different value ranges for the cycle time can be specified.
- **Cycle time** (only when Cyclic sending is active): Specification of the cycle time for the cyclic transmission.
  Input format hour:minute:second

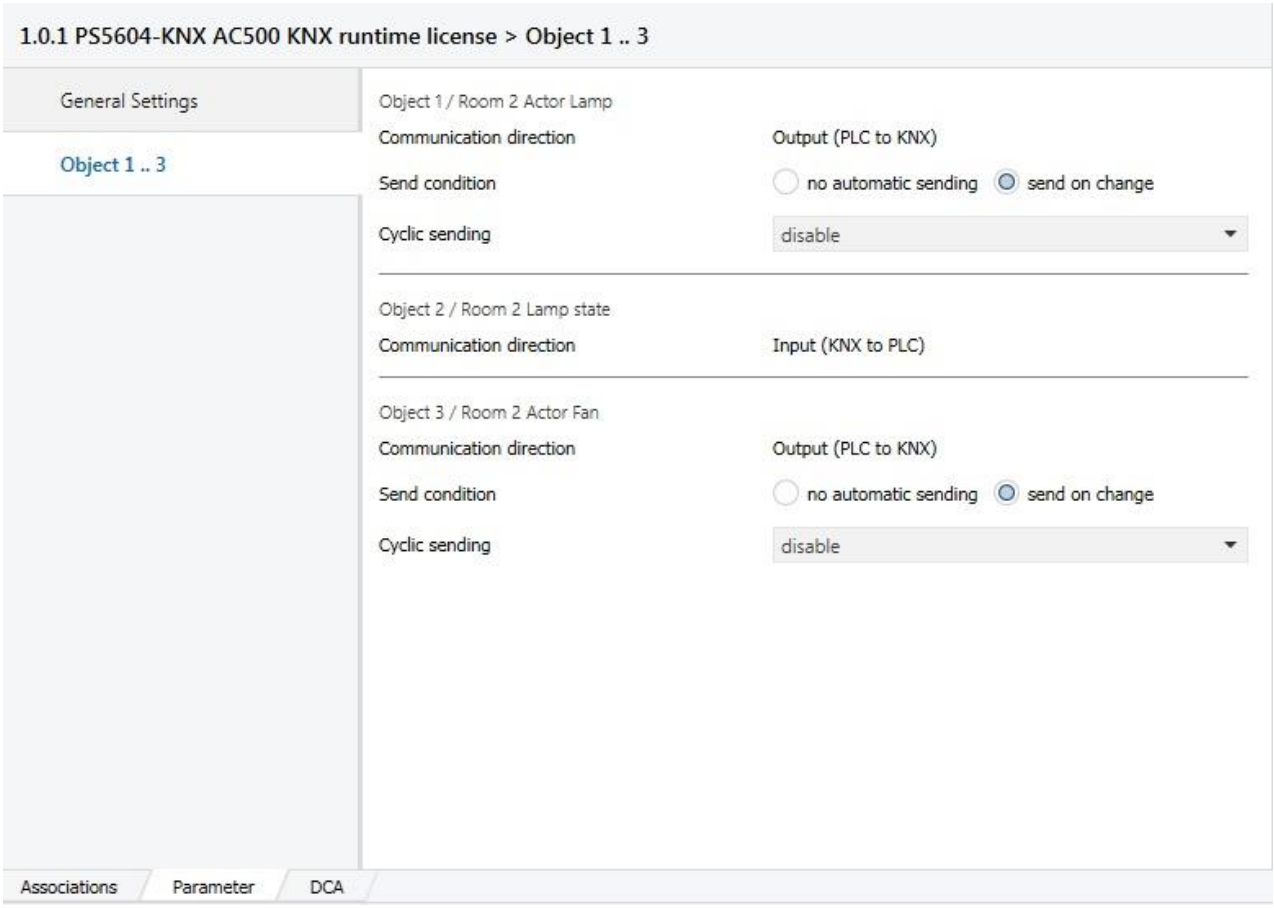


Figure 24: ETS parameters of KNX Group Addresses of the Controller

Regardless of the set transmission conditions, the program code can trigger by the flag **Trigger Output** a sending of the value to the KNX bus at any time.

By activating the function **Read on Init** of the KNX Group Objects in the right ETS properties panel, the Controller sends a value read query to the connected KNX device at startup. This then responds with its current object value.

In this properties panel you can also select the appropriate sub-data point type of the KNX Group Object. This defines the unit of the value in the KNX system. For example DPT 9.001 represents temperature in ° C.

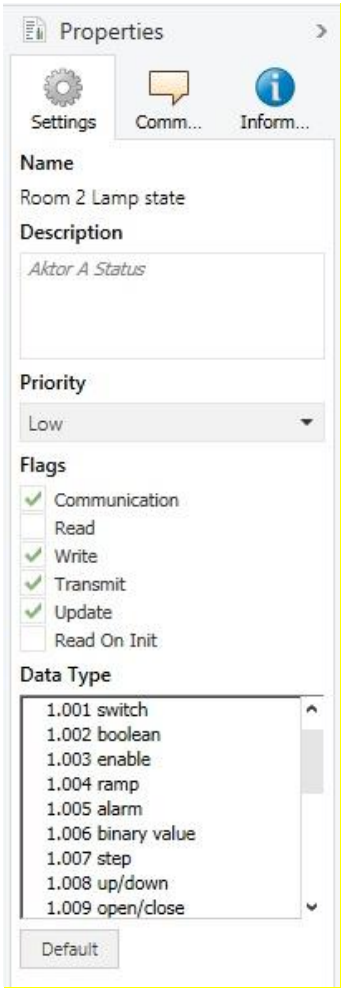


Figure 25a: Parameters of the KNX Group Objects

The example program *PS5604-KNX_V2-x-y_Exampleproject.project* needs the response of the actuator state for the input "Aktor A Status". This feature must be enabled in the parameter of the 1.1.6 SA/S4.6.2 Switch Actuator.

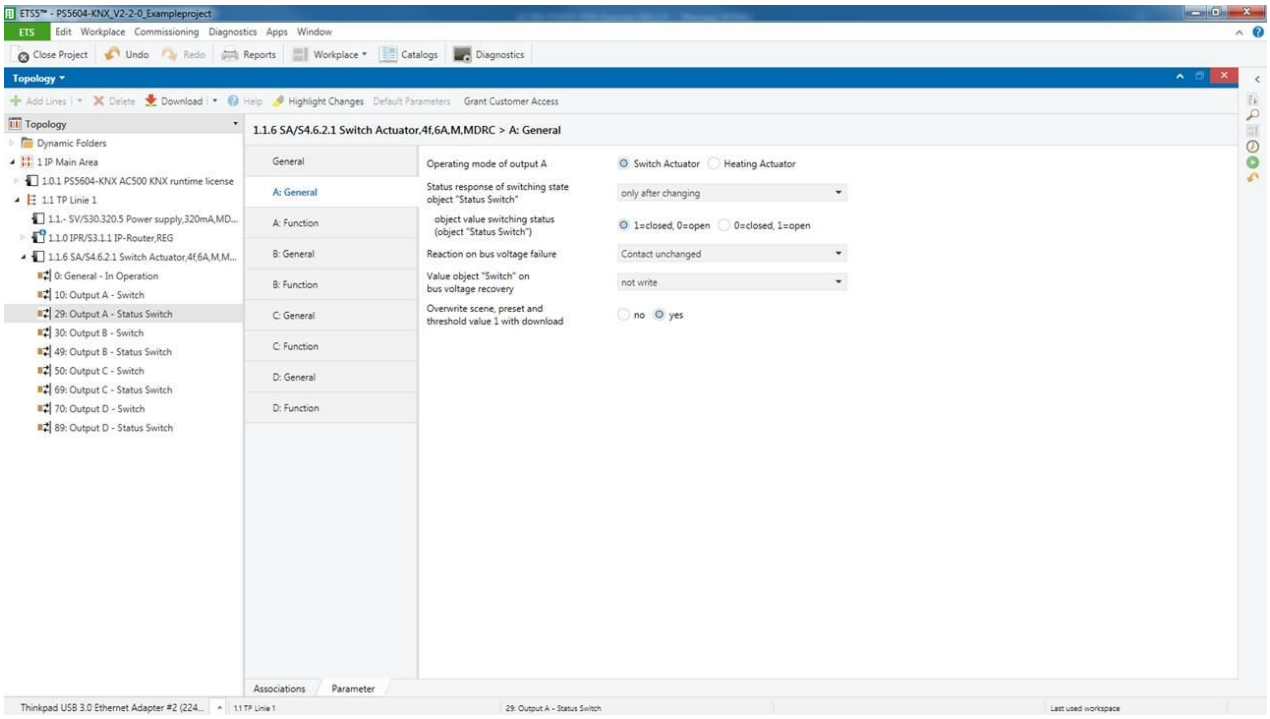


Figure 26: ETS parameters of KNX Group Addresses of the SA/S4.6.2 Switch Actuator

3.3.5 Download the ETS configuration to the controller

The download of the ETS configuration to the AC500Controller is done via the ETS function **Download** in the menu bar. This download happens via the KNX interface directly to the AC500 CPU.

Best you select in ETS the network interface of the computer as the bus interface. Thus, a fast data exchange is possible and the data is not routed via the KNX TP bus.

At the first download, the physical KNX address of the Controller is programmed. To do this, set the AC500Controller to KNX programming mode. This can be done either via the display or the web interface of the controller.

Display

Attention: The activation of the KNX programming mode via the display only works with Automation Builder Version 2.2.0 or newer only.

1. Der Controller must be in **Run** mode.
2. Press the **CFG** button.
3. Press the button **Down Arrow** button, so that **Pbut 0** is displayed.
4. Press the CFG button. The display shows **Pbut 1** flashing.
5. Confirm this with the **OK** button. The display permanently shows **Pbut 1**. The Controller is in KNX programming mode.
6. The Controller automatically terminates the KNX programming mode after the programming of the physical KNX address. Alternatively you can terminate the programming mode with Pbut 0 by pressing the CFG button.

You can exit the menu at any time with the **ESC** button.

Web interface

The AC500 Controller must be in **Run** mode. Open the web page of the controller by entering the IP address in a browser. In the factory settings it is http://192.168.0.10

The KNX programming mode can be activated and deactivated via the button. The red signal lamp indicates the status of the KNX programming mode.

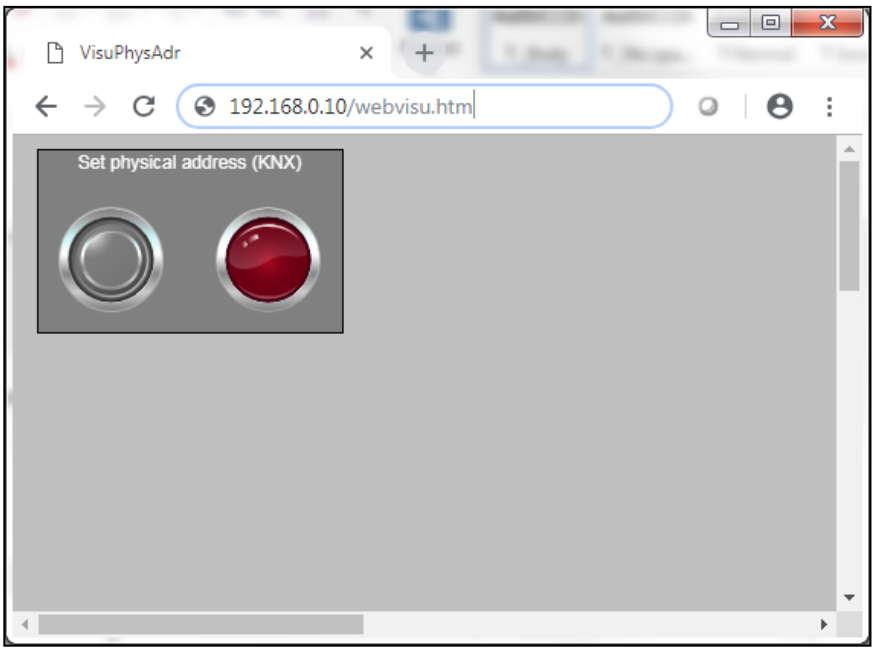


Figure 27: Webpage of the controller

The controller automatically terminates the programming mode after programming the physical address with the ETS.

The AC500 Controller has then besides the Automation Builder configuration also the appropriate ETS configuration and starts its KNX communication. This can be recognized by the green symbol on the KNX interface in the Automation Builder.

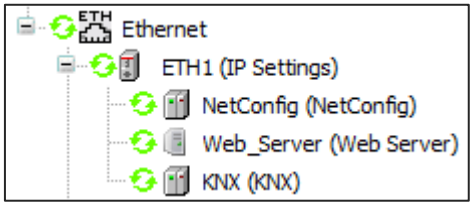


Figure 28: KNX Interface ready

Then download all other linked KNX devices as well as the KNX IP routers. The ETS automatically creates the filter tables of the KNX IP routers so that the KNX telegrams are routed from the KNX TP lines to the IP line of the AC500 Controller.

3.4 Make changes

Changes can be made in the ABB Automation Builder as well as in the ETS without the need for a change in the other software or the need for a new data exchange.

Only if changes are made to the KNX Group Objects in the ABB Automation Builder, a data ex-change with the ETS is again necessary. Afterwards, a download is required both in the Automation Builder and in the ETS. Only when these two configurations have been downloaded again to the AC500Controller, the KNX communication is in operation again.

The DCA detects changes to names and numbers of the KNX Group Objects when importing the configuration file in the ETS and keeps the already made settings and linked Group Addresses of these changed Group Objects.

3.5 Advanced Hints and Options

3.5.1 KNX runtime license

The standard V3 AC500 Controllers are shipped from the factory without FW and need an installed PS5604-KNX runtime license for KNX operation. The license document of this installed license needs to be purchased separately and can be transferred from one to another controller via Automation Builder.

3.5.2 Data Conversion

The KNX standard defines a big-endian byte order while the IEC 61131-3 is based on the little-endian byte order. Therefore, the controller automatically converts the data point types.

However, if you access the bits of the structured KNX data point types (DPT) for time, date and color (DPT 10.* , DPT 16.* , DPT 19.* and DPT 232.*) in your program code, you have to note the re-verse byte order. Therefore, as of ABB Automation Builder version 2.2.0, corresponding function libraries are available that provide conversion functions for these data point types.

3.5.3 Battery backup

The AC500Controller can be buffered via an optional battery, so that the data variables and the device clock are available even after a power failure. For this has the Controller has a battery compartment.

The variables must be defined as globally persistent retain variables. The controller then stores it in the battery's permanent memory which is buffered by the battery.

Alternatively, the program code may also manually store values in the flash memory.

For more information about these features, see the ABB Automation Builder Help.

3.5.4 Data storage

The memory of the AC500Controller can be extended by a memory card. The configuration of the Controller can also store on this memory card which made an easy device replacement possible.

For more information about these features, see the ABB Automation Builder Help.

3.5.5 Device clock

The AC500Controller has a built-in real-time clock. This can be set using the ABB Automation Builder software with the Statistics function or the program code.

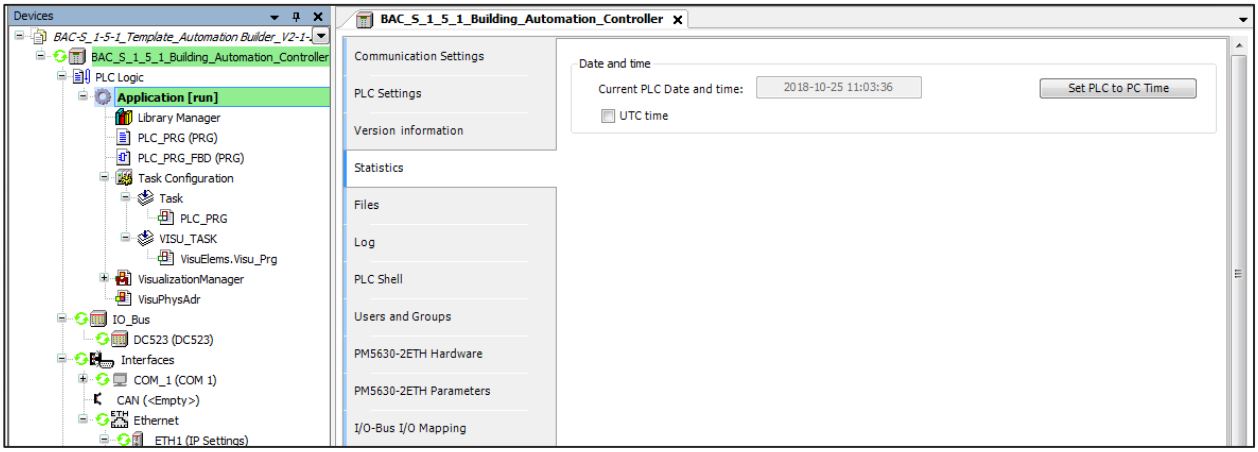


Figure 29: Set the clock of the Controller

At a power failure the clock is buffered via the optionally available battery.

For more information about these features, see the ABB Automation Builder Help.

3.5.6 Web server with web user interface

The AC500Controller has a web server. The web pages can be freely designed in the ABB Automation Builder. For more information, see the ABB Automation Builder Help.

3.5.7 Further communication protocols

In addition to KNX, the AC500Controller has additional communication protocols, such as: Modbus over TCP and RS-485 as well as CAN and OPCA UA. For more information, see the ABB Automation Builder Help.

3.6 Maintenance and fault detection

3.6.1 Firmware update

The software of the AC500Controller is updated via the Version information function in the ABB Automation Builder.

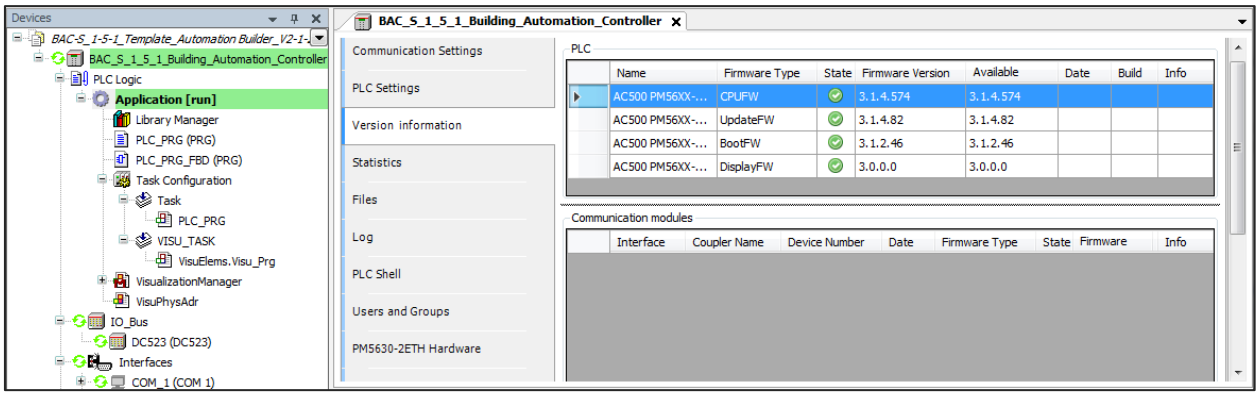


Figure 30: Run firmware update

3.6.2 Diagnostic

After you are logged in to the AC500Controller you can read the device protocol by the function Log in Automation Builder.

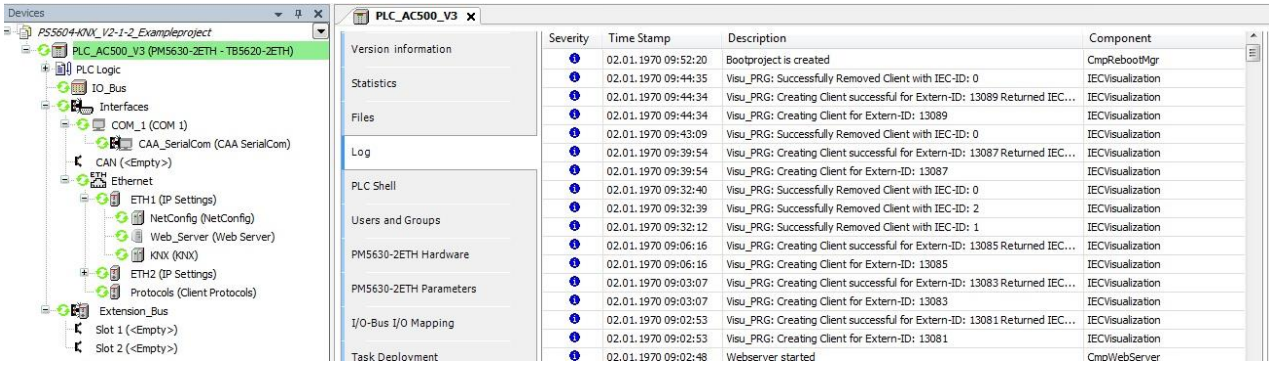


Figure 31: Protocol of the Controller

The current IP address as well as further information of the AC500Con-troller can be read via the ETS **Device Info** function. For this the physical KNX address is necessary. You can determine the address by the ETS function **Programming Mode**.

With the ETS function **Group Monitor** you can analyze the telegrams on the KNX bus. You can also use it to send KNX telegrams.

4. FURTHER INFORMATION

4.1 Support and training

4.1.1 Automation Builder and AC500 PLC

For more information, see the Automation Builder Help. You can open it by clicking the Help button. This help and as well as other documents about the products can be found at www.abb.com/plc

On this page you will also find information about the trainings.

Training videos can be found at www.youtube.com/user/abbplc

4.1.2 KNX and ETS

More information about the products, support and training can be found at www.abb.com/knx

The KNX Association offers with the **ETS eCampus** a free online training course about the basics of KNX and the ETS. For more information, visit www.knx.org at MyKNX.

4.2 Software libraries

The ABB Automation Builder is based on the established Software Codesys and can thus easily be extended by third-party IEC61131-3 software libraries, for example

Oscat Library www.oscat.de

Attention: ABB does not warrant and accept any liability for these third-party software libraries.

4.3 Known Issues

Issue Name	Issue Description	Workaround
Default value at Start-up for first KNX Object.	After Reboot/Restart of the PM56xx-2ETH the first configured KNX group object, connected with a group address, is sent to the KNX-Bus with its default value.	To avoid unplanned reactions at the KNX network, the first configured KNX group object has to be a dummy object and must be connected to an unused group address in the KNX network.

REVISION HISTORY

Rev.	Page	Change Description	Date / Initial
-r2	all	First Release	2018-11-16 IAMF
-r3	27	Chapter 4.3 Kown Issues added, adaptation f. AB2.2.0	2018-12-20 IAMF
-r4	Several	Adaptation to newer AB versions	2019-03-10
-r5	27	Available codesys libraries adapted	2019-04-25

ABB Automation Products GmbH
Wallstadter Str. 59
68526 Ladenburg, Germany
Phone: +49 62 21 701 1444
Fax : +49 62 21 701 1382
E-Mail: plc.sales@de.abb.com

www.abb.com/plc

Document Number 3ADR010265_5_en_US