

APPLICATION EXAMPLE

AC500 HOW TO USE OPC SERVER V3 FOR DA AND UA



Contents

1 Disclaimer	4
2 Introduction	5
2.1 Scope of the document	5
2.2 Compatibility	5
2.3 Overview	5
3 How to get and install the OPC Server	6
3.1 OPC Server V3 and CODESYS Gateway V3/V2	7
3.2 Manual Registration and Unregistration	11
3.2.1 Parallel installation and registration.....	11
3.2.2 Registration as service	11
3.2.3 Uninstalling	12
3.3 Hardware and Software Version requirement.....	12
3.4 Overview and connections	13
4 Configuration of the symbols in the programming system	14
5 Configuration of the OPC server v3	17
5.1 Open a particular INI file	17
5.2 Settings for OPC server	18
5.3 Settings for <PLC>	19
5.4 Settings for connection to <PLC>	20
6 Check OPC function with AC500	23
6.1 Check OPC Server V3 without AC500.....	23
6.2 Check OPC Server V3 with AC500	25
7 OPC UA Client for test	27
8 Connection AC500 V2 and OPC UA Client	31
9 Behavior of the OPC UA with CP600	31
9.1 Protocol Configuration in Automation Builder	32
9.1.1 Create new project.....	32
9.1.2 Add OPC UA Server	34
9.1.3 Task Configuration	35
9.1.4 Global variables	36
9.1.5 PLC Logic.....	37
9.1.6 Symbol Configuration	37
9.1.7 Online to the PLC.....	41
9.2 Panel Builder 600	42
9.3 Establishing an Encrypted Connection of CP600 OPCUA Client to an OPC UA Server AC500 V3.....	51
9.3.1 Creating a certificate for the OPC UA Server AC500 V3	51
9.3.2 Setting up an encrypted connection with the CP600 OPC UA Client	54
9.3.3 Setting up an encrypted connection with the CP600 HMI simulator	63
10 Connection AC500 V3 High Availability and OPC DA Client	66
10.1 Requirements.....	66
10.2 Configure and download symbol file	66
10.3 OPC Config.....	68
10.4 Check OPC Server with MatrikonOPCEXplorer.....	71
11 Connection AC500 V3 High Availability and OPC UA Client	73

11.1	Configure and download symbol file	73
11.2	Check OPC Server with ABB Panel CP600	75

1 Disclaimer

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 these 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."

2 Introduction

2.1 Scope of the document

This document refers to an example program named 'V3_OPC_Example_V2.5.project' on how to configure the OPC server for V3 CPU, including OPC DA and UA.

2.2 Compatibility

The application example explained in this document have been used with the below engineering system versions. They should also work with other versions, however some small adaptations may be necessary, for future versions.

- AC500 V3 PLC
- Automation Builder 2.2.0 or newer

2.3 Overview

OPC Unified Architecture (UA) is an open standard created by the OPC Foundation with help from dozens of member organizations. Although UA intends to provide a platform independent interoperability standard (in-order to move away from Microsoft COM), it is not a replacement for OPC Data Access (DA) technologies. For most industrial applications, UA will complement or enhance an existing DA architecture. It will not be a system-wide replacement. OPC UA complements OPC DA infrastructures in the following ways:

- It offers a secure method of client-to-server connectivity without depending on Microsoft DCOM and has the ability to connect securely through firewalls and over VPN connections. For users connecting to remote computers within the corporate network (inside the firewall) on a domain, an OPC DA and DCOM connection may be satisfactory.
- It provides an additional way to share factory floor data to business systems (shop-floor to top-floor). OPC UA can aggregate data from multiple OPC DA sources into non-industrial systems.

For the majority of user applications, the most relevant components of the UA standard are as follows:

- Secure connections through trusted certificates for client and server endpoints.
- Robust item subscription model to provide efficient data updates between clients and servers
- An enhanced method of discovering available information from participating UA servers.

**NOTICE**

For more information on OPC UA, please search the key words 'OPC UA' in search engine website.

3 How to get and install the OPC Server

This section describes how to install the OPC Server OPC V3 and Gateway V3.

Before you execute the installation, you must close all OPC Client, the ABB OPC Tunnel and the Gateway (CODESYS gateway server) on your PC. Please check these with the Windows Task Manager.

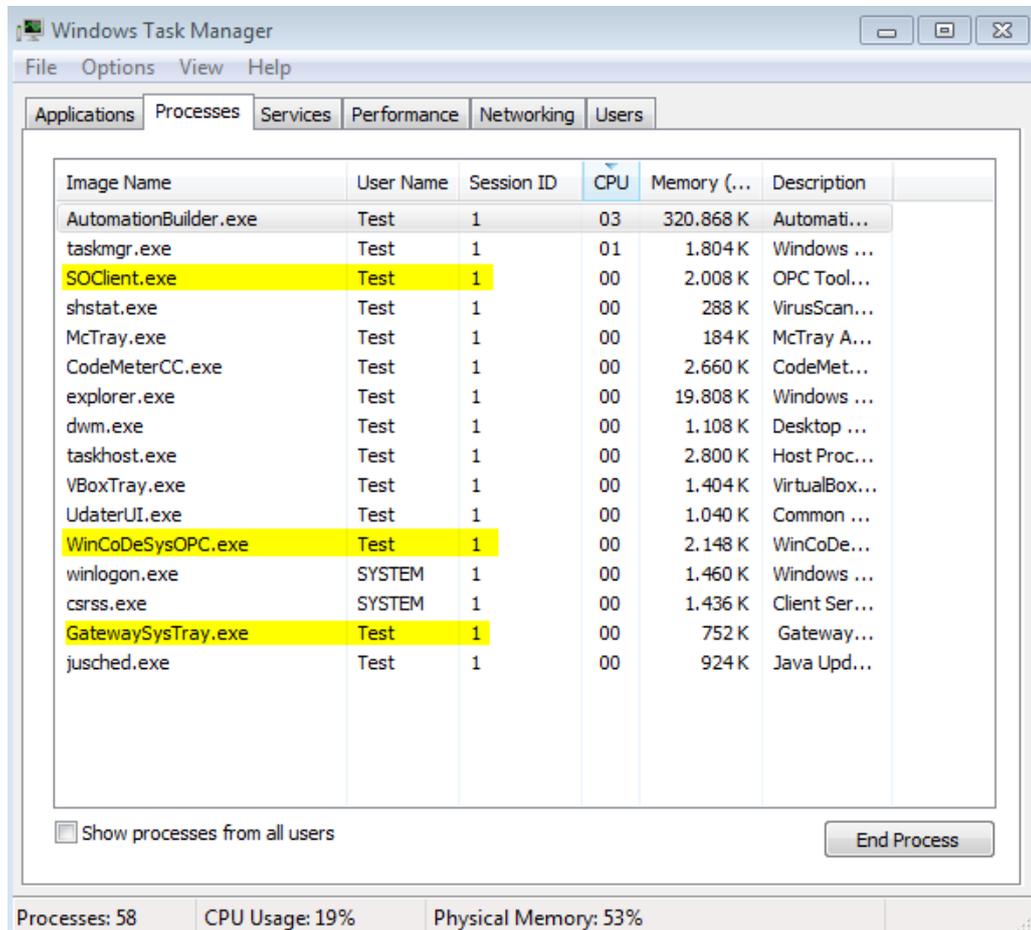


Figure 1: Windows Task Manager

The Processed of

- Gateway.exe
- CoDeSysOPC.exe
- WinCoDeSysOPC.exe
- OCTsvc.exe
- Must have disappeared.

If not:

- End the processes with the Windows Task-Manager.
- Stop the ABB OPC Tunnel Windows Component Service, Services (local).

3.1 OPC Server V3 and CODESYS Gateway V3/V2

Install the OPC Server V3 directly from the Additional Tools in Automation Builder. You can download the installation package from website www.abb.com/automationbuilder or <https://new.abb.com/plc/automationbuilder/platform/software>

Automation Builder software download

Automation Builder is available in Basic, Standard and Premium editions meeting the needs of small projects and managing the challenges of many and large projects for OEM and system integrators.

Start working immediately: After installation, on the first start-up of Automation Builder you can choose from different licenses:

- Free 30-day trial license – unlocking standard and premium features
- Free Basic edition

Purchased standard or premium license

- Licenses can be activated, removed and transferred anytime
- Availability of network licenses for installation on a license server

Life-cycle support: When installing Automation Builder you can include former version profiles into your installation to maintain compatibility with projects done in former versions of Automation Builder. Alternatively you will find installation files for selected former versions in the ABB Library below.

Programmable Drive support is discontinued with Automation Builder 2.2. But Automation Builder can still be used to access previous versions of drive IEC 61131 programming. The successor engineering tool for ACS880, DC5880 and DCT880 programming will be the Drive Application Builder. Planned date of first release is end of March 2019.

All tools that are required to configure and commission a drive as a field device connected to the AC500 PLC will remain integrated part of Automation Builder, mainly:

- Drive Manager
- Drive composer pro

Note: If you want to upgrade projects with third party safety devices we recommend that you create project archives of your projects BEFORE installing Automation Builder 2.2 (File -> Project Archive -> Save/Send Archive ...).

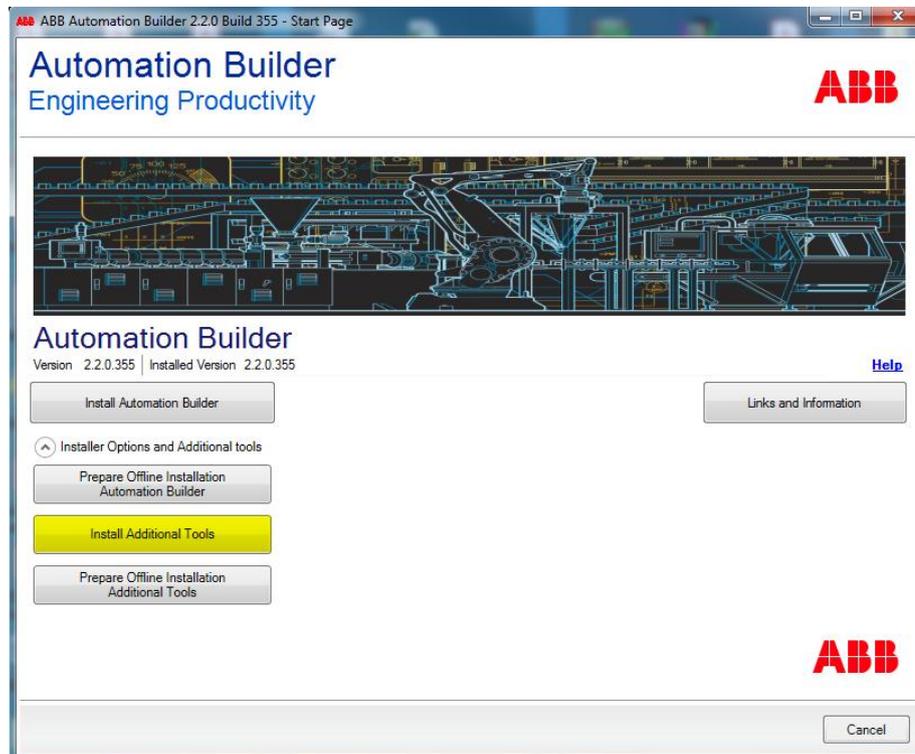


Are you looking for support or purchase information?

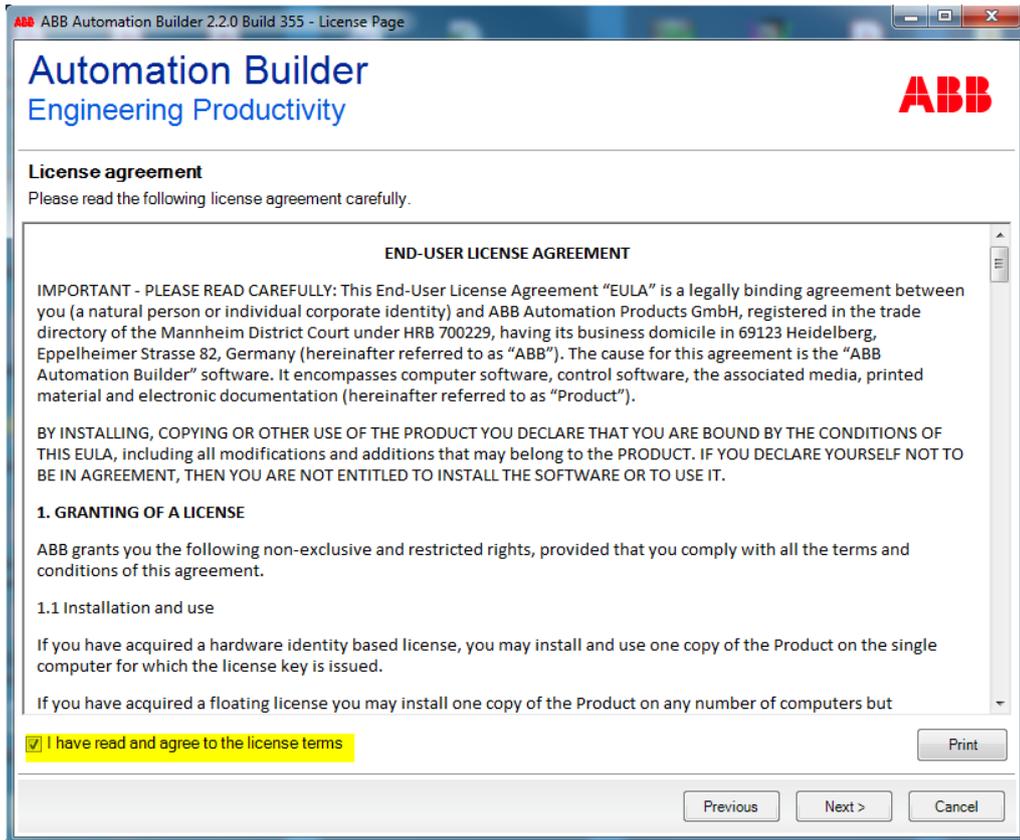
↓ Contact us

OPC Server V3 is delivered in the Ssetup from Automation Builder installer. After downloading the package, Launch the ABB Automation Builder installation wizard.

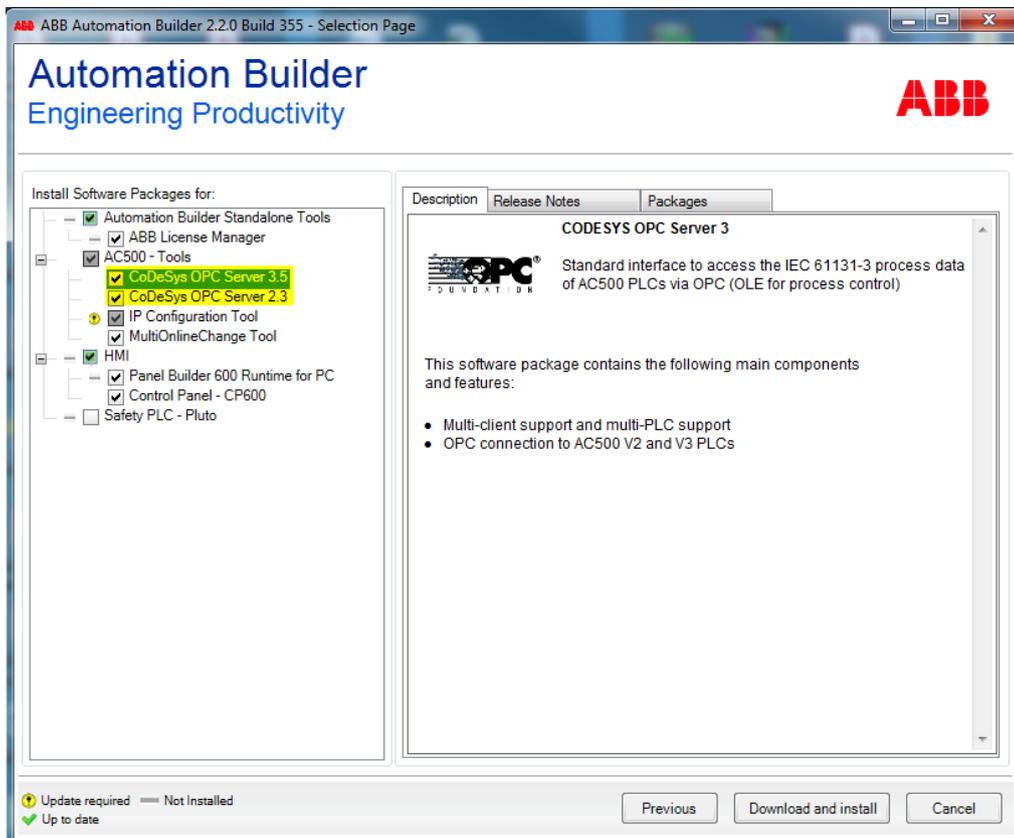
Select Install Addition Tools.



Agree to the License terms and click 'Next' button.

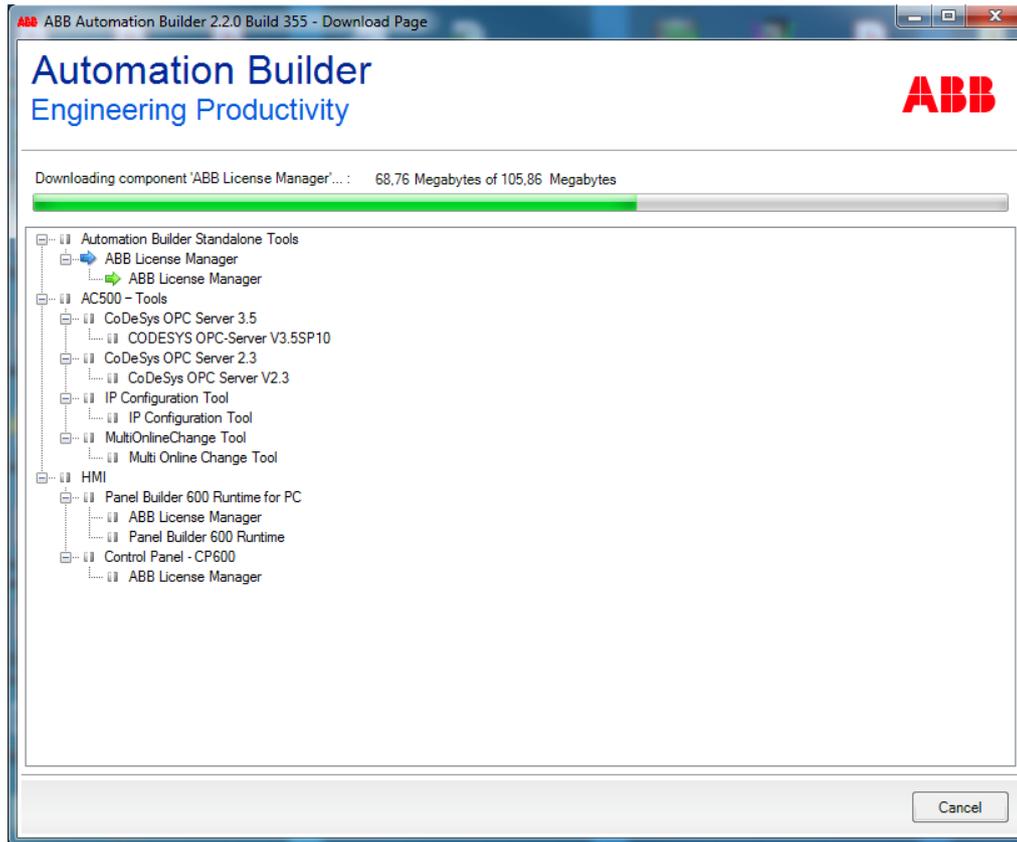


Choose CoDeSys OPC Server 3.5, with this the OPC Server V3 is installed and registered.

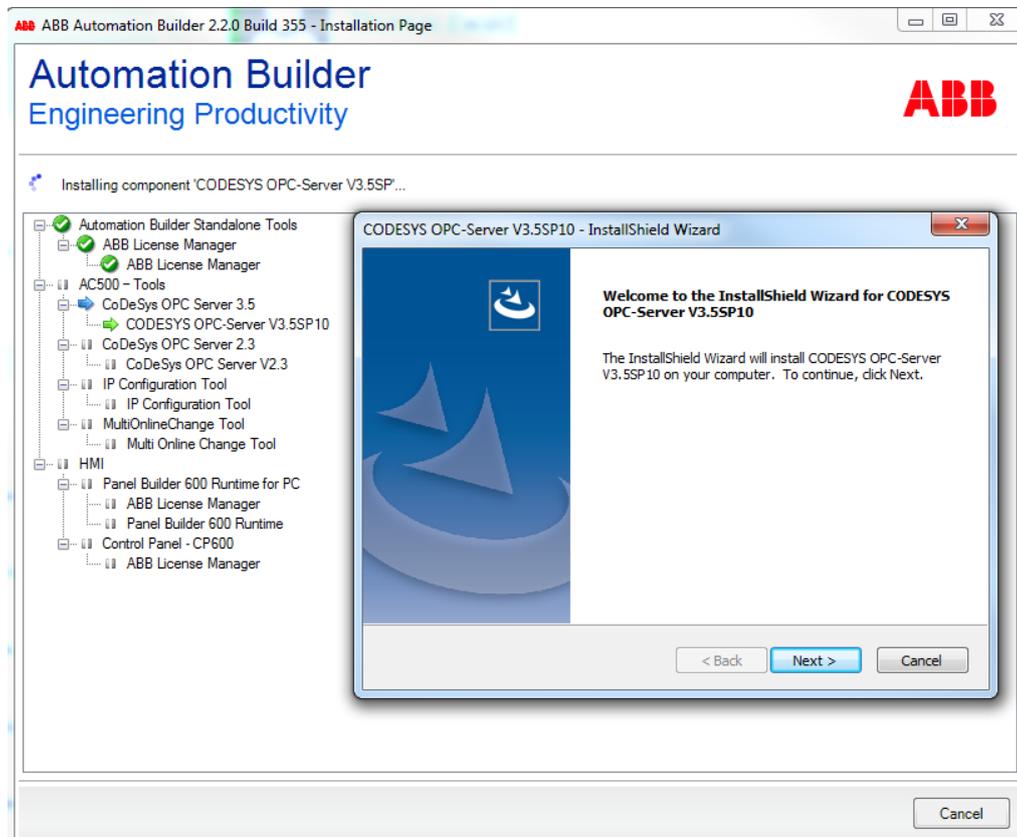


CoDeSys OPC Server 3.5 and CoDeSys OPC Server 2.3 can be installed in parallel.

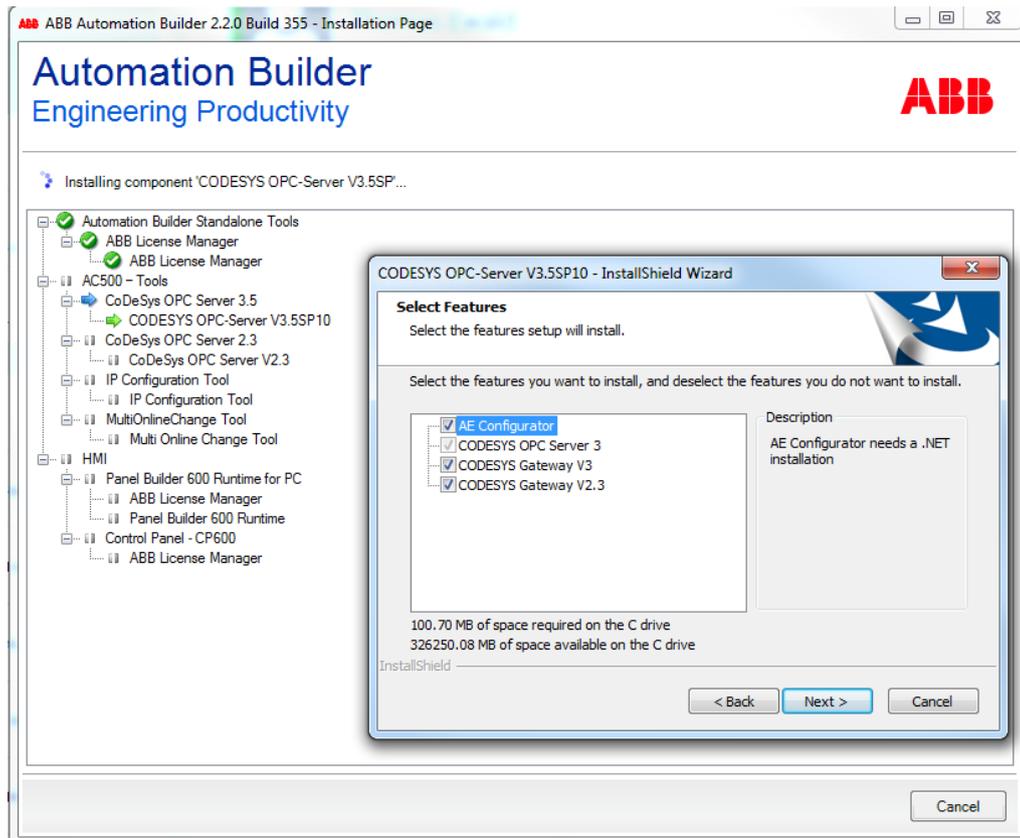
Click 'Download and install'



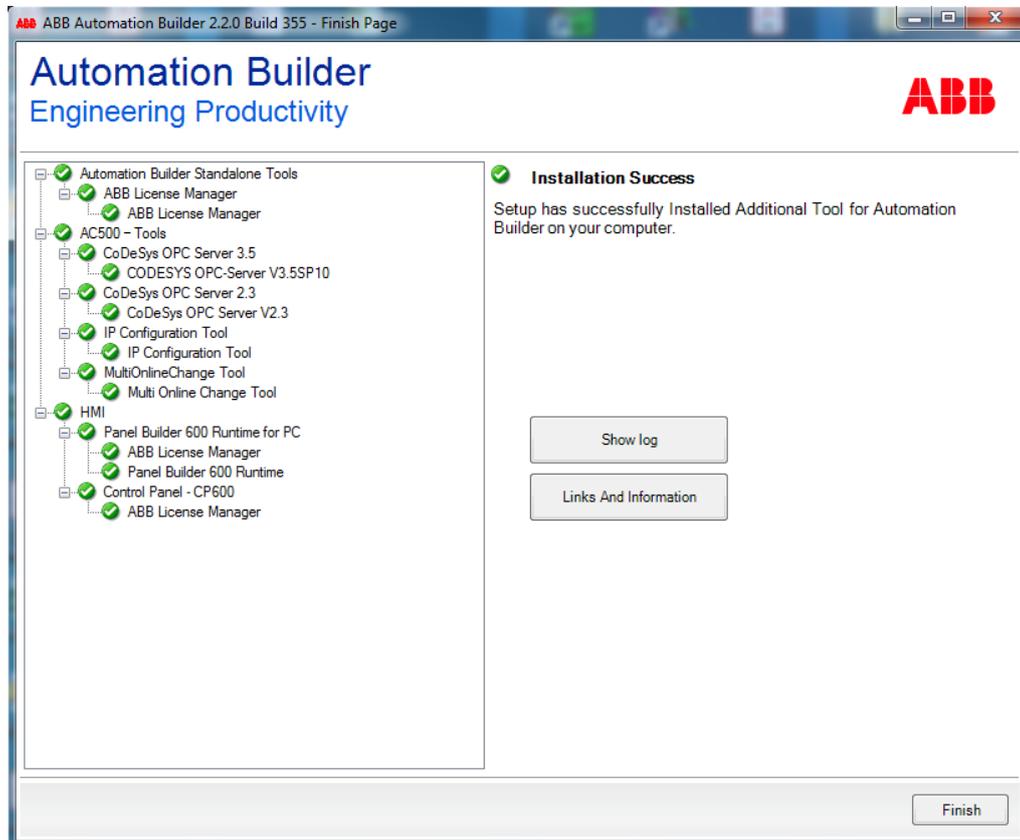
Choose the CoDeSys Gateway 3.5. And deselect the features you do not want to install.



The OPC Server is able to communicate with both, AC500 V3 and AC500 V2 PLCs, at the same time.



Click "Finish" to complete the installation.





For using the OPC Server V3 on a Windows Vista system, it's strongly recommended to run the OPC client in Windows XP SP2 compatibility mode and with extended administrator rights in order to get a continuous updating of the values.

You can install the gateway about the installation of the server OPC V2/V3 (see above). Servers OPC V2 and V3 can be installed in parallel.

It is not necessary to install CODESYS OPC Server V3/V2 if you want to communicate PLC with only OPC UA.

3.2 Manual Registration and Unregistration

During the installation with Automation Builder 2.2.0, all needed files are installed for OPC and the OPC Server is registered automatically as user application.

Further on, there is the possibility to register resp. to uninstall the OPC Server manually either as COM Server (user application) or as service.

3.2.1 Parallel installation and registration

With command

```
WinCODESYSOPC /RegServer
```

the server is registered as COM server. Thereby as location path always the current position of WinCODESYSOPC.exe will be used. Thus, the call only may be done from a local path.

3.2.2 Registration as service

With command

```
WinCODESYSOPC /Service
```

WinCODESYSOPC.exe gets installed as system service. Started once, the service will stay "started" until the system gets terminated. The communication to the configured PLCs survives.

The service is also installed here at the current position of WinCODESYSOPC.exe.

3.2.3 Uninstalling

With command

WinCODESYSOPC /UnRegServer

All entries of the OPC Server will be removed from the registry. The installed files will not be removed.



TIP

Register the OPC server in the registry as interactive software with command:

- For OPC 3: WinCoDeSysOPC/RegServer
- For OPC 2: CoDeSysOPC/RegServer

Register the OPC server as system service with command:

- For OPC 3: WinCoDeSysOPC/Service

Unregister the OPC server from registry and from service entry with command:

- For OPC 3: WinCoDeSysOPC/UnRegServer
- For OPC 2: CoDeSysOPC/UnRegServer



NOTICE

Don't forget to reboot your PC after Registration or Uninstalling.

3.3 Hardware and Software Version requirement



NOTICE

The following configuration and version are used in this example. Newer versions of hardware and software should work, however if not mentioned here, it should be tested by the user himself.

Hardware	Software
AC500 CPU: PM5650-2ETH, FW: V3.2.0	Automation Builder 2.2.0 Build 355
PC	Windows 10 (32/64 Bit) Professional / Enterprise
CP6607	Panel Builder 600 4.0.1 build 462

3.4 Overview and connections

The overview of the system is explained in the image below where it is connected to a SCADA system (Clients) and devices PLCs (or DCS, remote IOs..).

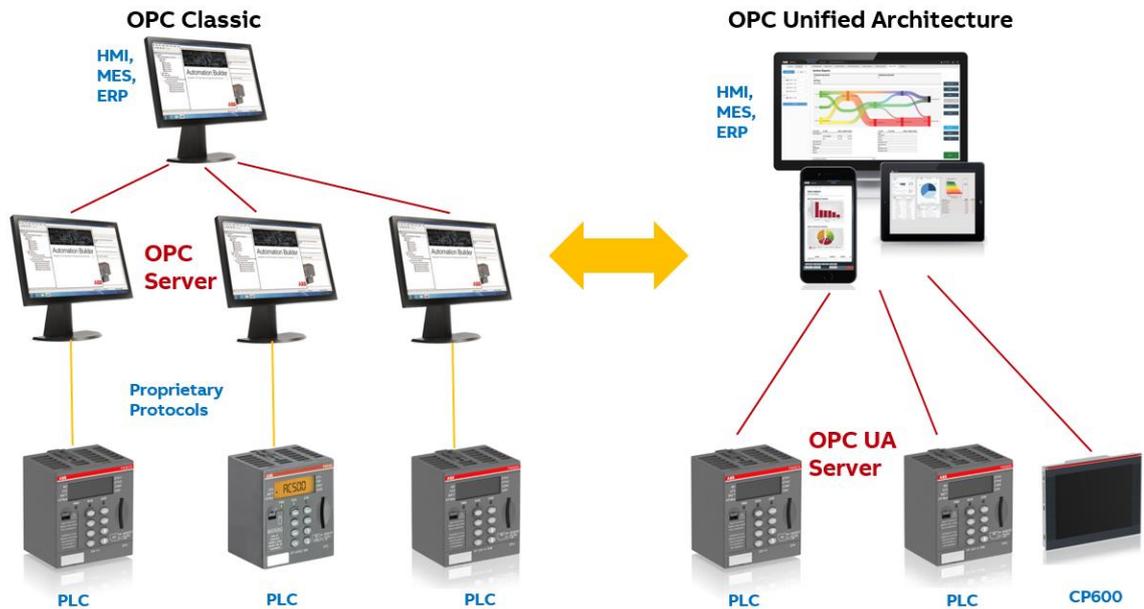


Figure 2: Simple hardware connection example

We've seen some of the effort involved in making an HMI tag and historian database

- Build a tag data base
- Get the devices to talk to HMI software
- Configure an historian to log the data

The one in the red circle involves current OPC DA

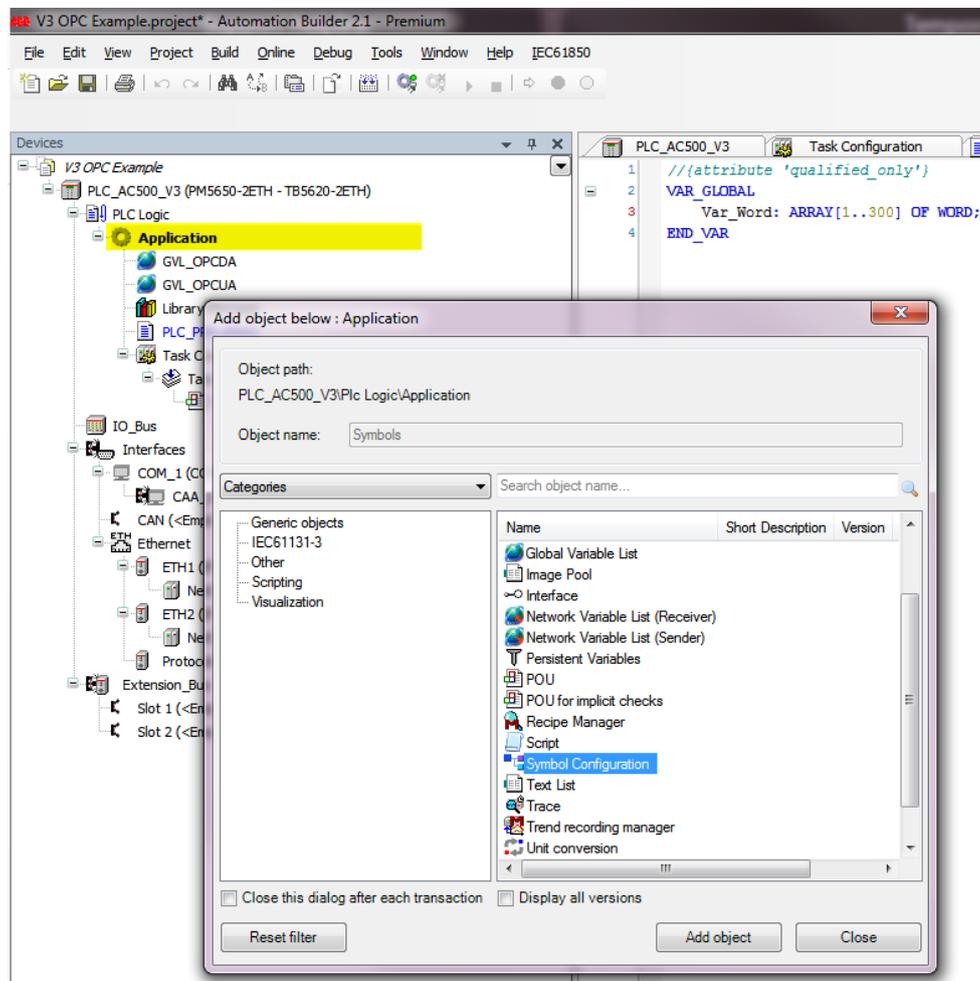
The one in the blue circle involves OPC UA

4 Configuration of the symbols in the programming system

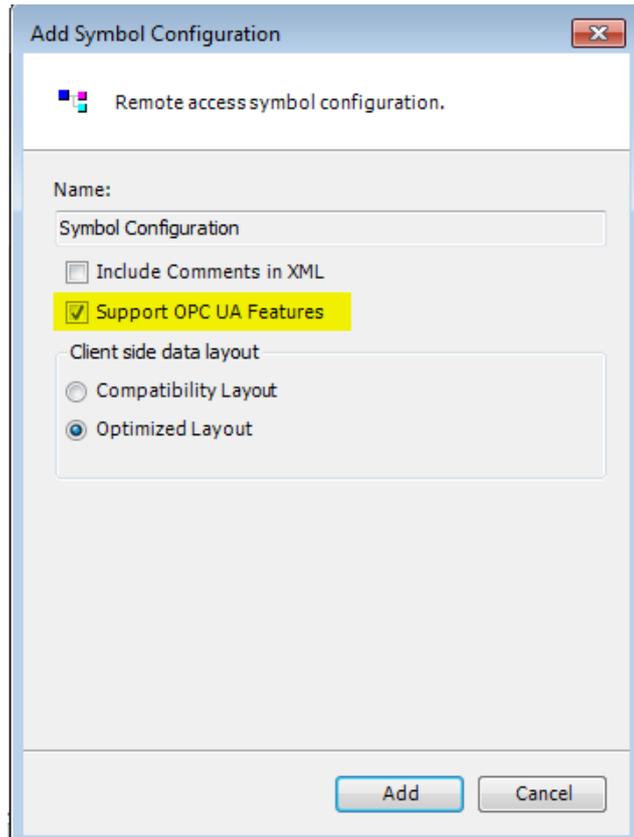
This chapter shows you how to configure the symbols for OPC communication in V3 CPU (for example, AC500 PM5650).

Processor Module	Program Memory	Network Interface	Other Interfaces	Suitable Terminal Base
PM5630-2ETH	8 MB	2 x onboard Ethernet	CAN and COM1	TB56xx-2ETH
PM5650-2ETH	80 MB	2 x onboard Ethernet	CAN and COM1	TB56xx-2ETH
PM5670-2ETH	160 MB	2 x onboard Ethernet	CAN and COM1	TB56xx-2ETH
PM5675-2ETH	160 MB, 8 GB flashdisc	2 x onboard Ethernet	CAN and COM1	TB56xx-2ETH

To add Symbol Configuration, select the Application object in the Automation Builder 2.2.0 device tree and add object Symbol configuration (Project-> Add Object->Symbol configuration). The symbol configuration editor appears.



Check the option 'Support OPC UA Features', then click Add button.

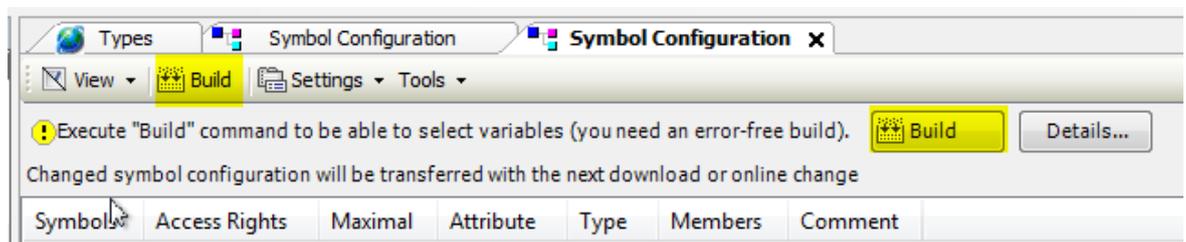


NOTICE

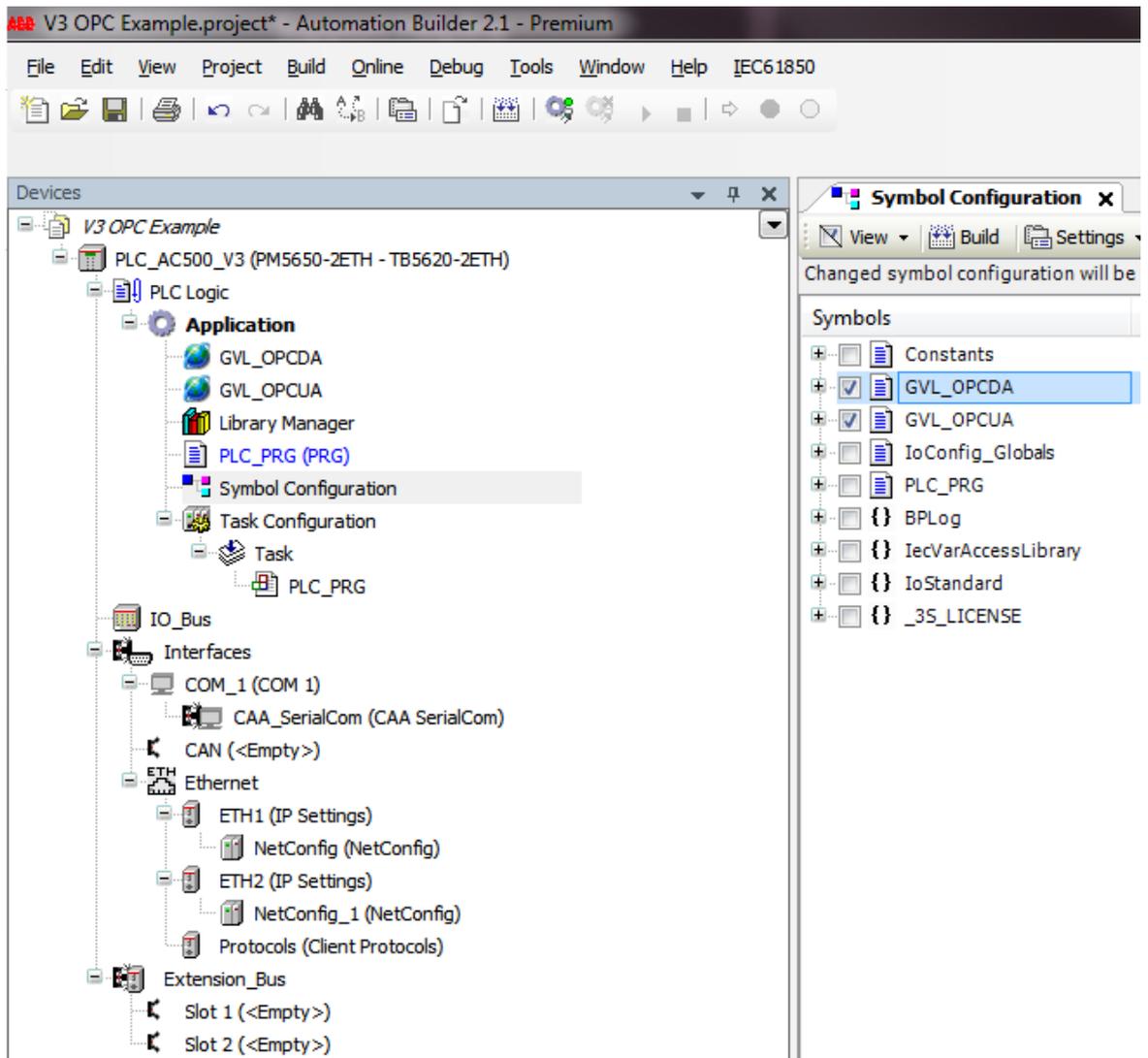
Symbol includes the Items (variables) which exchanges with PLC, this is needed for OPC communication.

You can ignore this step if you only want to configure OPC DA communication, instead of pressing 'Add' button with default settings directly.

In Symbol Configuration editor, execute 'Build' command to enable the variables selection (you need an error-free build).



Then you will see all Items (variables) in the window and select the desired Items which exchanges with PLC, these are need for OPC communication.



Go to Online-> Login, with commanding Login the application will be compiled and downloaded.



NOTICE

Do not configure the program as a cyclic program, please use a task configuration. Go to the Task Configuration and have a look to the Monitor information. For example the program has a cycle time of 40ms, use a task time of 50 or 60ms. So the CPU has time to answer the OPC request from the OPC Server between the tasks.

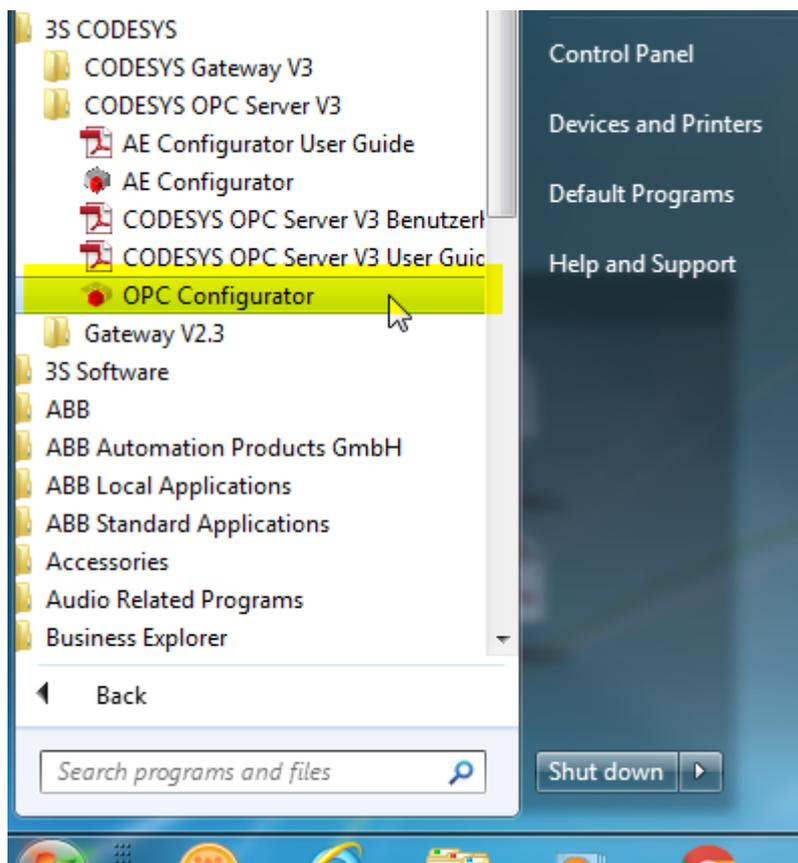
5 Configuration of the OPC server v3

For the connection between OPC DA(AE) client and PLCs there are parameters defined in the file 'OPCServer.ini'.

The configuration of the OPC Server can be done with the tool OPCconfig.exe and then will be stored in the OPCServer.ini file.

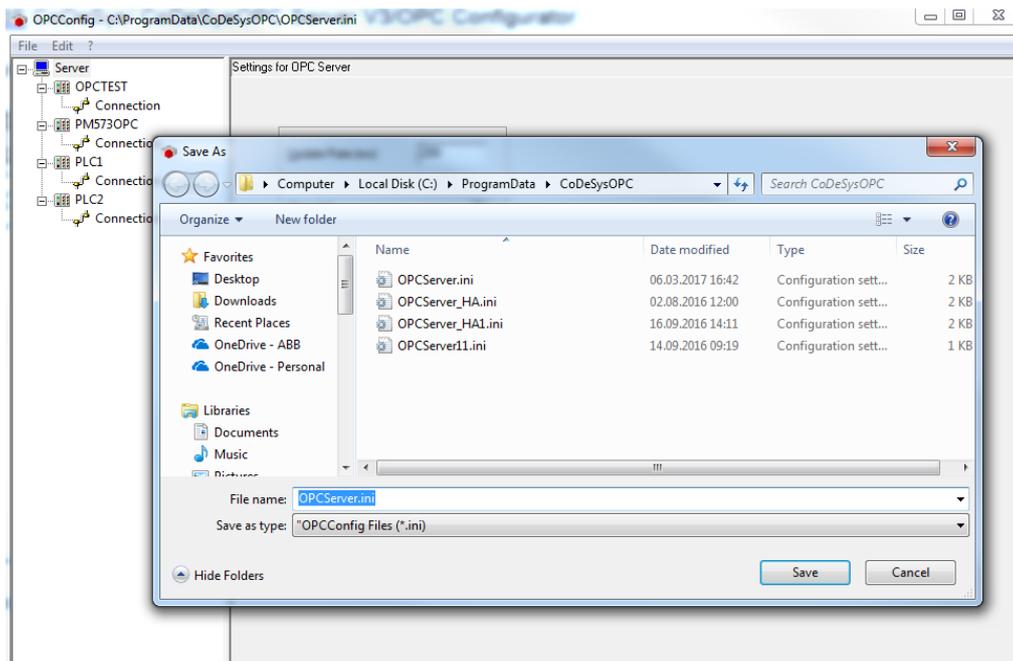
5.1 Open a particular INI file

Start via 3S CODESYS/CoDeSysOPC Server V3/OPC Configurator, or start the OPC configurator by folder C:\Program Files (x86)\3S CODESYS\CODESYS OPC Server 3\ OPCConfig.exe directly.



When "OPCConfig.exe" is called, the current configuration of the "OPCServer.ini" is displayed.

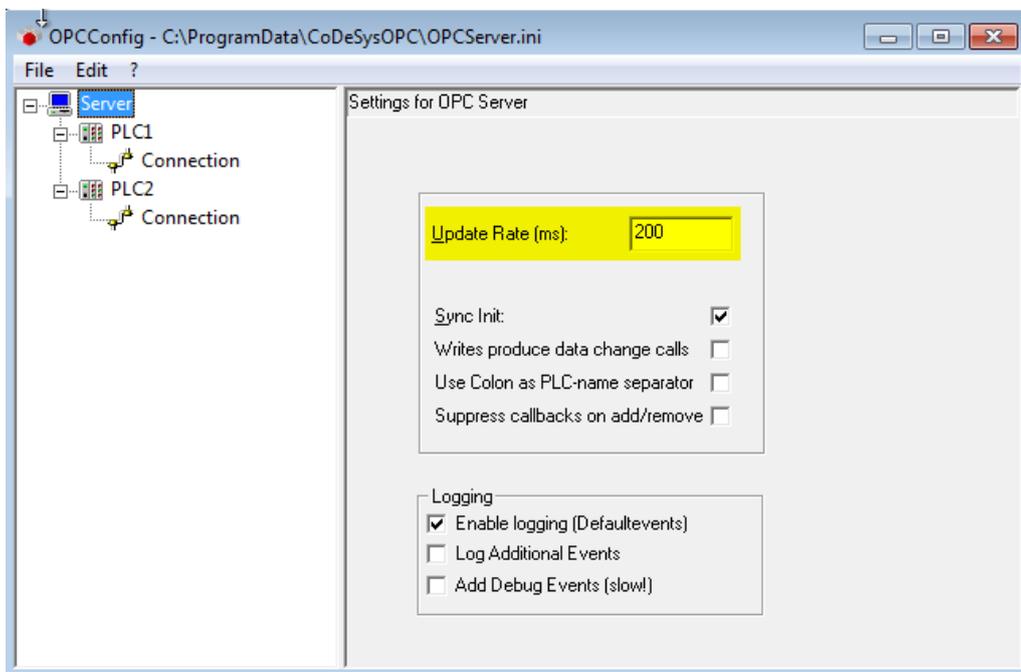
If the configuration is required for other project purposes, also save it under a new name.



5.2 Settings for OPC server

This dialog is available when the root entry Server is selected in the configuration tree. Basic update Rate of the OPC-Server in milliseconds. This is the cycle time according to which all item data are read from the PLC. The data get written into the cash with which the client communicates with a separately defined update rate.

Normally the default setting would be preferred. However, it depends on the application how many Items are expected to communicate.



Logging

- Enable Logging (Defaultevents): Default: active

If this option is activated, any actions as well as errors on the OPC server will be recorded in a LOG file. This file will be stored in the installation directory and will be named OPCServer.log. After a shutdown of the OPC Server the LOG file can be evaluated. The messages of several OPC sessions are lined up in one LOG file until it reaches the size of 1 MB. Then the current date will be added in the file name (OPCServer<date>.log, e.g. OPCServer17.03.2017.log) and the file will be saved. After that a new LOG file OPCServer.log will be created.

- Log Additional Events: Default: not active

If this option is active, then additional events are written into the LOG file.

- Add Debug Events (slow): Default: not active

If this option is active, then debug events are written into the LOG file.

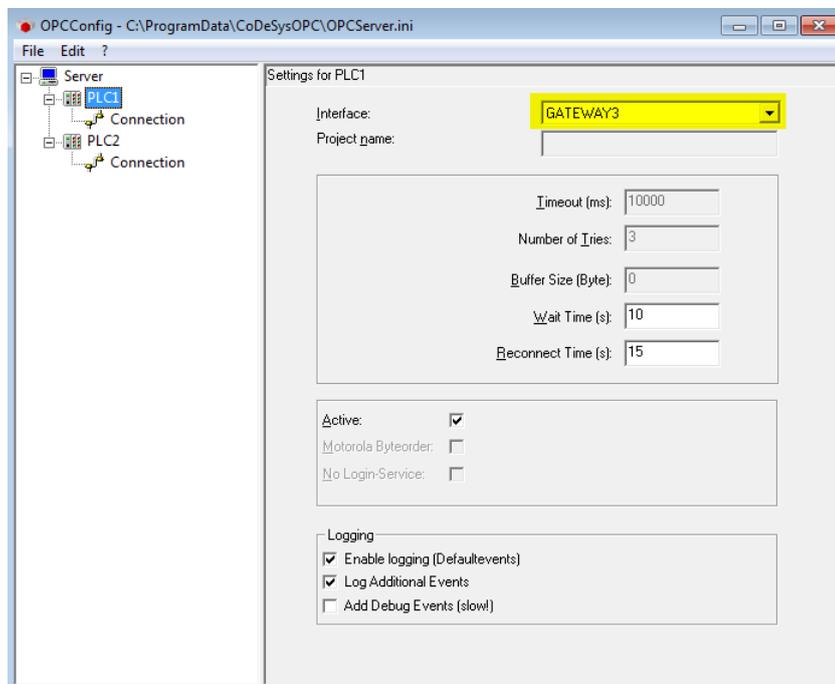


NOTICE

Update Rate may not be 0ms, The default value of 200ms is suitable value of many applications. The adjustment for the Update Rate depends on the number of symbols (variables). For a big number of symbols it would be better to increase the update rate.

5.3 Settings for <PLC>

This dialog is available, if the PLC entry is selected in the configuration tree.



Choose 'GATEWAY3' as the Interface.

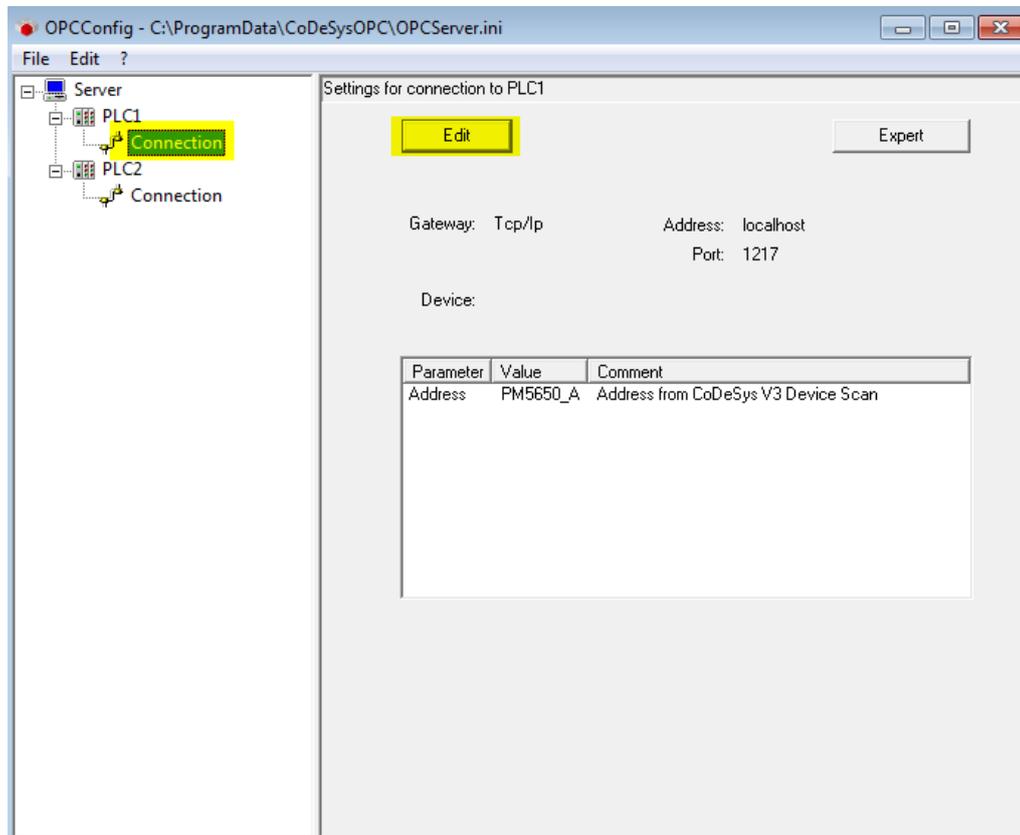
Concerning the Project name, this is only used if interface SIMULATION3 is contemporary used too. The name of the symbol file has to be set here accordingly.

The checkbox 'Active' must be checked.

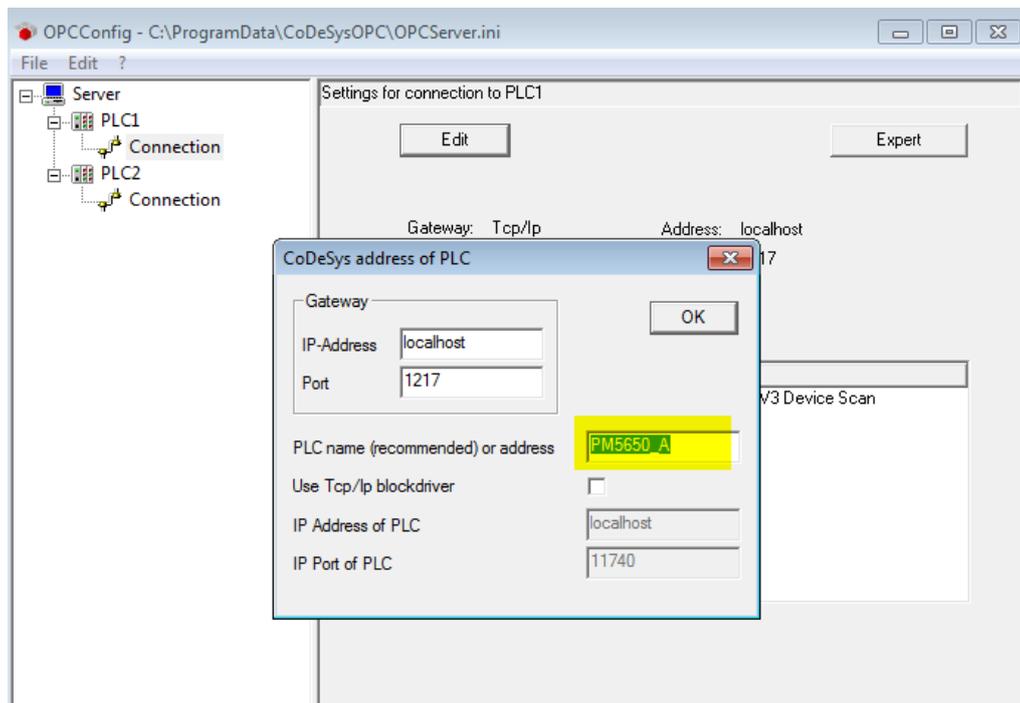
The checkboxes 'Enable logging', 'Log Additional Events' allows a later diagnosis.

5.4 Settings for connection to <PLC>

This dialog will be available if entry Connection below a <PLC> is selected in the configuration tree.

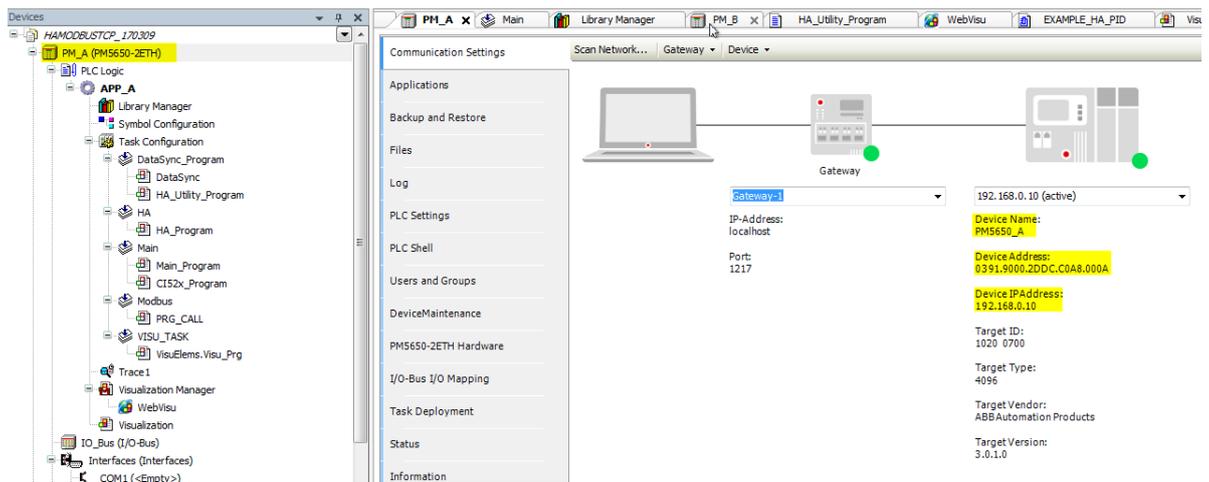


To create a new communication or to change an existing one, open the CoDeSys address dialog of the PLC via the Edit button.

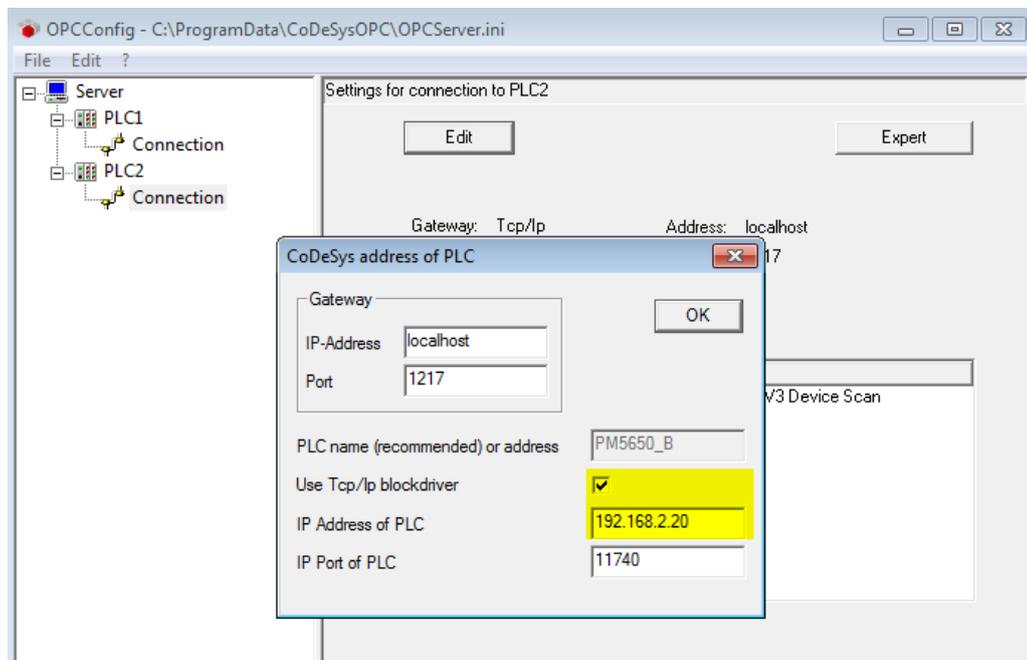


Enter the IP address and port for the gateway and for identifying the PLC specify either the name or address.

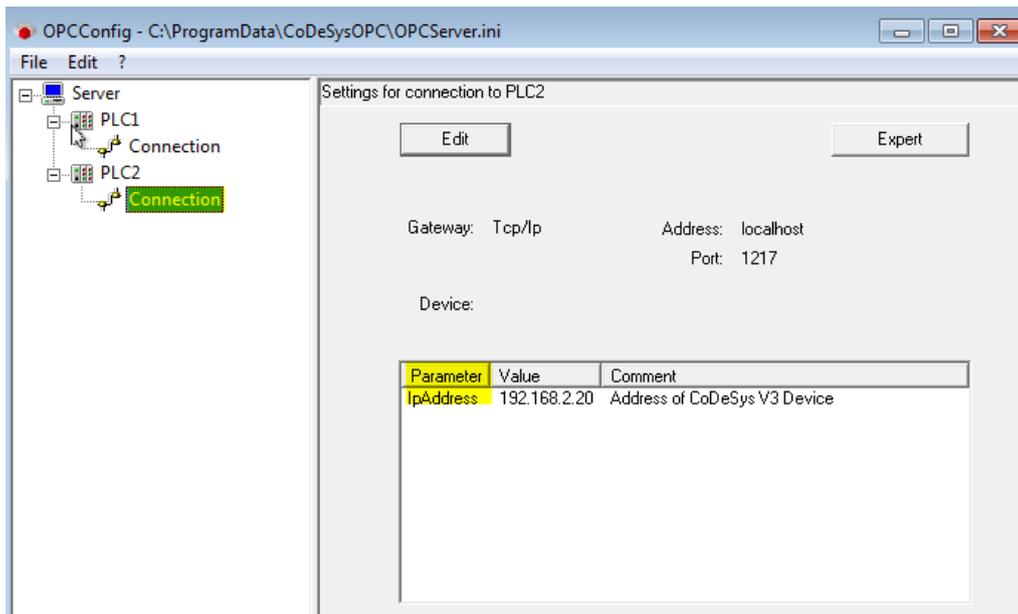
Each the settings must correspond to those defined in Communication Settings tab of the PLC.



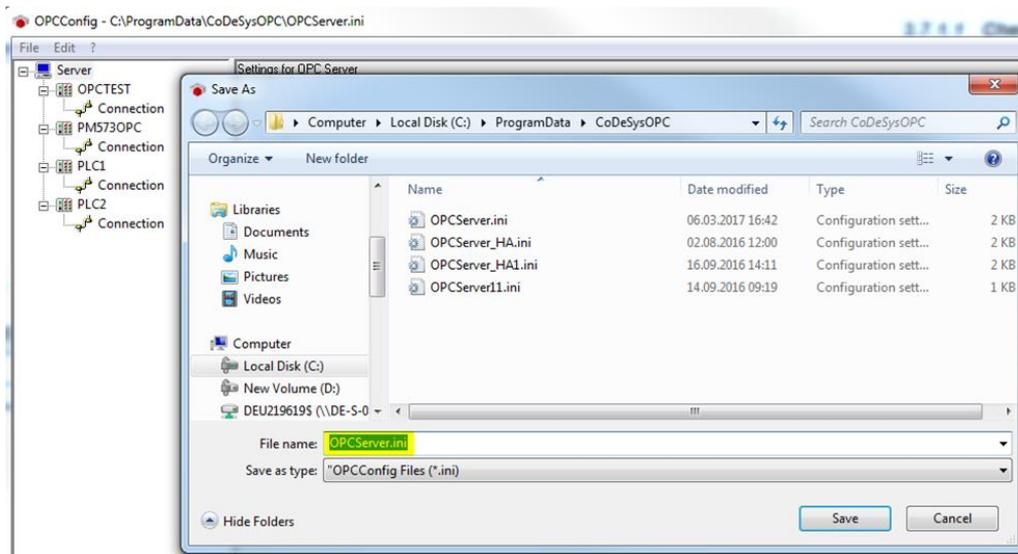
Alternatively, you can enter the address and port of the TCP/IP blockdriver.



After closing the dialog with OK, the settings will be displayed in the main dialog.



If more than one PLC, repeat the same steps for the other PLCs.
 Save as by default "OPCServer.ini" in the installation directory.



Exit

6 Check OPC function with AC500

If the PLC(Device)configured in the CODESYS project is not available, the OPC server allow to work in simulation mode.



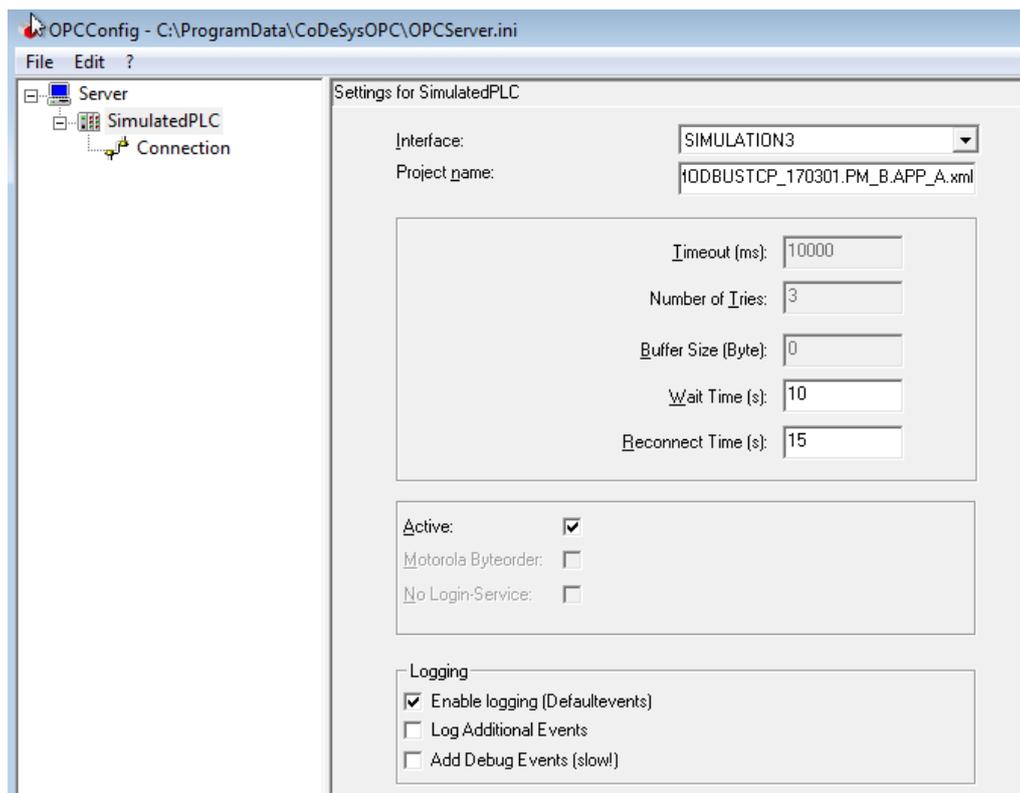
NOTICE

It is strongly recommended to check the function of the previous configuration steps.

6.1 Check OPC Server V3 without AC500

In the OPC server INI file, a simulation access is configured by selecting the Interface 'SIMULATION3' and by setting the name of the symbol file in Project name. The symbol file is automatically generated by a build command of a V3 project when a symbol configuration exists and is stored with the name extension XML next to the project file. If this file is stored in the OPC server directory, then only the project name has to be specified here. But it can also be copied to any location, then under Project name also the directory name has to be specified.

Dialog with configuration of a simulated access to the stock of symbols of Simulated PLC



The directory and the name of the symbol file has to be set. The directory need not to be set, if the file is stored in the same directory as OPCServer.ini.

Start MatrikonOPC Explorer, Connect CoDesys.OPC.DA, Add Group, Add Items, Select Available Items in 'Server CoDeSys.OPC.DA', Add to Tag List, Close the Item browser...

The screenshot shows the MatrikonOPC Explorer interface. The left pane displays a tree view with 'Localhost \\\TEST-PC' expanded to show 'CoDeSys.OPC.AE.1' and 'CoDeSys.OPC.DA'. Under 'CoDeSys.OPC.DA', there is a group named '\$Group_2'. The main pane shows the 'Contents of \$Group_2' with a table of 37 items. All items have a 'Quality' of 'Good, non-specific' and a 'Status' of 'Active'. The bottom status bar indicates the server is 'CoDeSys.OPC.DA', it is 'Connected: Yes', and 'State: Running'. A central banner for MatrikonOPC is also visible.

Item ID	Access Path	Value	Quality	Timestamp	Status
PLC1.APP_A.Types.A...		0,0,0,0,0,...	Good, non-specific	03.17.2017 3:57:42.601 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.A...		0	Good, non-specific	03.17.2017 3:57:42.610 PM	Active
PLC1.APP_A.Types.C...		*1406631,...	Good, non-specific	03.17.2017 3:58:14.679 PM	Active
PLC1.APP_A.Types.C...		1406631,25	Good, non-specific	03.17.2017 3:58:14.679 PM	Active
PLC1.APP_A.Types.C...		13913584	Good, non-specific	03.17.2017 3:58:14.679 PM	Active

If anything is right, then CoDeSys.OPC.DA is connected. The OPC Client is running and the quality of the items is good.

NOTICE Connection settings is not necessary for the simulation. MatrikonOPC Explorer is a free OPC Client packed with functionality for testing and troubleshooting OPC servers and OPC connections.

Check Processes with Windows Task Manager

The screenshot shows the Windows Task Manager 'Processes' tab. The 'MobileOPCExplorer.exe' and 'WinCoDeSysOPC.exe' processes are highlighted in green. The status bar at the bottom shows 'Processes: 56', 'CPU Usage: 1%', and 'Physical Memory: 28%'.

Image Name	User Name	Session ID	CPU	Memory (...)	Description
MobileOPCExplorer.exe	Test	1	02	6,336 K	MatrikonOPC Explorer
taskmgr.exe	Test	1	00	1,772 K	Windows Task Manager
shstat.exe	Test	1	00	292 K	VirusScan tray icon
McTray.exe	Test	1	00	188 K	McTray Application
WinCoDeSysOPC.exe	Test	1	00	2,244 K	WinCoDeSysOPC.EXE
explorer.exe	Test	1	00	16,408 K	Windows Explorer
dwm.exe	Test	1	00	1,068 K	Desktop Window Manager
taskhost.exe	Test	1	00	5,768 K	Host Process for Windows Tasks
GatewaySysTray.exe	Test	1	00	760 K	GatewaySysTray
jucheck.exe	Test	1	00	3,508 K	Java Update Checker
VBoxTray.exe	Test	1	00	1,408 K	VirtualBox Guest Additions Tra...
UdaterUI.exe	Test	1	00	1,056 K	Common User Interface
jusched.exe	Test	1	00	1,300 K	Java Update Scheduler
winlogon.exe	SYSTEM	1	00	1,500 K	Windows Logon Application
csrss.exe	SYSTEM	1	00	1,220 K	Client Server Runtime Process
CodeMeterCC.exe	Test	1	00	2,668 K	CodeMeter Control Center

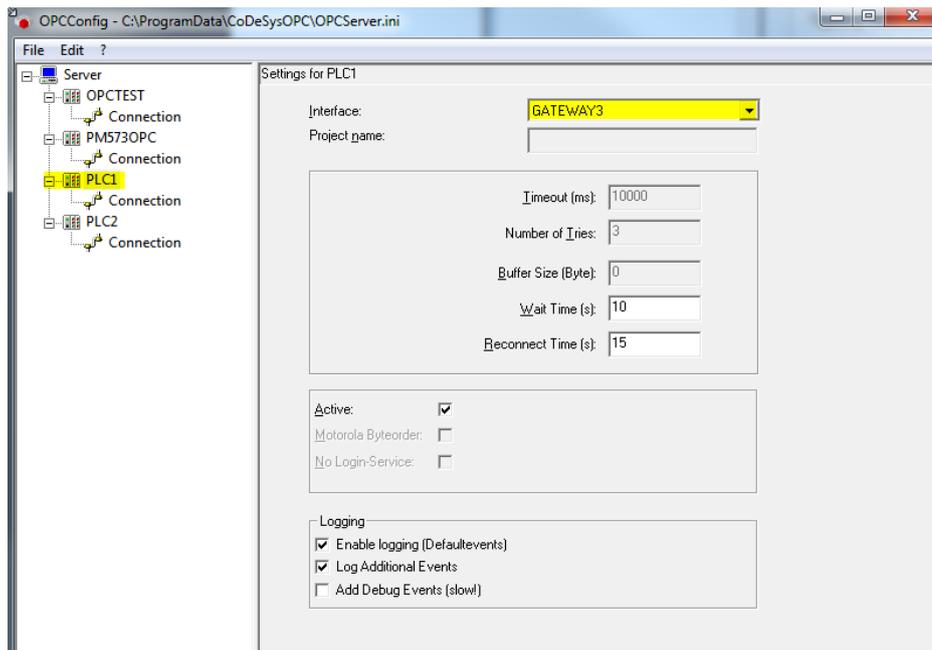
**NOTICE**

The correct function of OPC Server V3 can be checked without AC500. With OPC Server V3 with the configuration SIMULATION3 the Project name with the directory name has to be specified. The values of the items can be read and wrote by one OPC client.

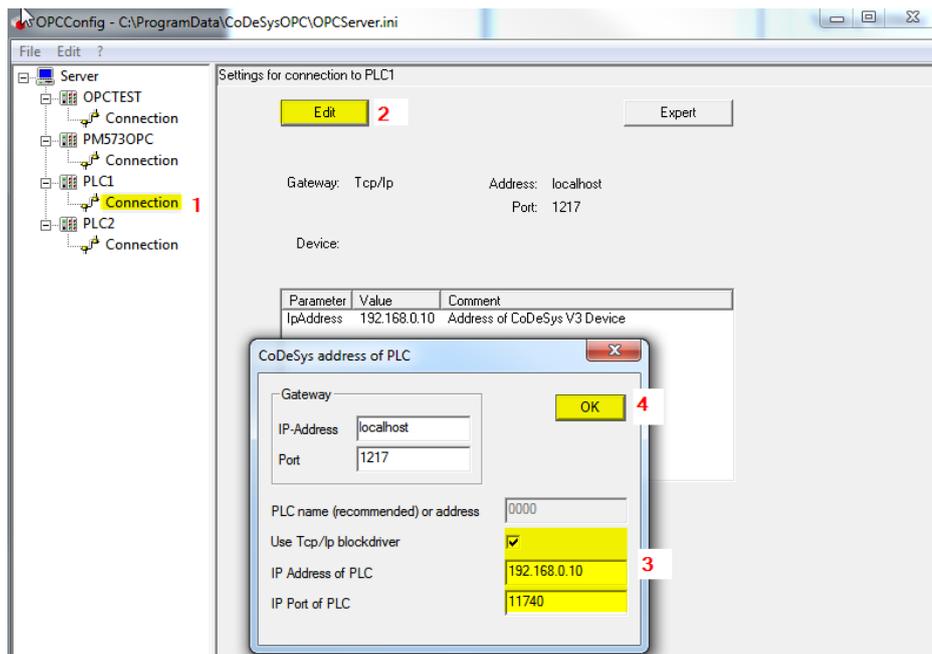
6.2 Check OPC Server V3 with AC500

In this section, you will make the configuration for OPC Server V3 and check the communication with AC500.

On the client computer, launch the application 'OPC configurator'. After that, select Edit>Append PLC, a new PLC will be added under the Server tree.



Now you need to set the IP address and Port in Connection settings Editor.



Save and close the OPCConfig Editor.

Start MatrikonOPC Explorer, Connect CoDesys.OPC.DA, Add Group, Add Items, Select Available Items in 'Server CoDesys.OPC.DA',Add to Tag List, Close the Item browser...

The screenshot shows the MatrikonOPC Explorer interface. The main window displays a tree view on the left with a group named '\$Group_1' selected. The central pane shows the 'Contents of \$Group_1' with a table of items. The bottom of the window is divided into three sections: 'Server Info', a central banner for MatrikonOPC, and 'Group Info'.

Item ID	Access Path	Value	Quality	Timestamp	Status
PLC1.APP_A.Types.Application_Data		44,66,88,99,100,11,222,333,444,888,55,66...	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[0]		44	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[10]		55	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[11]		66	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[12]		77	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[13]		88	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[14]		99	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[15]		11	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[16]		22	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[17]		33	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[18]		44	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[19]		55	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[1]		66	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[20]		77	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[2]		88	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[3]		99	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[4]		100	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[5]		11	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[6]		222	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[7]		333	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[8]		444	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.Application_Data[9]		888	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.CP600		"679823,9375","1359647,875","2023247,25"...	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.CP600[0]		679823,9375	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.CP600[10]		6807022,5	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.CP600[1]		1359647,875	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.CP600[2]		2023247,25	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.CP600[3]		2719295,75	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.CP600[4]		3392416,5	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.CP600[5]		8057964,5	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.CP600[6]		4317621	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.CP600[7]		4934424	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.CP600[8]		5551227	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.CP600[9]		6168030	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.HA_A_PRIMARY		True	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.HA_B_PRIMARY		False	Good, non...	03.20.201...	Active
PLC1.APP_A.Types.fg_HA_PrIm_Mod_OP...		True	Good, non...	03.20.201...	Active

Server Info:
 Server: CoDesys.OPC.DA
 Connected: Yes
 State: Running
 Groups: 1
 Total Items: 37
 Current Local Time: 03.20.2017 12:22:22.156 PM
 Update Local Time: 03.20.2017 12:22:22.027 PM

Group Info:
 Group: \$Group_1
 Connected (Async I/O): Yes (2.0)
 Active: Yes
 Items: 37
 Current Update Rate: 1000 ms
 Percent Deadband: 0,00%
 Data Change Rate: 13,14 Items/Sec

Now the variables will be shown and automatically updated.

If anything is right, then CoDesys.OPC.DA is connected. The OPC Client is running and the quality of the items is good. The values of the items can be read and wrote by one OPC client.

7 OPC UA Client for test

This section describes how to connect between AC500 V3 and OPC UA Client, for example, 'Unified Automation UaExpert', which you download from Unified Automation GmbH company.

The configuration Tools is used to configure the UaExpert's runtime settings. It is a specialized UA-Client, capable of the OPC UA DI Profile. The configuration Tool connects to the UaExpert using an OPC UA connection. To use the Configuration Tool, the UaExpert has to be started and running.

On the first start, the configuration tool will create a UA application instance certificate. Therefore, several information is requested and must be set in the dialog.

New Application Instance Certificate

Subject:

Common Name: PLC_Supportt@DE-L-0242757 ✓

Organization: ABB ✓

Organization Unit: ✕

Locality: Heidelberg ✓

State: Baden württemberg ✓

Country: DE ✓
(Two letter code, e.g. DE, US, ...)

OPC UA Information

Application URI: n:DE-L-0242757:UnifiedAutomation:UaExpert@DE-L-0242757 ✓

Domain Names: DE-L-0242757 ✓

IP Addresses: ✕

Certificate Settings

RSA Key Strength: 2048 bits Certificate Validity: 1 Year

Password protect private key

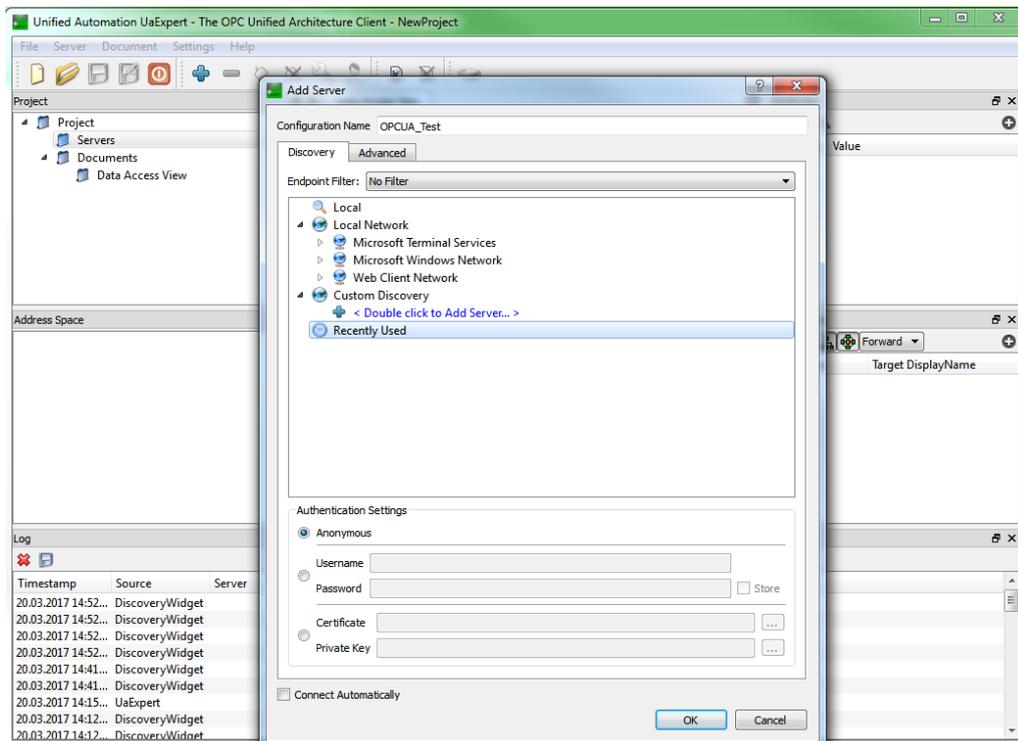
Password: ✕

Password (repeat): ✕

OK Cancel

After all sections are filled out, the application instance certificate can be generated by pressing the 'OK' button.

To add and connect a UA Server to the UaExpert. Click on 'Add Server...' to add a connection to a new UA Server.

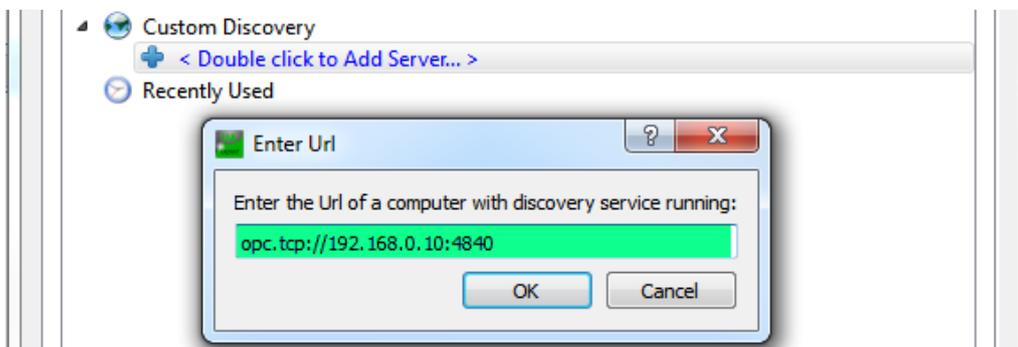


A dialog 'Add UA Server' appears where all need information about the target UA Server has to be configured. To find the remote UA Server, you can browse the network and discover individual computers, or double click on 'Double click to Add Server...' and enter the remote address of the targeted UA Server.

This can be done in different ways

opc.tcp://<hostname>:4840

opc.tcp://<IP Address>:4840



Concerning the Security Settings, choose 'None' at Security Policy for the moment.

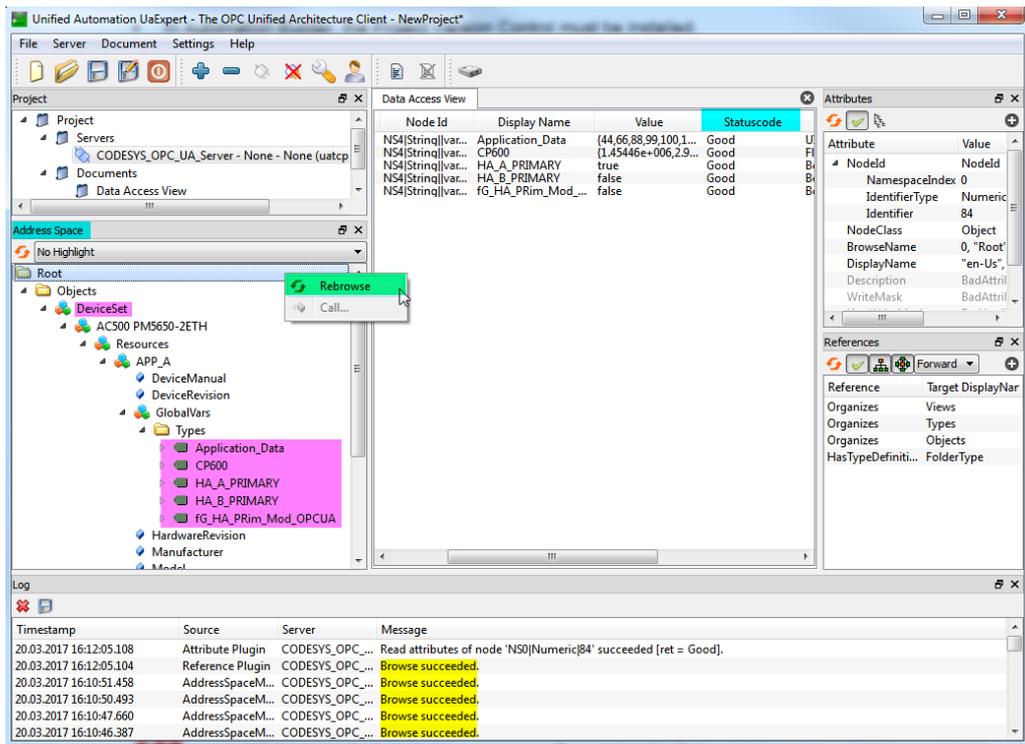
The screenshot shows a dialog box titled "Server Settings - OPC UA TEST". It contains the following fields and options:

- Server Information:** Endpoint Url: `opc.tcp://192.168.0.10:4840`
- Security Settings:** Security Policy: `None`; Message Security Mode: `None`
- Authentication Settings:** Anonymous; Username: ; Password: Store; Certificate: ; Private Key:
- Session Settings:** Session Name: `757:UnifiedAutomation:UaExpert@DE-L-0242757`

Buttons:

To check if the connection has been established, right click on the server in the 'Address Space' and click 'Rebrowse'. The displayed connection status of the server should now show 'succeeded'.

Drag the desired Items to the Data Access View.



The values of the items can be read and write by one OPC UA client.

For more detailed descriptions please refer to the website

www.unifiedautomation.com.

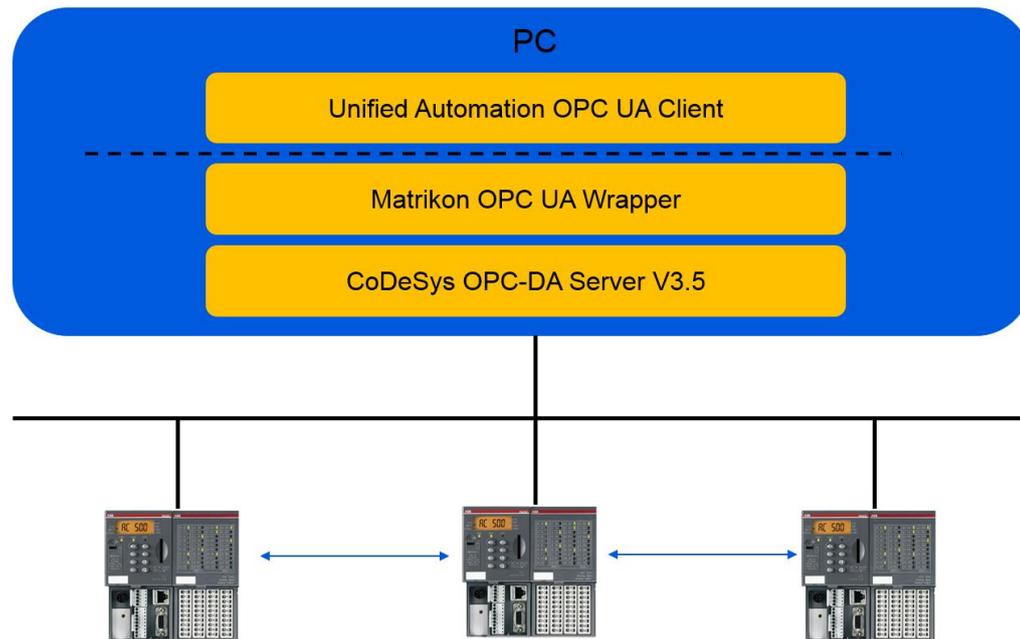


NOTICE

Regarding the OPC UA Client test tool, you can also choose the OPC UA Client from Kepware (Download Free Demo), ... etc.

8 Connection AC500 V2 and OPC UA Client

This chapter describe an opportunity to connect AC500 PLC with standard OPC Server and an OPC UA Wrapper to an OPC UA Client.



Configure your AC500 with Symbol file in the same way you do it for all projects with OPC Access. Repeat the same for the OPC Server with the OPC Configurator.

Please install the Matrikon OPC UA Wrapper (for example) as Administrator and refer to the instructions provided by the OPC UA Wrapper accordingly.

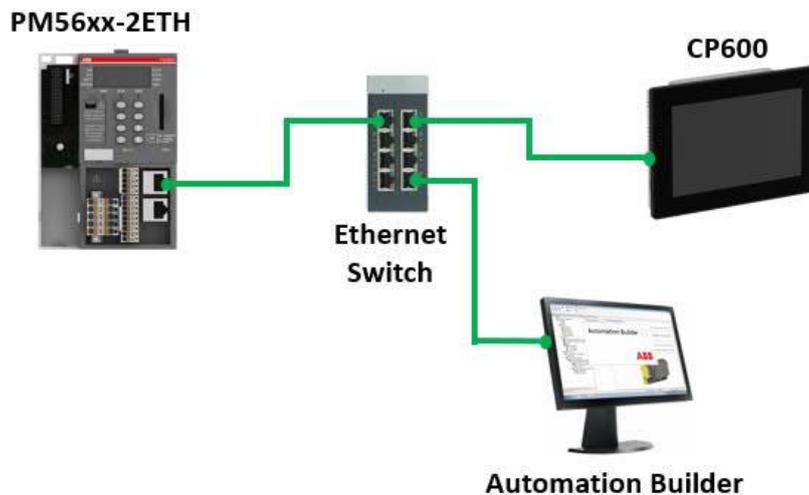
9 Behavior of the OPC UA with CP600

With Panel Builder 600 Version 2.4.0.89 and onwards, the CP600 has the OPC UA Server/Client communication driver. The OPC UA Client is designed to connect HMI devices to OPC UA Servers (e.g. AC500 V3 PM5650-ETH in the application example 'V3 OPC Example_V2.5.project').



OPC UA Server can be from 3rd-party devices or another CP600 as OPC UA Server.

In this chapter, adding the OPC UA Server and PLC configuration will be explained as well. The diagram below shows the layout of this system.

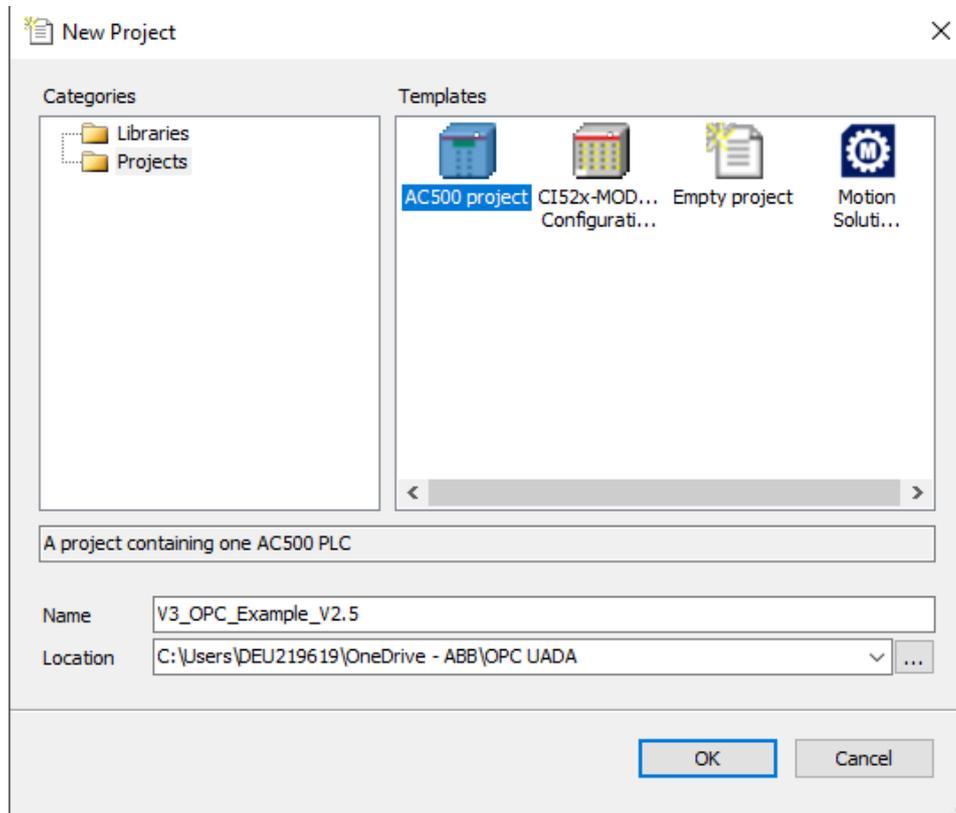


(MAC) address	Device	Port	Serial number	Device ID	IP Address
00-24-59-0D-03-6A	PM5650-2ETH	ETH1	00000259	0x00	192.168.22.10
00-24-59-0D-03...	PM5650-2ETH	ETH2	00000259	0x00	192.168.2.10

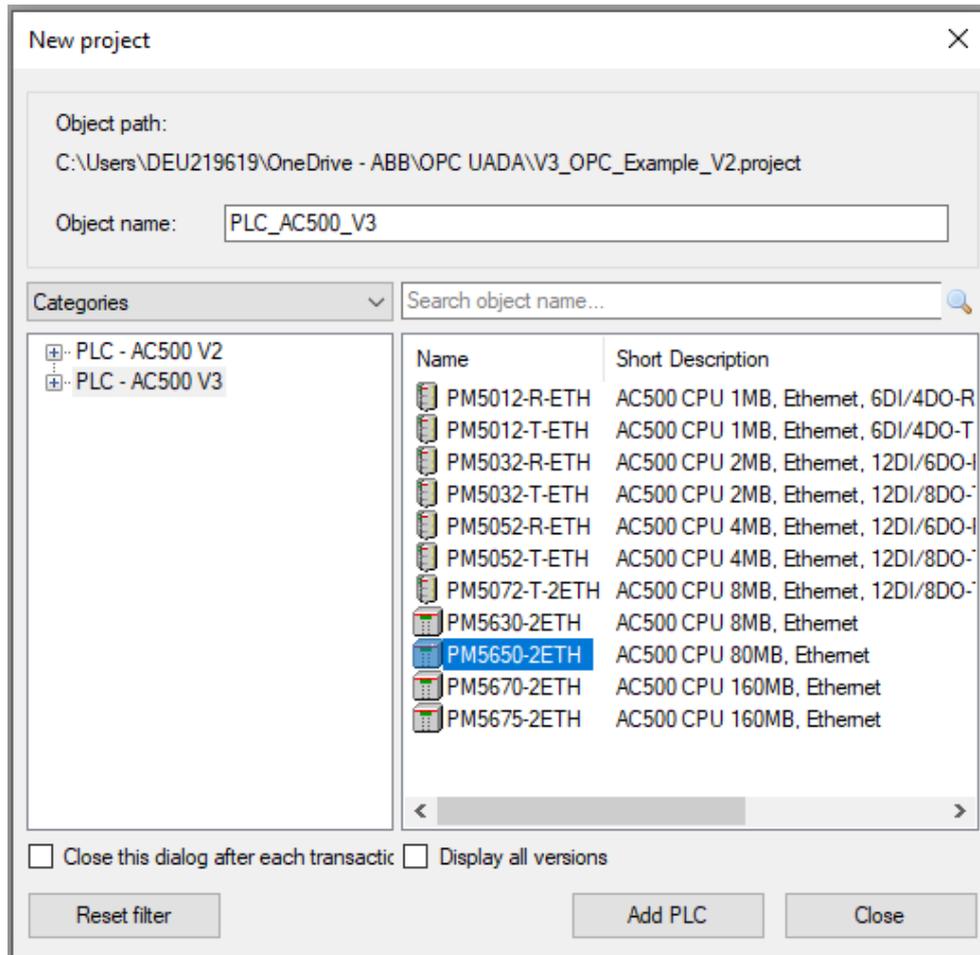
9.1 Protocol Configuration in Automation Builder

9.1.1 Create new project

Select the 'New project...' from the Menu > Files in Automation Builder. At the New Project dialog, select the 'AC500 project' and name your project file.

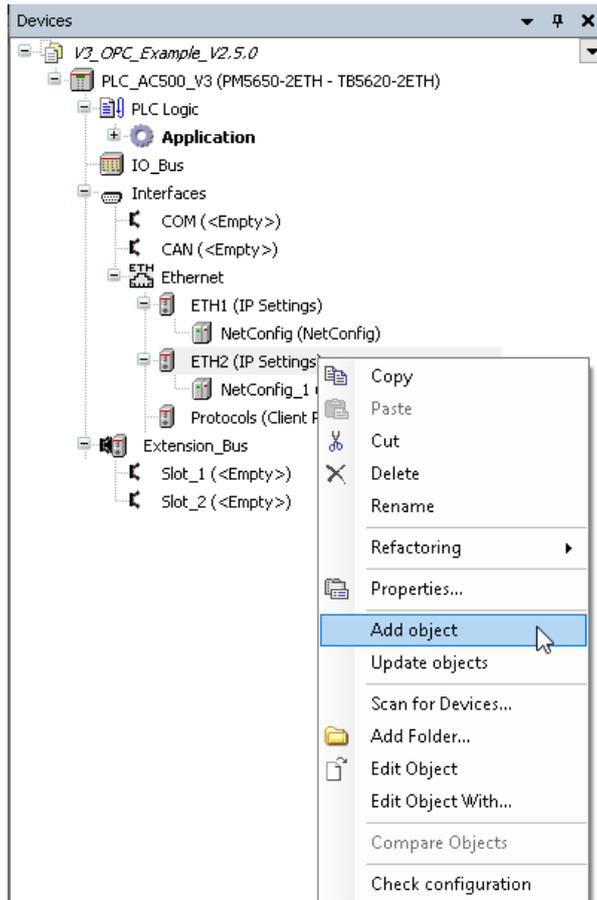


Click OK to continue and choose the AC500 CPU according to your equipment and click 'Add PLC' to proceed.

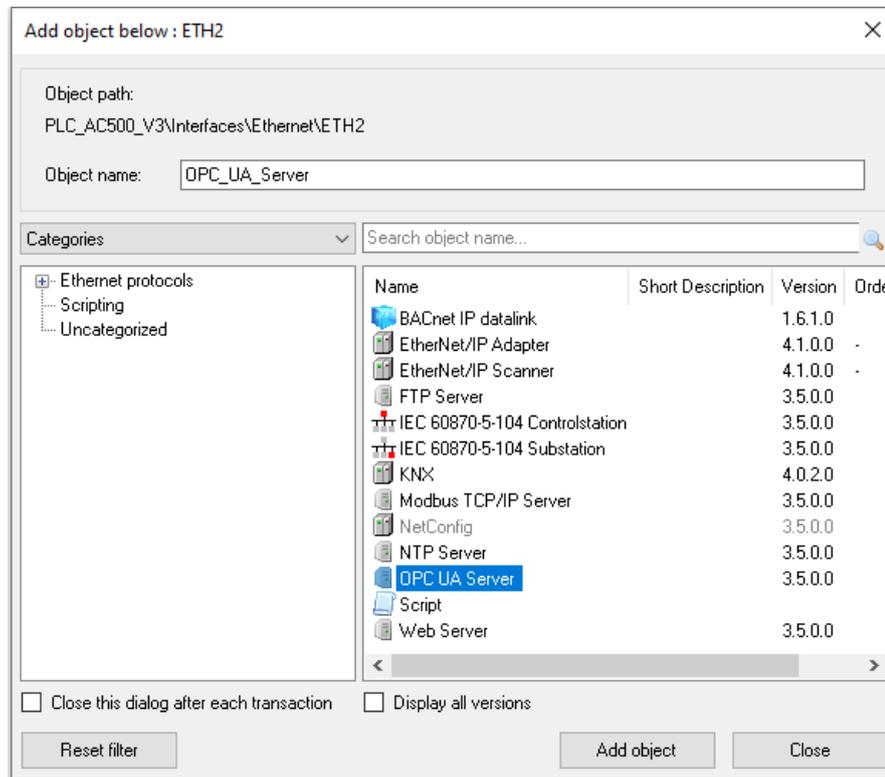


9.1.2 Add OPC UA Server

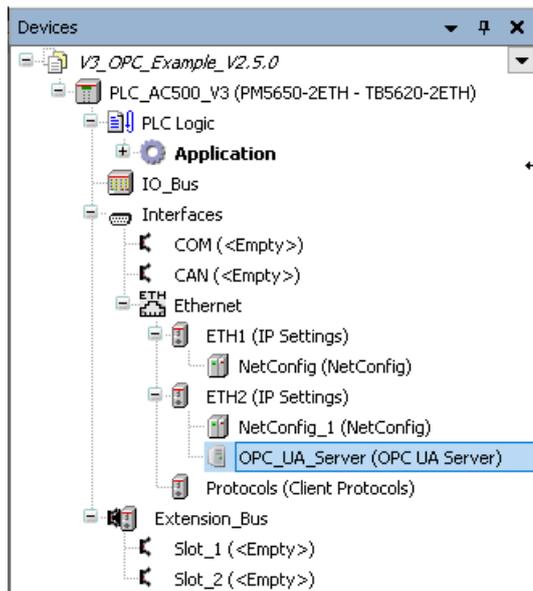
Right-click at the “ETH2 (IP Settings)” under the “Ethernet” and select “Add object”.



Select the “OPC UA Server” and click the “Add object” button to continue.

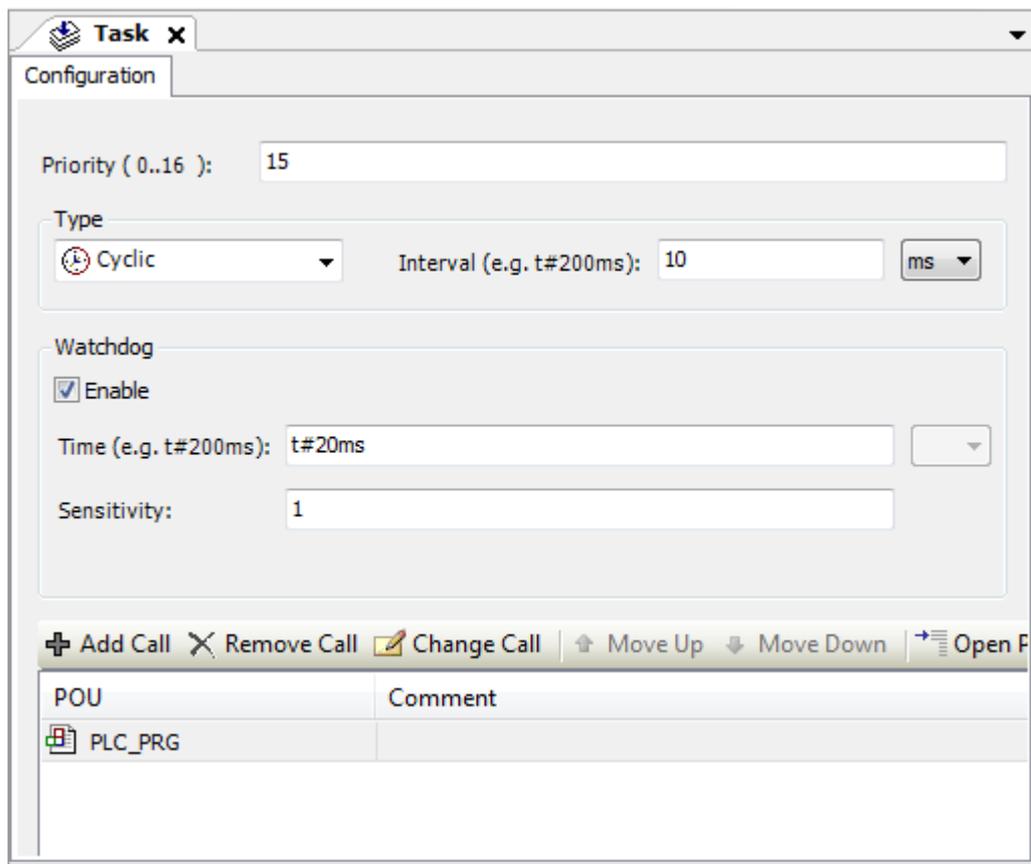


You now have the OPC UA Server on ETH2, if you need the OPC UA Server on ETH1, please repeat the step and select ETH1.



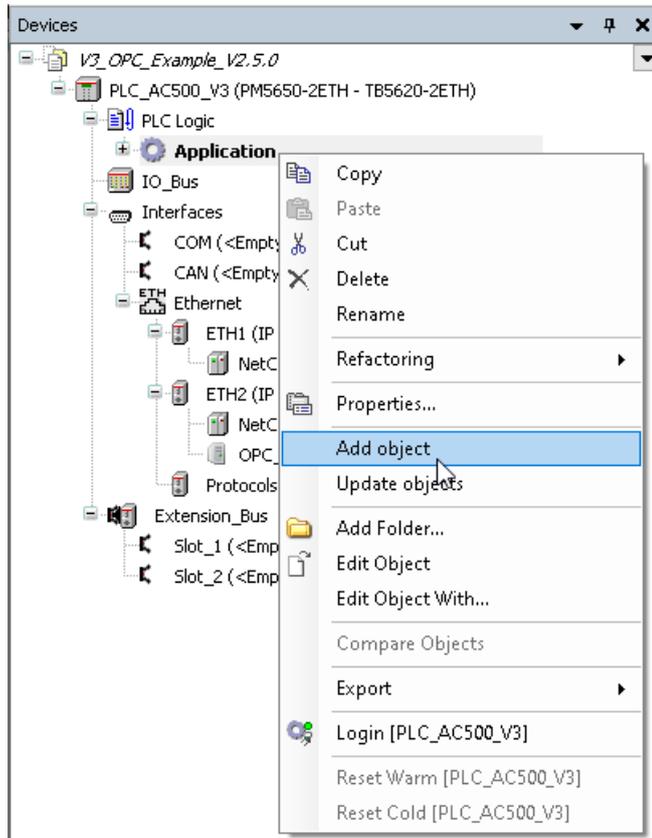
9.1.3 Task Configuration

Double-click on the “Task” and change the setting as below.

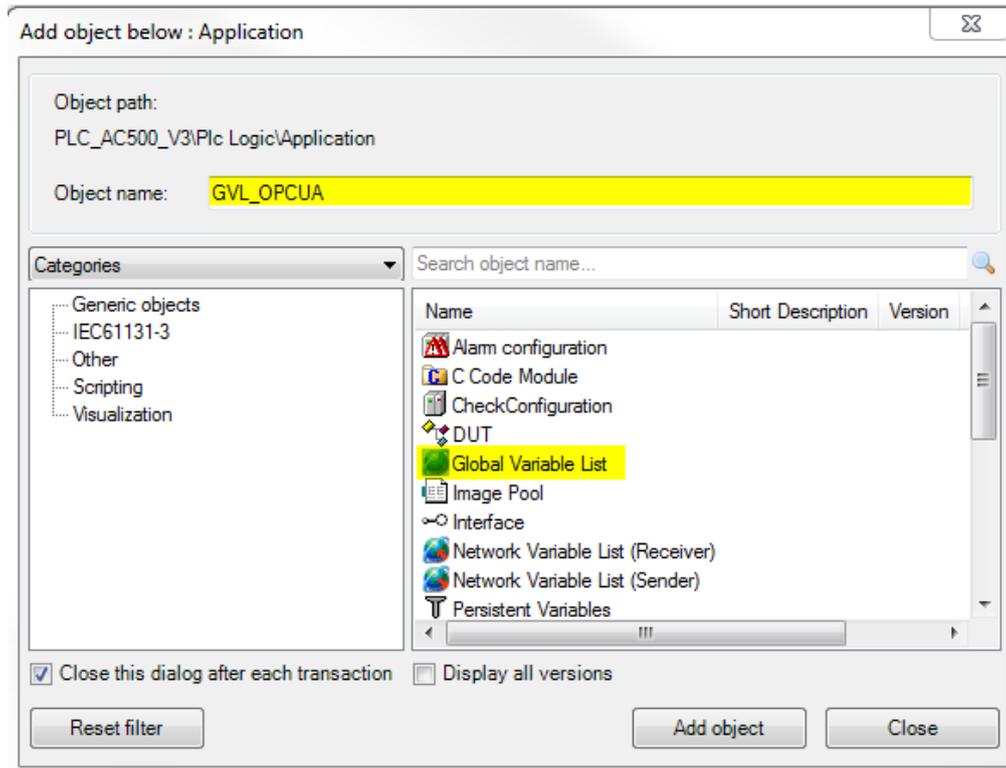


9.1.4 Global variables

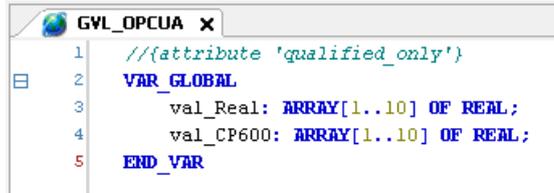
Right-click “Application” from the Device tree in the Automation Builder project and select “Add object”.



Select the “Global Variable List” and change the Object name to ‘GVL OPCUA’, then click “Add object”.



Double-click on the “GVL OPCUA” to open the editor. Create the variables as below.



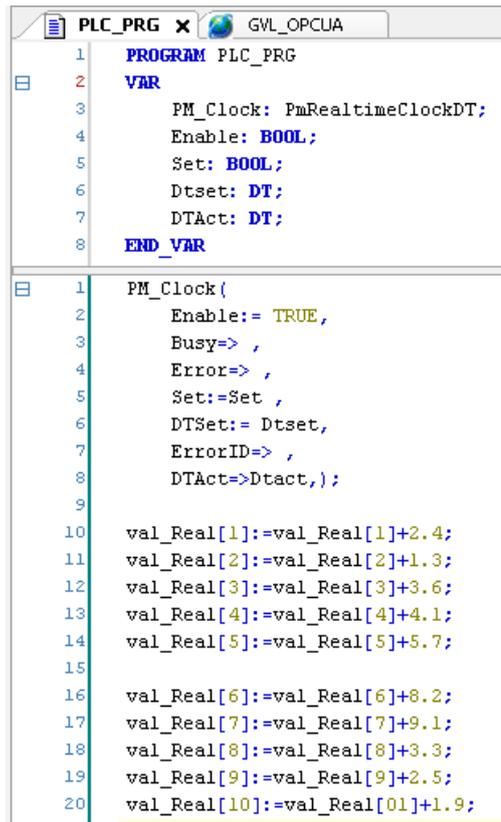
```

1  //(attribute 'qualified_only')
2  VAR_GLOBAL
3      val_Real: ARRAY[1..10] OF REAL;
4      val_CP600: ARRAY[1..10] OF REAL;
5  END_VAR

```

9.1.5 PLC Logic

Double-click on the “PLC_PRG” to open the editor and add the coding as below.



```

1  PROGRAM PLC_PRG
2  VAR
3      PM_Clock: PmRealtimeClockDT;
4      Enable: BOOL;
5      Set: BOOL;
6      Dtset: DT;
7      DTAct: DT;
8  END_VAR

1  PM_Clock(
2      Enable:= TRUE,
3      Busy=> ,
4      Error=> ,
5      Set:=Set ,
6      DTSet:= Dtset,
7      ErrorID=> ,
8      DTAct=>Dtact,);

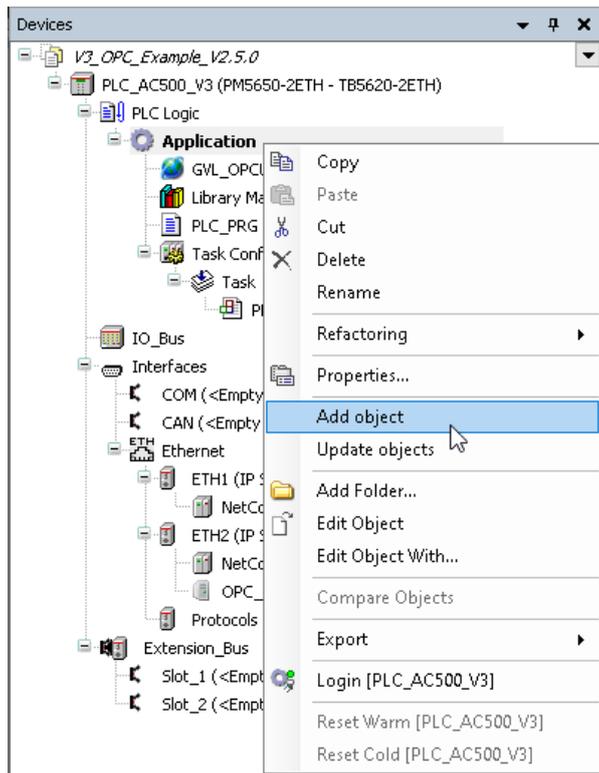
10 val_Real[1]:=val_Real[1]+2.4;
11 val_Real[2]:=val_Real[2]+1.3;
12 val_Real[3]:=val_Real[3]+3.6;
13 val_Real[4]:=val_Real[4]+4.1;
14 val_Real[5]:=val_Real[5]+5.7;

16 val_Real[6]:=val_Real[6]+8.2;
17 val_Real[7]:=val_Real[7]+9.1;
18 val_Real[8]:=val_Real[8]+3.3;
19 val_Real[9]:=val_Real[9]+2.5;
20 val_Real[10]:=val_Real[01]+1.9;

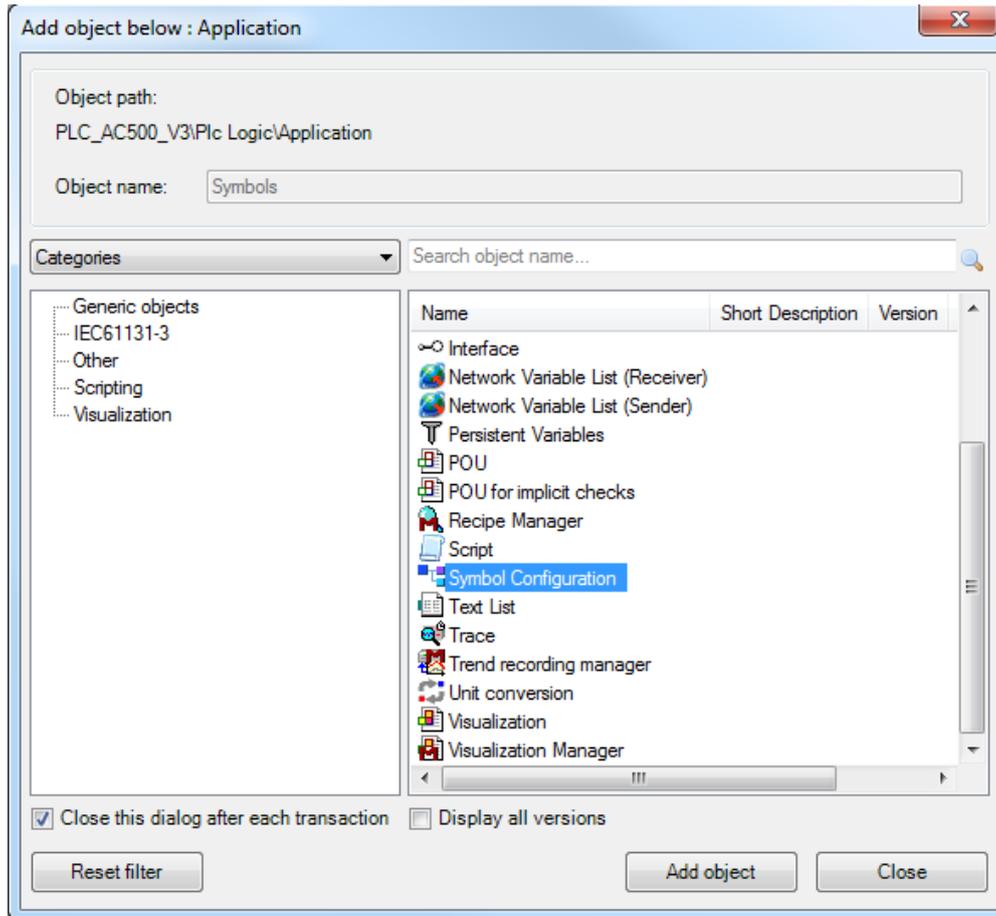
```

9.1.6 Symbol Configuration

Right-click “Application” from the Device tree in the Automation Builder project and select “Add object”.



Select the “Symbol Configuration” and click “Add object”.

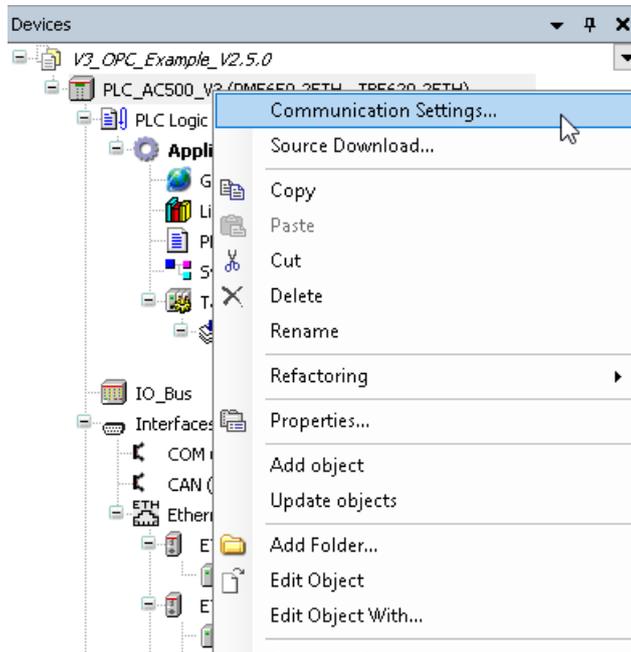


A new “Add Symbol Configuration” windows will pop up.

Check the “Support OPC UA Features” box and click “Add” button to continue.

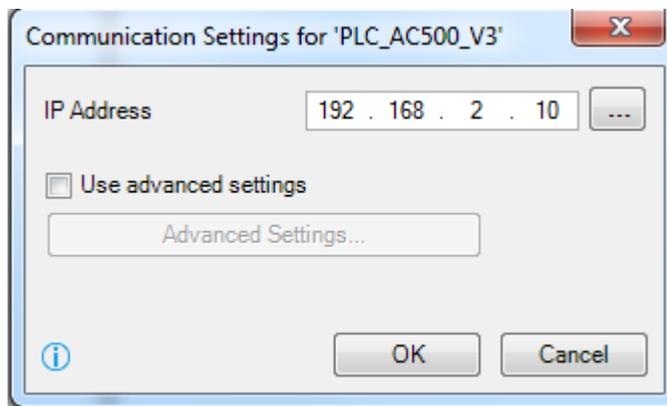
9.1.7 Online to the PLC

Right-click on the “PLC_AC500_V3” and select ‘Communication Settings’.

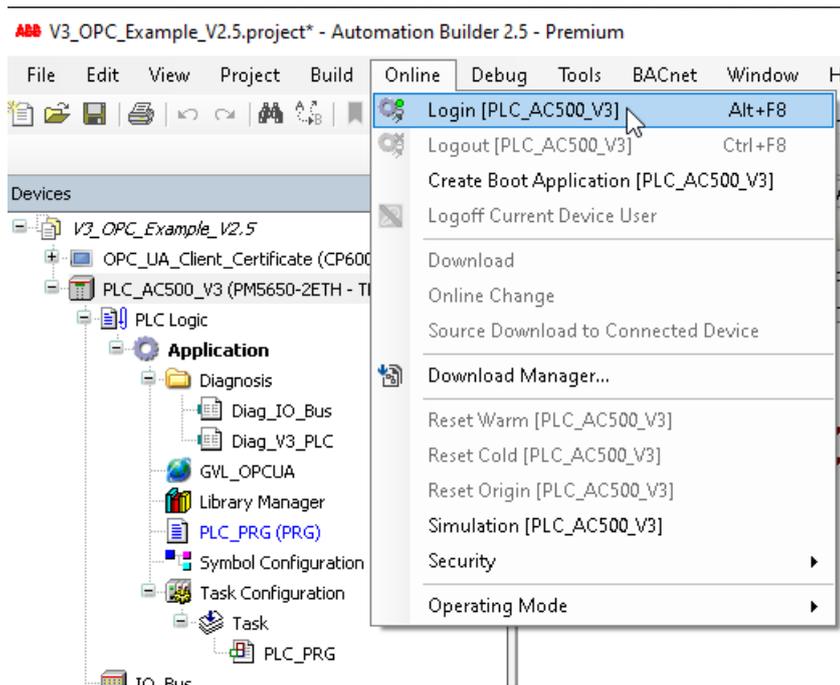


The IP address for ETH2 is “192.168.2.10”. Click “Ok” button to continue.

If your PLC have different IP Address, click the ‘...’ button beside the IP address to launch the network scanning tool to search the CPU IP address.

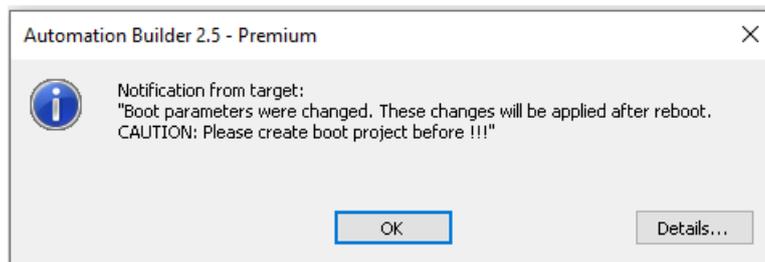


You now ready to go online with the PLC by choosing ‘Login’ from the ‘Online’ menu and in the following pop-up window, click “Yes”.



A windows pop-up with the notification: “Boot parameters were changed. These changes will be applied after reboot.”

This is due to new “OPC UA Server” protocol have been added to the ETH2.

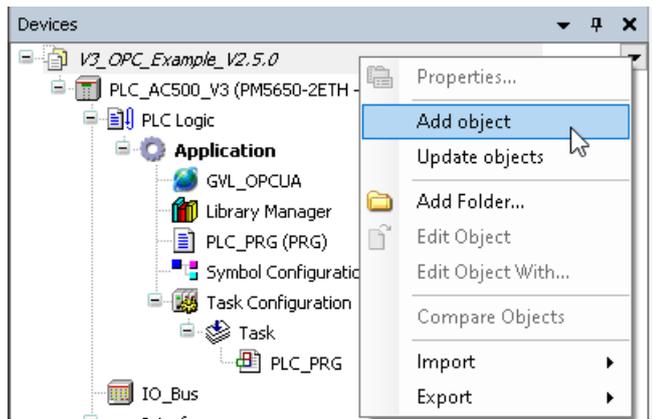


After downloading completed, go offline and reboot the PLC.

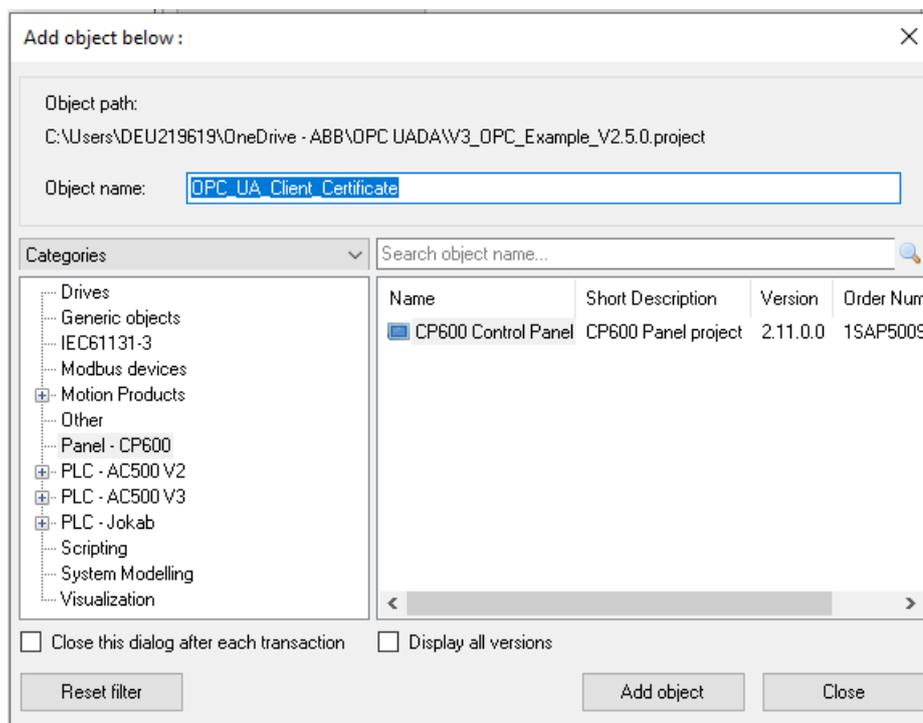
9.2 Panel Builder 600

In this chapter, we will add the CP6607 panel into the project.

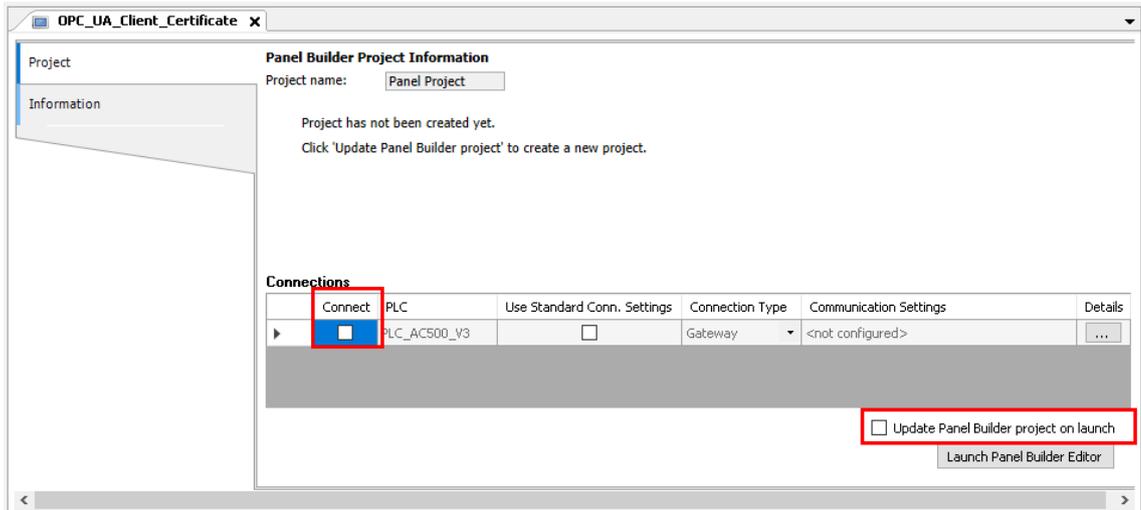
Right-click “V3_OPC_Example_V2.5” from the Device tree in the Automation Builder project and select “Add object”.



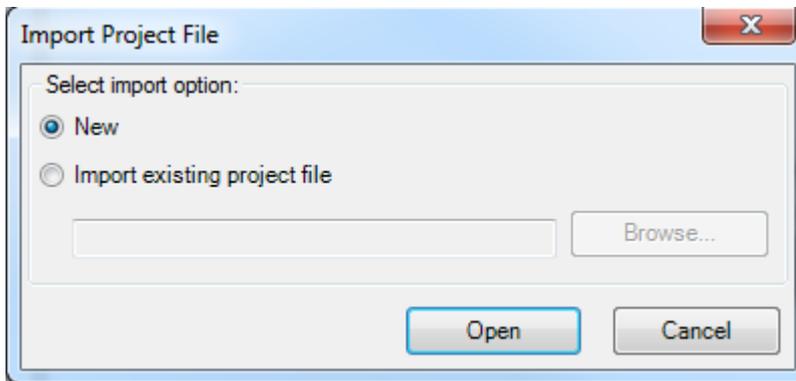
Select the CP600 Control Panel and change the object name to 'OPC_UA_Client_Certificate'



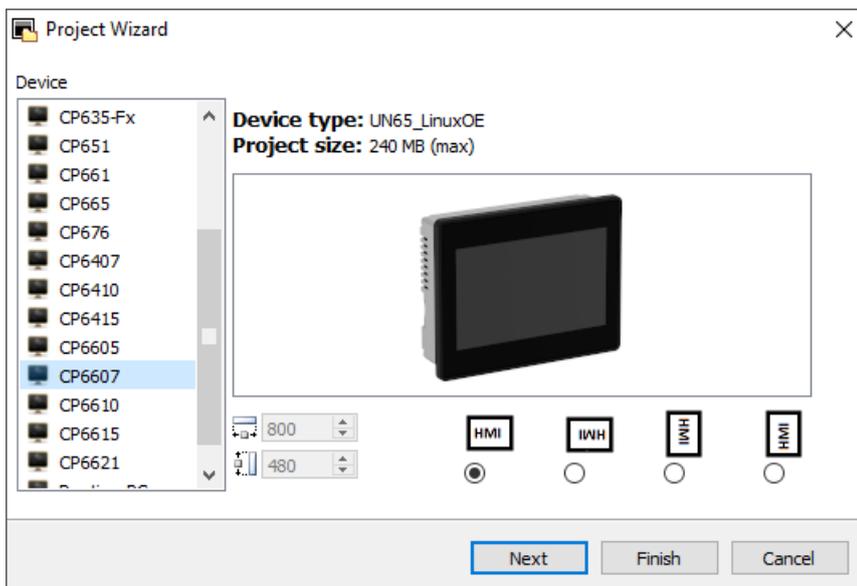
Double-click on the object 'OPC_UA_Client_Certificate'.
 Uncheck the 'Connect' box and uncheck the 'Update Panel Builder project on launch'.
 This is because we don't use the ABB CoDeSys V3 ETH protocol.



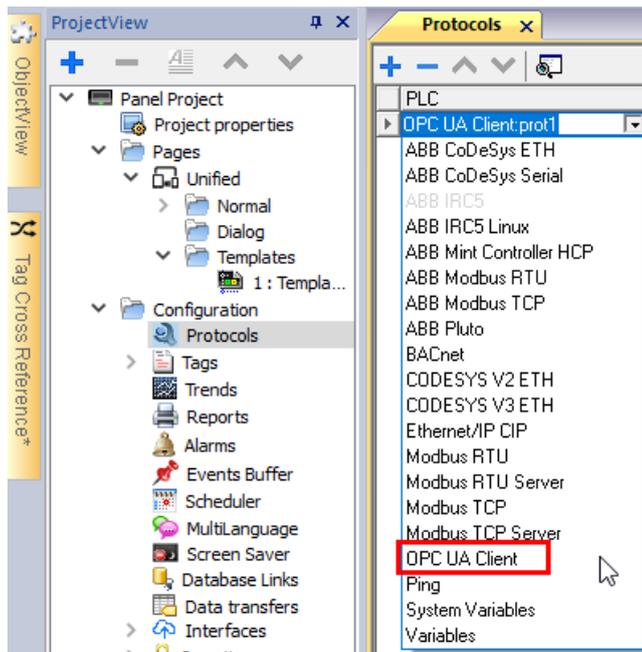
Click the 'Launch Panel Builder Editor'
In the Pop-up Windows, select 'New' and click 'Open' to continue.



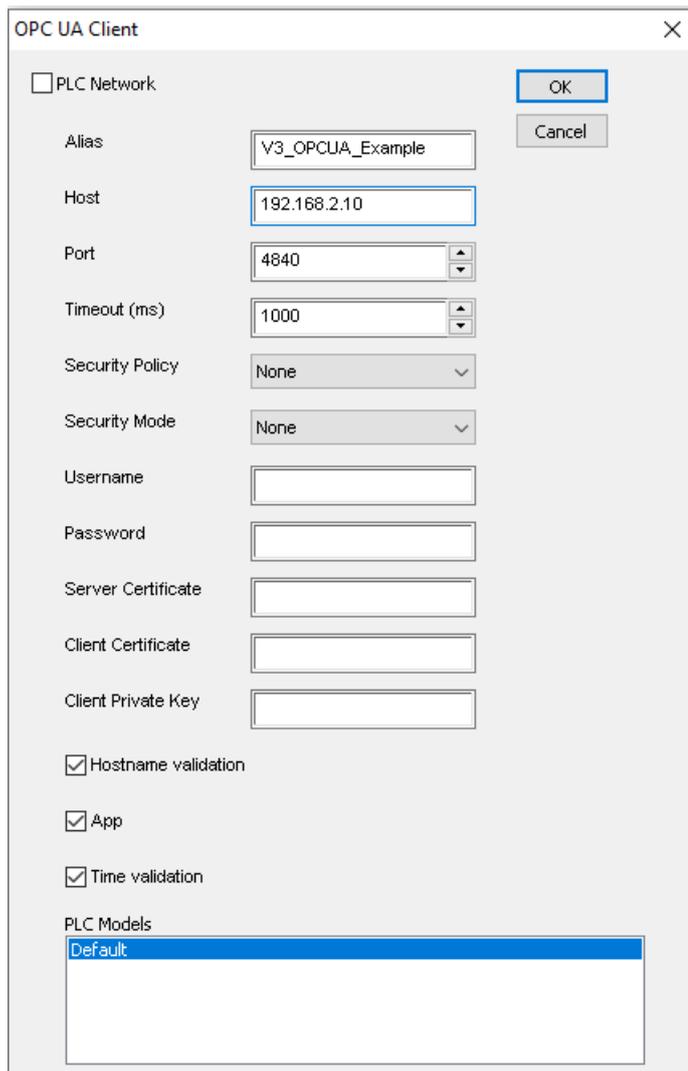
At the project wizard, select the CP6607 panel and click 'Finish' to continue.



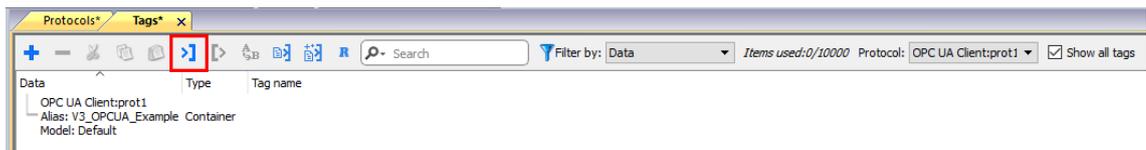
Double-click on the "Protocols" to open the editor.
Click on the "+" to add the protocol and select "OPC UA Client" in the dropdown list.



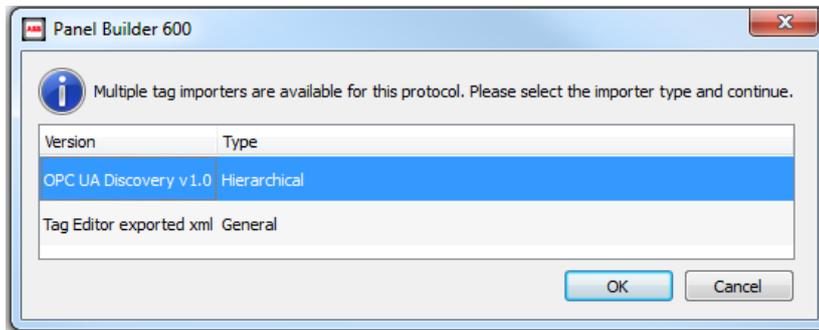
At the OPC UA Client Windows, type in the “Alias” and “IP address” as below. Leave the Port unchanged.



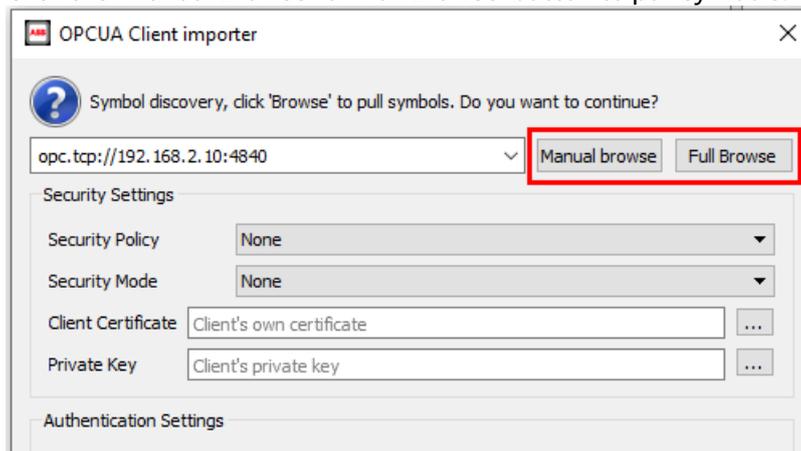
Click “Ok” button to continue and save the project.
Double click on the “Tags” to open the editor, then click on the “Import tags” button.



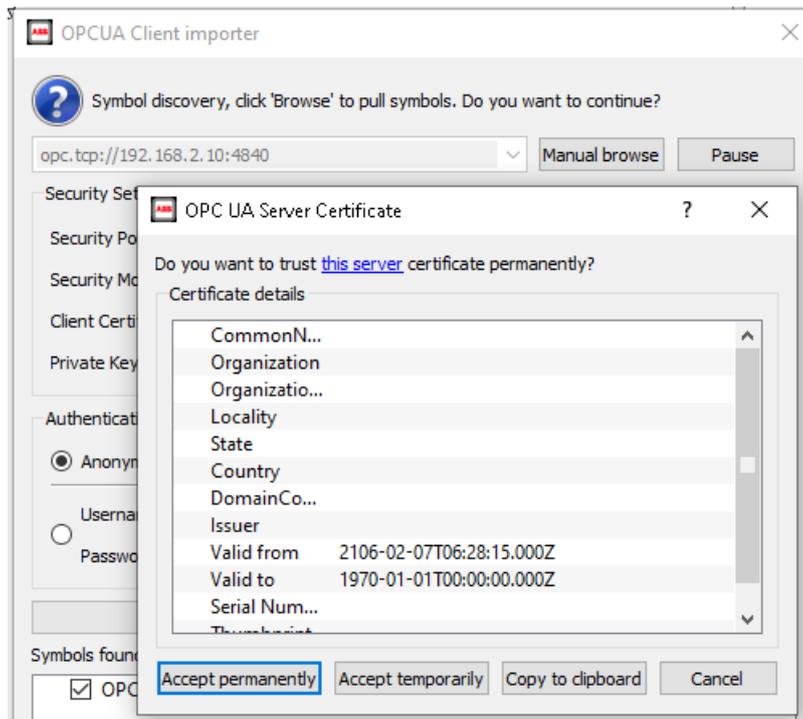
Select the OPC UA Discovery v1.0 and click “Ok” button to continue.



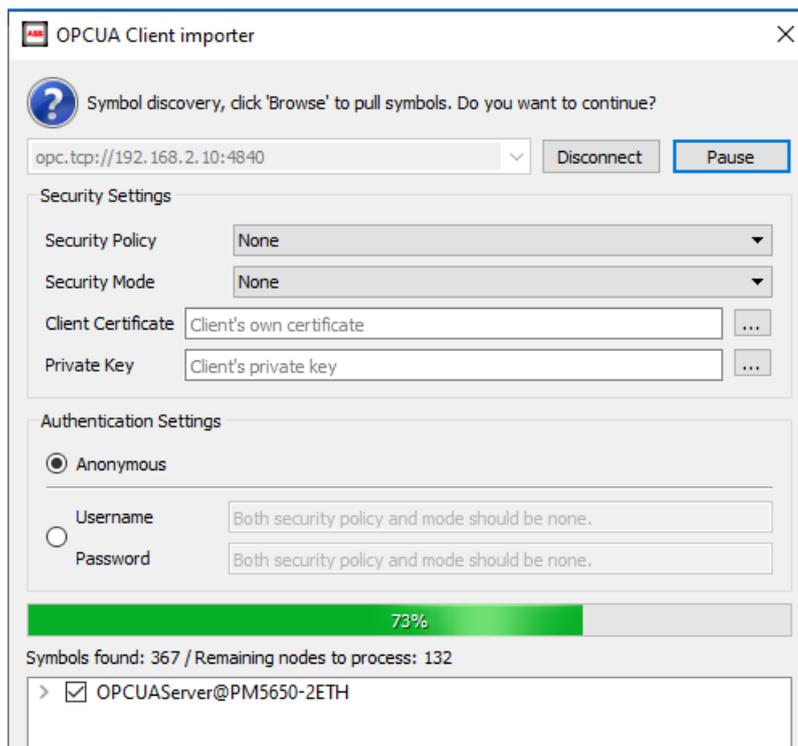
Click the “Manual Browse” or ‘Full Browse’ button to pull symbols.



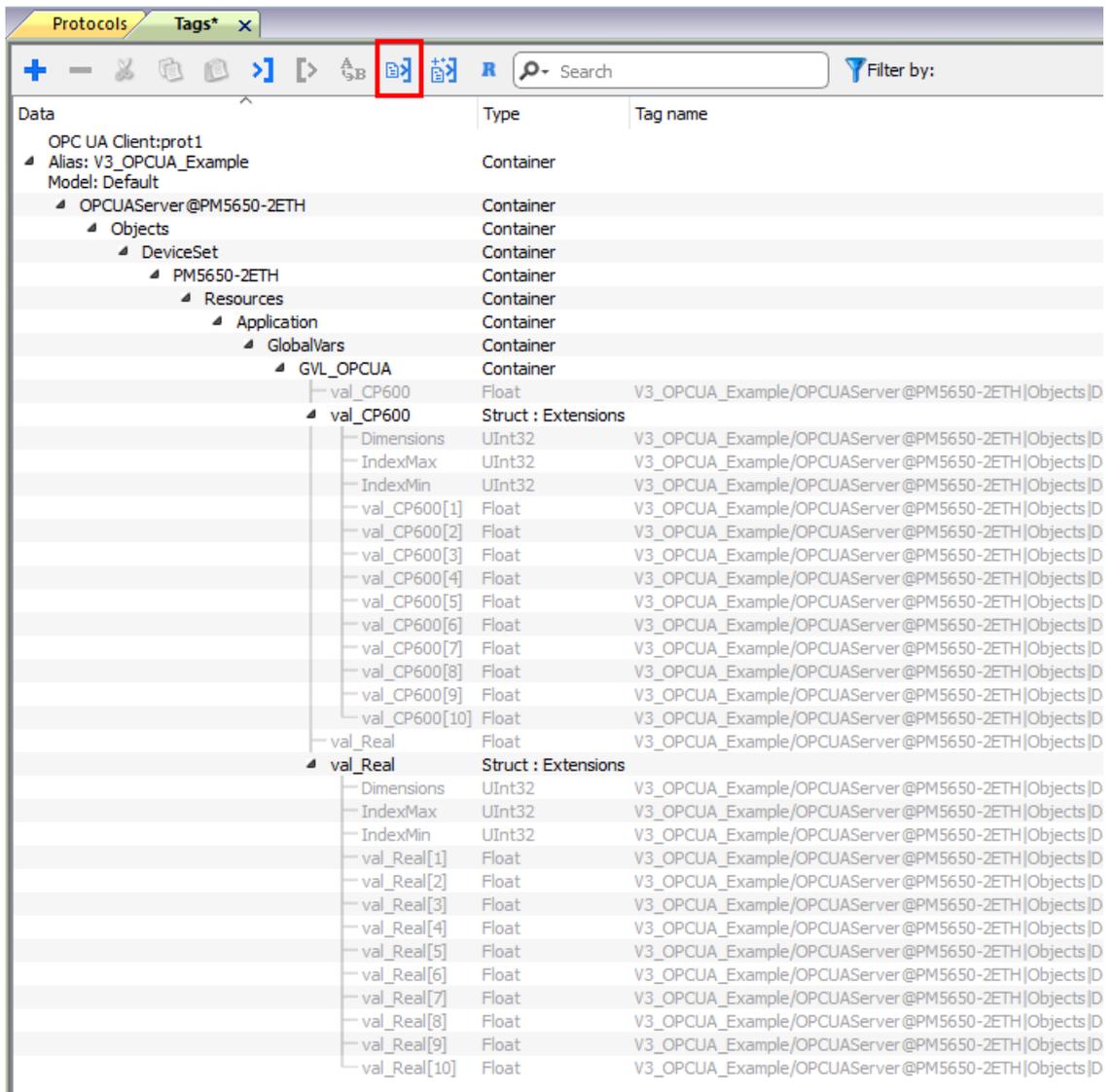
Click the ‘Accept temporarily’ button to trust OPC UA server certificate.



Wait for it to scan the network.



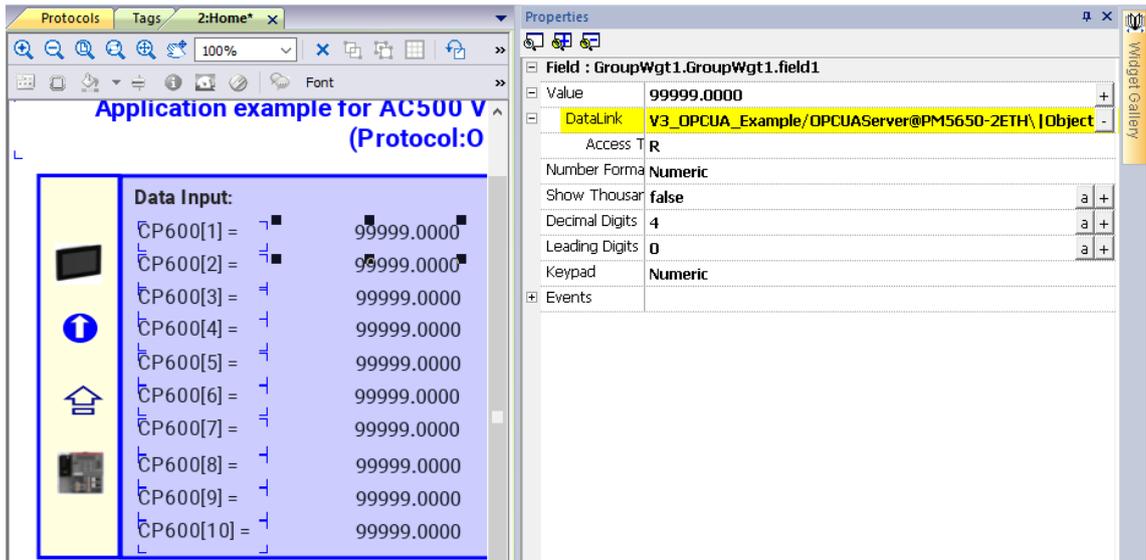
You can select all tags or only the tags you need. Then click "Ok" to continue.



At the 'Home', Draw the graphic as below.

   	Data Input:	CP600[1] =	99999.0000	   	Data Output:	AC500[1] =	99999.0000
	CP600[2] =	99999.0000	AC500[2] =		99999.0000		
	CP600[3] =	99999.0000	AC500[3] =		99999.0000		
	CP600[4] =	99999.0000	AC500[4] =		99999.0000		
	CP600[5] =	99999.0000	AC500[5] =		99999.0000		
	CP600[6] =	99999.0000	AC500[6] =		99999.0000		
	CP600[7] =	99999.0000	AC500[7] =		99999.0000		
	CP600[8] =	99999.0000	AC500[8] =		99999.0000		
	CP600[9] =	99999.0000	AC500[9] =		99999.0000		
	CP600[10] =	99999.0000	AC500[10] =		99999.0000		

After that link the tags for the value field in the table.



Repeat the same for others.
When complete, download the project to the CP6607 panel.

04/28/22 - 10:36:08

Application example for AC500 V3/CP600 - OPC UA without Certificate (Protocol:OPC UA Client)

Data Input:		Data Output:	
CP600[1] =	238796.3281	AC500[1] =	25.2300
CP600[2] =	129103.7031	AC500[2] =	888.0000
CP600[3] =	357511.5312	AC500[3] =	444.0000
CP600[4] =	407182.4062	AC500[4] =	254.0000
CP600[5] =	566053.3750	AC500[5] =	452.2200
CP600[6] =	814373.0000	AC500[6] =	555.0000
CP600[7] =	905402.5625	AC500[7] =	547.0000
CP600[8] =	328119.9688	AC500[8] =	21.0000
CP600[9] =	248520.0000	AC500[9] =	45.0000
CP600[10] =	238798.2344	AC500[10] =	365.2400



Attention:
Please note that this example is designed to show the general usage of the functionality.
They are not designed for a safe and complete implementation in a field application.

Stand: V4.0.1.462
(Release)
Date: 28.04.2022

Online with Automation Builder to observe the data values.

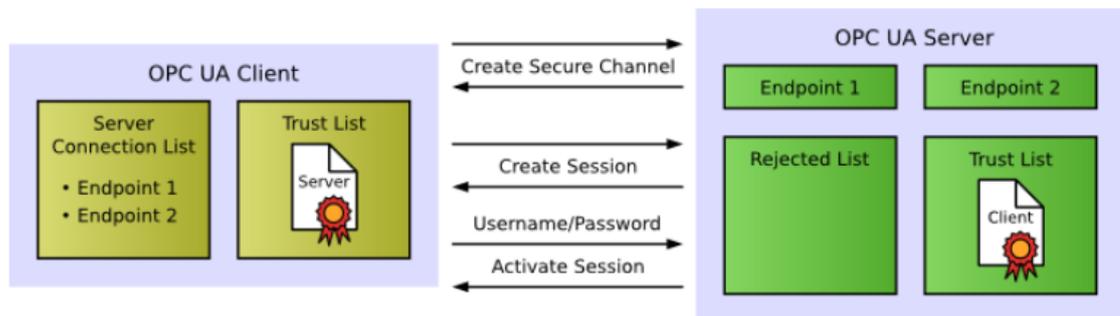
The screenshot shows the SIMATIC Manager interface. On the left, the 'Devices' tree is expanded to show the configuration of a CP600 device. The 'Application [run]' folder is selected, showing various components like 'Diagnosis', 'GVL OPCUA', and 'Task Configuration'. On the right, a table displays the values for various expressions in the 'PLC_AC500_V3.Application.GVL OPCUA' application.

Expression	Type	Value
val_Real	ARRAY [1..10] OF REAL	
val_Real[1]	REAL	184446.359
val_Real[2]	REAL	99811.19
val_Real[3]	REAL	276339.5
val_Real[4]	REAL	314716.875
val_Real[5]	REAL	437589.781
val_Real[6]	REAL	629433.75
val_Real[7]	REAL	699287.063
val_Real[8]	REAL	253338.938
val_Real[9]	REAL	192050
val_Real[10]	REAL	184448.266
val_CP600	ARRAY [1..10] OF REAL	
val_CP600[1]	REAL	25.23
val_CP600[2]	REAL	888
val_CP600[3]	REAL	444
val_CP600[4]	REAL	254
val_CP600[5]	REAL	452.22
val_CP600[6]	REAL	555
val_CP600[7]	REAL	547
val_CP600[8]	REAL	21
val_CP600[9]	REAL	45
val_CP600[10]	REAL	365.24

9.3 Establishing an Encrypted Connection of CP600 OPCUA Client to an OPC UA Server AC500 V3

The OPC UA server communicates with connected OPC UA clients over a separate TCP connection. Therefore, these connections have to be examined again separately with regard to security.

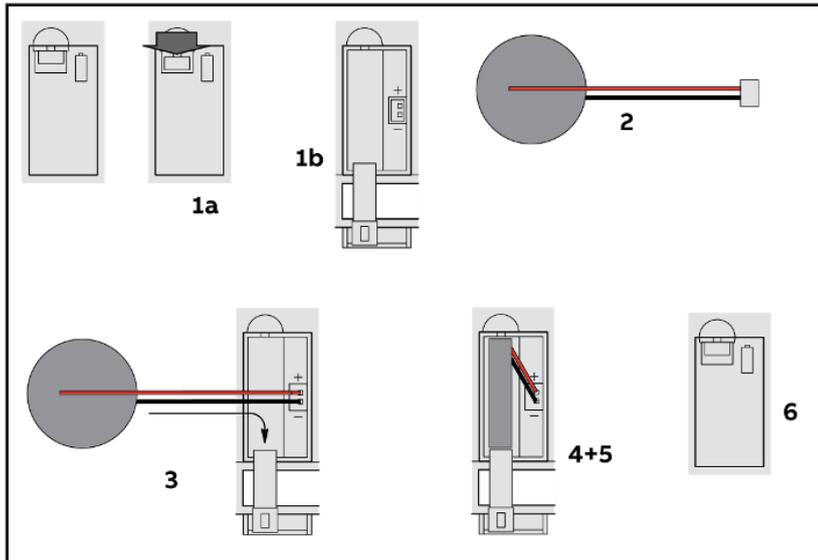
This sub-chapter is a step by step example that explains how to configure CP600 device to communicate with AC500 V3 using self-signed certificates.



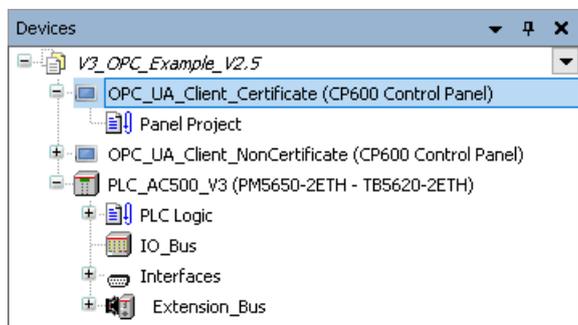
9.3.1 Creating a certificate for the OPC UA Server AC500 V3

In order to encrypt data and exchange with the client safely, the server needs a certificate that the client must classify as 'Trusted' when a connection is established for the first time.

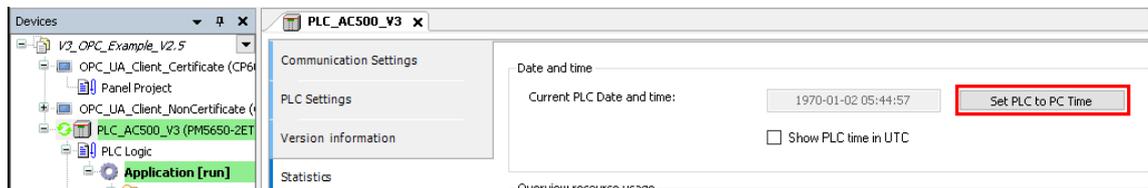
Prerequisite, battery is installed in the AC500 V3, and the clock is set to actual time.



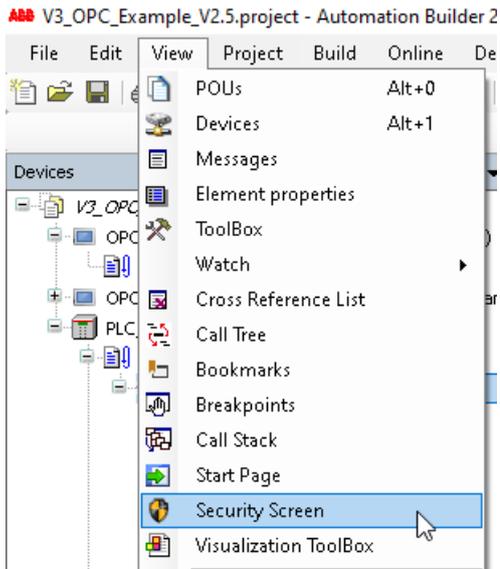
To create a simple project or directly start the example created above chapter.



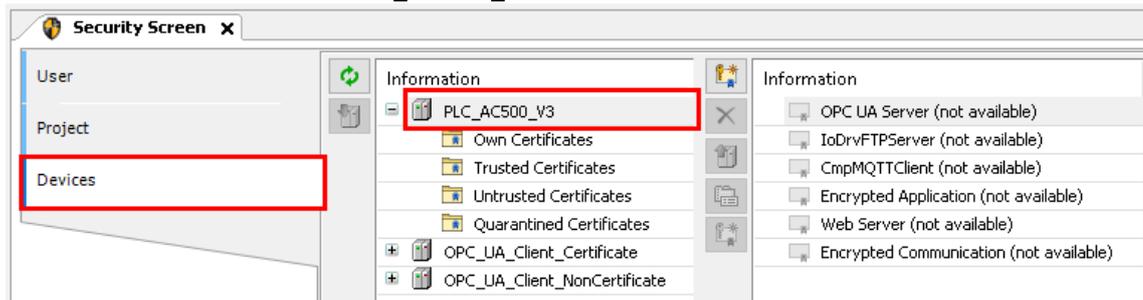
Login the PLC and push the 'Set PLC to PC Time' button from option Statistics to set time with PC time.



Click Menu View and open the security screen (or double click to Security screen button).

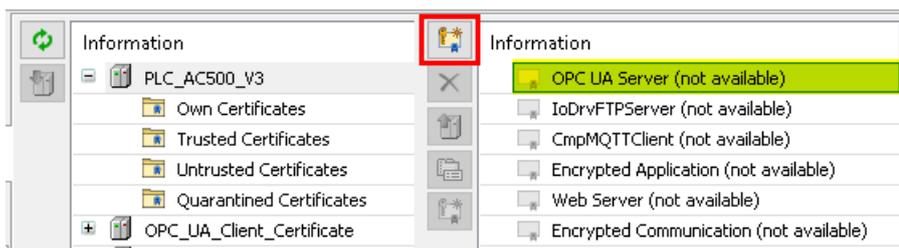


Select the Devices tab and PLC_AC500_V3 in the left view.

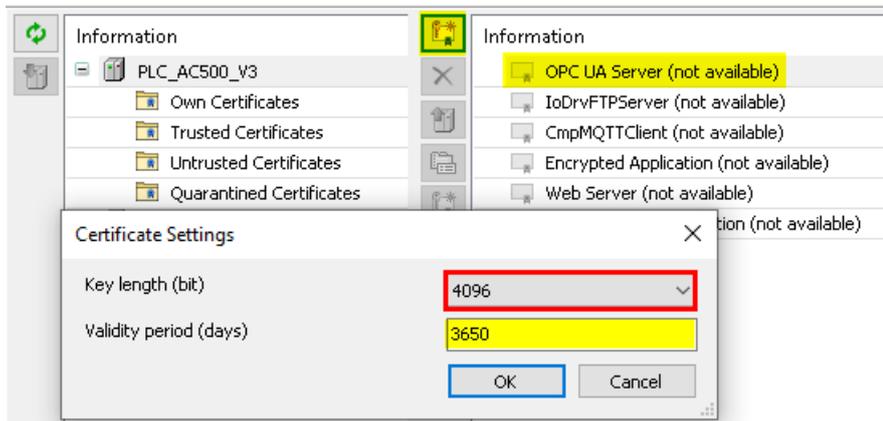


All services of the PLC that require a certificate are displayed in the right view.

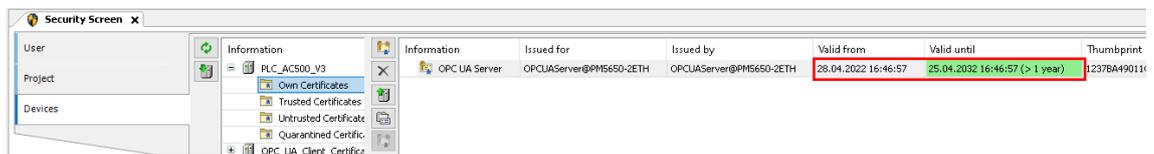
Select the service 'OPC UA Server(not available)' and click the Icon  to create a new certificate for the device.



The Certificate Settings dialog opens



Define the certificate parameters (4096 as Key Length) and click OK to close the dialog. The certificate is created on the PLC.





NOTICE Keep the online connection to the device before clicking OK button.
Certificate creation will last about 2-3 minutes.

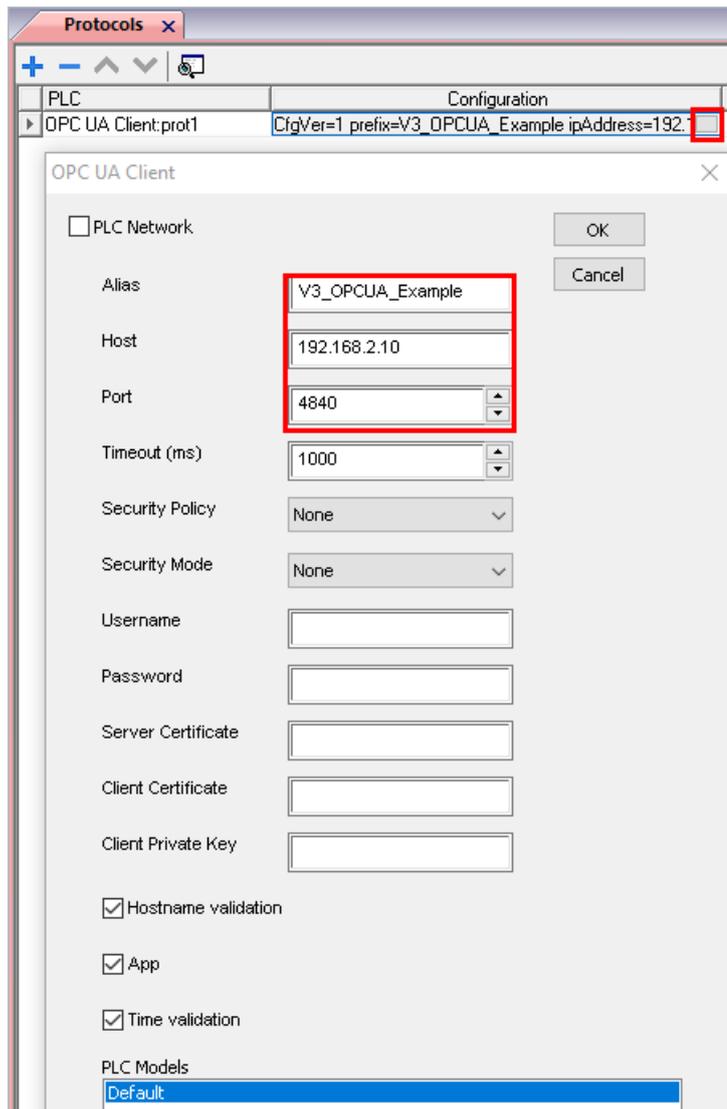
When complete, reboot the AC500 V3 PLC (Power 'OFF' and 'ON' the system)!

9.3.2 Setting up an encrypted connection with the CP600 OPC UA Client

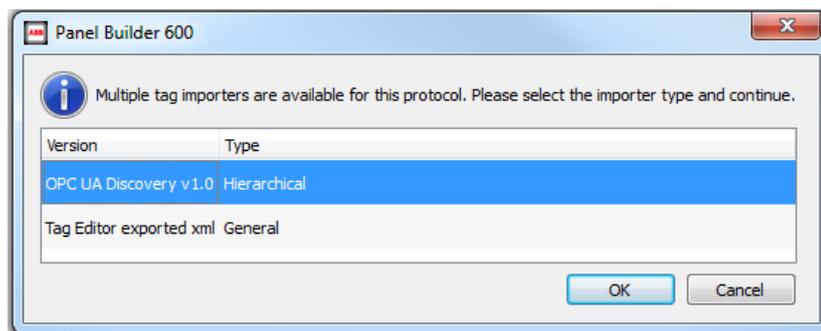
The following describe using CP600 as OPC UA client and connect to the AC500 V3 OPC UA server. Other OPC UA clients should work in a similar way.

Create a simple CP600 Panel project

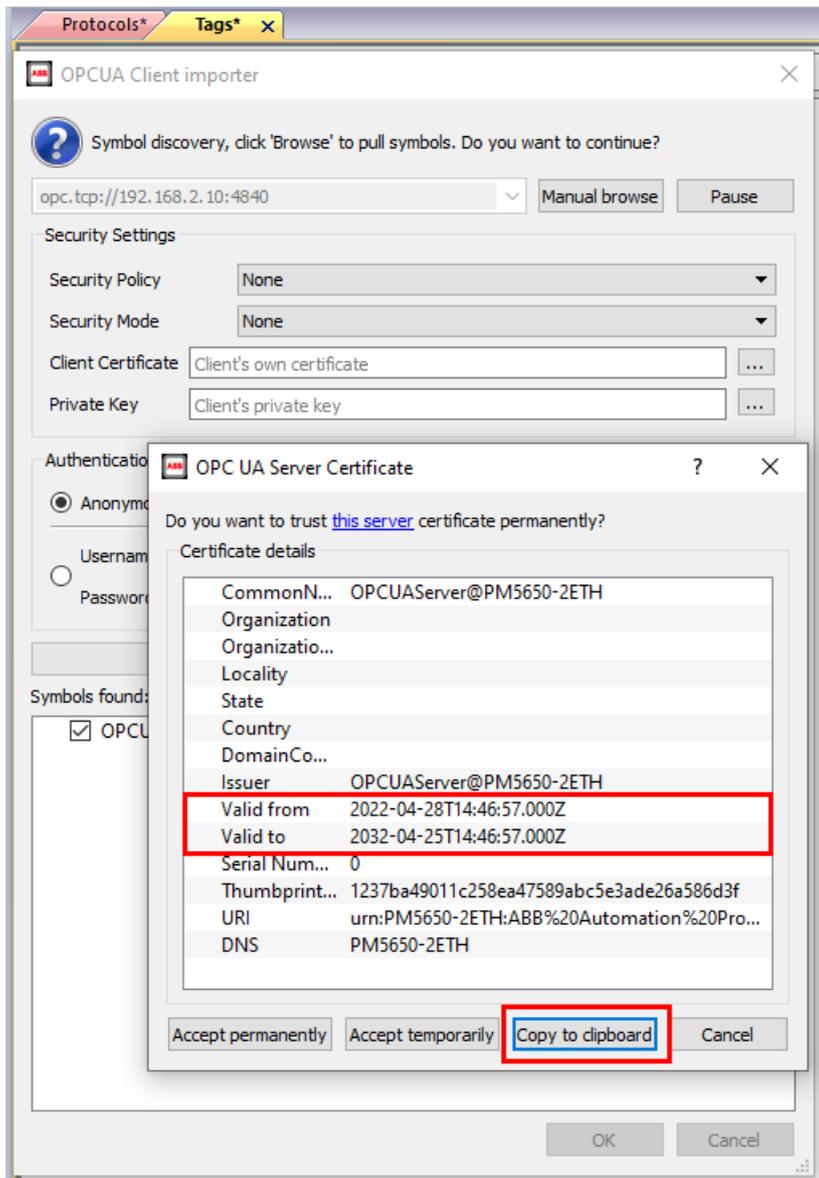
Add the OPC UA Client protocol. Enter the IP address of the remote OPC UA server and its port number (4840 for V3 PLC). Leave certificate parameters empty.



Open tag editor and import tags. Select 'OPC UA Discovery v1.0' mode



Choose the button 'Copy to clipboard' to copy the certificate to the clipboard as shown in the figure.



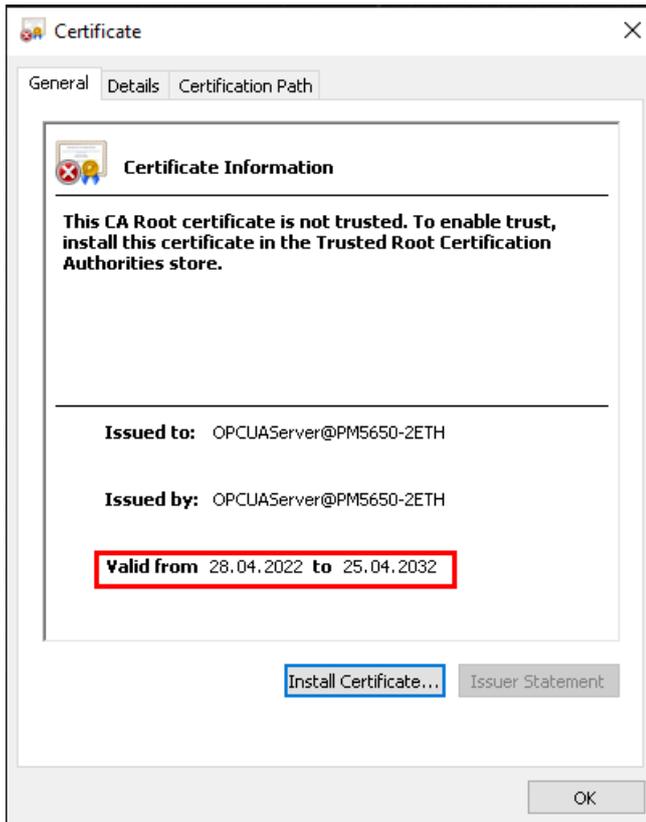
 **NOTICE** Please be note that the valid time of the certificate must be the same as the time set in the OPC UA Server (V3 PLC), otherwise, the communication connection will not be established. The reason for the time inconsistency may be that you forget to restart the system after setting the OPC UA Server certificate, or other reasons. e.g. 'parameter settings..'

Then, close this dialog and return to protocol configuration dialog to paste the certificate inside the “Server Certificate” field.

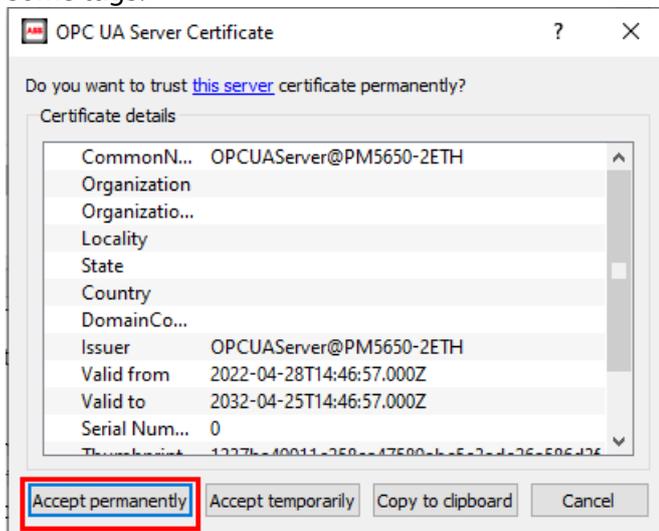
The image shows a dialog box titled "OPC UA Client" with a close button (X) in the top right corner. The dialog contains several configuration fields and checkboxes. The "Server Certificate" field is highlighted with a red rectangle and contains the text "MIFpTCCA42gAwIBAgI". Other fields include "Alias" (V3 OPCUA_Example), "Host" (192.168.2.10), "Port" (4840), "Timeout (ms)" (1000), "Security Policy" (None), "Security Mode" (None), "Username", "Password", "Client Certificate", and "Client Private Key". There are also checkboxes for "Hostname validation", "App", and "Time validation". At the bottom, there is a "PLC Models" section with a list box containing "Default". Buttons for "OK" and "Cancel" are located in the top right area.

<input type="checkbox"/> PLC Network	
Alias	V3 OPCUA_Example
Host	192.168.2.10
Port	4840
Timeout (ms)	1000
Security Policy	None
Security Mode	None
Username	
Password	
Server Certificate	MIFpTCCA42gAwIBAgI
Client Certificate	
Client Private Key	
<input checked="" type="checkbox"/> Hostname validation	
<input checked="" type="checkbox"/> App	
<input checked="" type="checkbox"/> Time validation	
PLC Models	Default

To verify a certificate, use a text editor to paste it from the clipboard to a text file with the extension .crt. You can then double click the .crt file to allow Windows to view the properties of certificate to be sure about the certificate you are validating.



Open tag editor and import tags again, accept the Server OPC UA certificate and import some tags.



Note that you can accept the certificate permanently or temporarily. If you accept the certificate permanently, a copy of the certificate will be saved inside your computer for later use without having the popup dialog asking for confirmation.

Data	Type	Tag name
OPC UA Client:prot1		
Alias: V3 OPCUA_Example	Container	
Model: Default		
OPCUAServer@PM5650-2ETH	Container	
Objects	Container	
DeviceSet	Container	
PM5650-2ETH	Container	
Resources	Container	
Application	Container	
GlobalVars	Container	
GVL OPCUA	Container	
val_CP600	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_CP600	Struct : Extensions	
Dimensions	UInt32	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
IndexMax	UInt32	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
IndexMin	UInt32	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_CP600[1]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_CP600[2]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_CP600[3]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_CP600[4]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_CP600[5]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_CP600[6]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_CP600[7]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_CP600[8]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_CP600[9]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_CP600[10]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_Real	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_Real	Struct : Extensions	
Dimensions	UInt32	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
IndexMax	UInt32	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
IndexMin	UInt32	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_Real[1]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_Real[2]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_Real[3]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_Real[4]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_Real[5]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_Real[6]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_Real[7]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_Real[8]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_Real[9]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl
val_Real[10]	Float	V3 OPCUA_Example/OPCUAServer@PM5650-2ETH Objects DeviceSet PM5650-2ETH Resources Appl

After that, link the tags for the value field in the table.



NOTICE

The certificate file will be copied to the folder:
%AppData%\Roaming\ABB\studio\OPCUA\pki\trusted\certs

Open again the protocol dialog box. Select the Security Policy = Basic256Sha256, and Security Mode = SignAndEncrypt.

Be sure to uncheck the box of the 'Hostname validation' option.

OPC UA Client

PLC Network

OK Cancel

Alias: V3_OPCUA_Example

Host: 192.168.2.10

Port: 4840

Timeout (ms): 1000

Security Policy: Basic256Sha256

Security Mode: SignAndEncrypt

Username:

Password:

Server Certificate: MlIFpTCCA42gAwIBAgIBAl

Client Certificate:

Client Private Key:

Hostname validation

App

Time validation

PLC Models

Default

NOTICE Username and Password creation please refer to the application note 'AC500 V3 and OPC UA - Configuration and Handling.pdf', which can be downloaded from the PLC website <https://new.abb.com/plc/documentsanddownloads>

Download the project to the CP600 device.

Download to Target

copying /protocols/protocolFileList.xml

192.168.2.80

Download Close

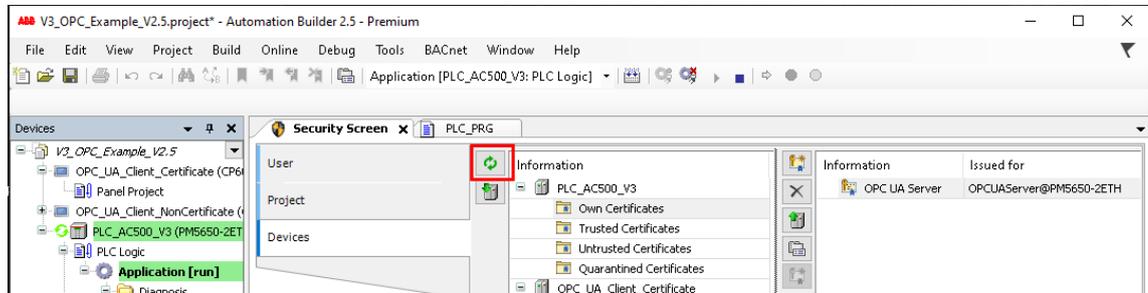
+ Advanced

In the OPC UA Client protocol parameters, we left the “Client certificate” field empty, the OPC UA Client protocol will generated its own certificate and sent it to the OPC UA Server

(AC500 V3). Since the server does not recognize the certificate, therefore it rejects the connection request.

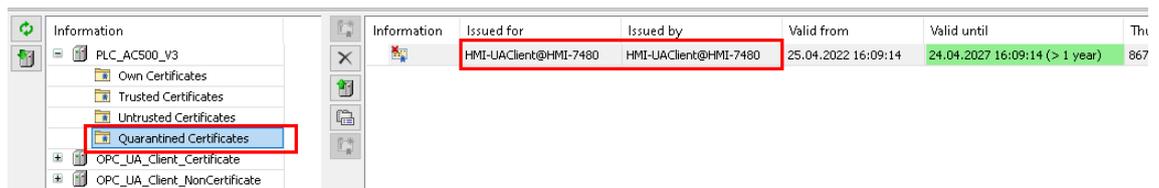
Now, we need to tell the server to trust these certificates.

In Automation builder development system, reopen the Security Screen and click the  symbol button. The view is refreshed.

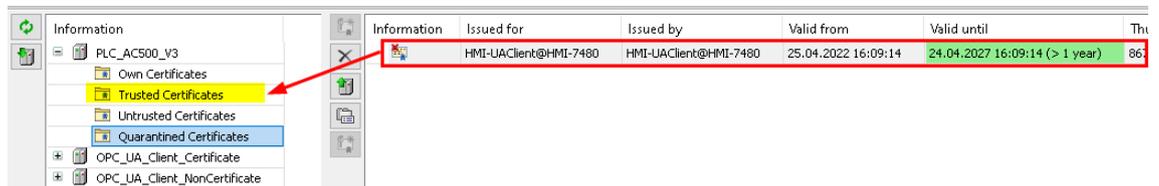


Select the certificate folder Quarantined Certificates.

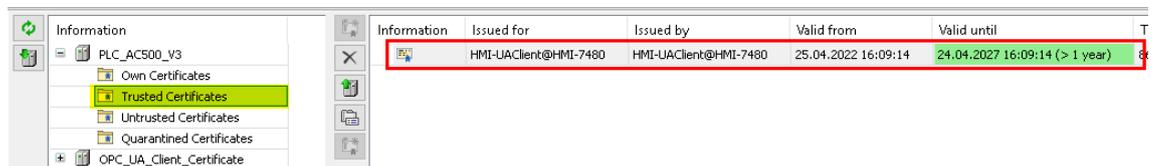
The client certificate HMI-UAClient@HMI-7480 is displayed in the right view, including the valid time information, etc.



Drag the certificate to the certificate folder Trusted Certificates.



Now the client certificate is classified by the server as trusted.



Now the communication will start

Application example for AC500 V3/CP600 - OPC UA with Certificate (Protocol:OPC UA Client)

	Data Input:		Data Output:
	CP600[1] = 900607.0000		AC500[1] = 55.3000
	CP600[2] = 491190.6250		AC500[2] = 24.4000
	CP600[3] = 1360537.8750		AC500[3] = 21.5000
	CP600[4] = 1549313.6250		AC500[4] = 165.2500
	CP600[5] = 2154058.5000		AC500[5] = 55.0000
	CP600[6] = 3098627.2500		AC500[6] = 0.0000
	CP600[7] = 3415600.7500		AC500[7] = 25.6420
	CP600[8] = 1242278.7500		AC500[8] = 0.0000
	CP600[9] = 941262.5000		AC500[9] = 45.3000
	CP600[10] = 900608.8750		AC500[10] = 99.0000



Attention:
Please note that this example is designed to show the general usage of the functionality.
They are not designed for a safe and complete implementation in a field application.

Stand: V4.0.1.462
(Release)
Date: 28.04.2022

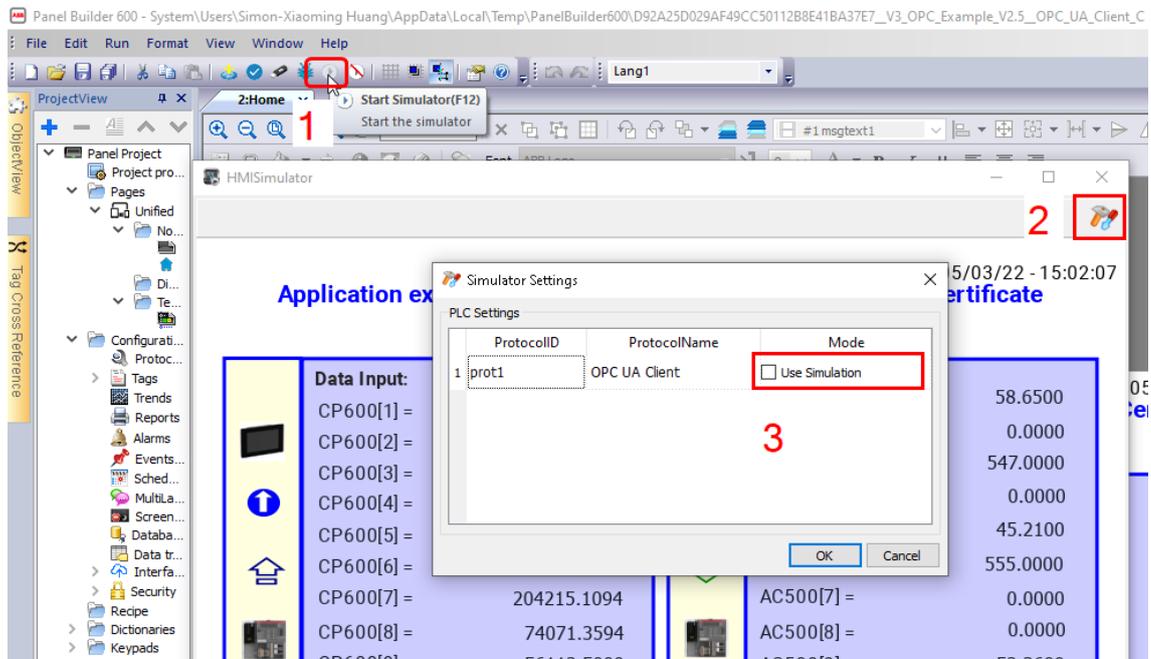
Online with the Automation Builder to observe the data value.

Security Screen PLC_PRG GVL OPCUA x		
PLC_AC500_V3.Application.GVL OPCUA		
Expression	Type	Value
[-] val_Real	ARRAY [1..10] OF REAL	
val_Real[1]	REAL	910439.5
val_Real[2]	REAL	496623.063
val_Real[3]	REAL	1375541.75
val_Real[4]	REAL	1566387
val_Real[5]	REAL	2177857.75
val_Real[6]	REAL	3132774
val_Real[7]	REAL	3452842.75
val_Real[8]	REAL	1255727.25
val_Real[9]	REAL	951607.5
val_Real[10]	REAL	910441.4
[-] val_CP600	ARRAY [1..10] OF REAL	
val_CP600[1]	REAL	55.3
val_CP600[2]	REAL	24.4
val_CP600[3]	REAL	21.5
val_CP600[4]	REAL	165.25
val_CP600[5]	REAL	55
val_CP600[6]	REAL	0
val_CP600[7]	REAL	25.642
val_CP600[8]	REAL	0
val_CP600[9]	REAL	45.3
val_CP600[10]	REAL	99

9.3.3 Setting up an encrypted connection with the CP600 HMI simulator

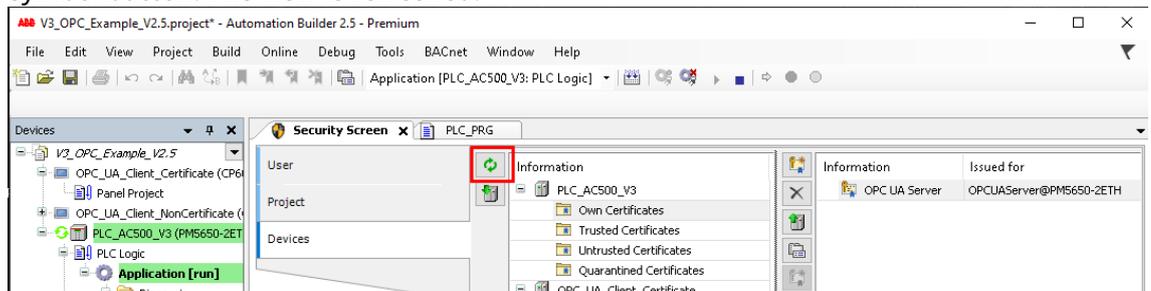
The Simulator works by default with simulated protocols. It can also work with real protocols in case we don't have the CP600 hardware.

At CP600 project, launch the simulator and click on the 'Simulator Settings' icon. Then remove the option 'Use Simulation' to use the real protocols for communication with external devices.



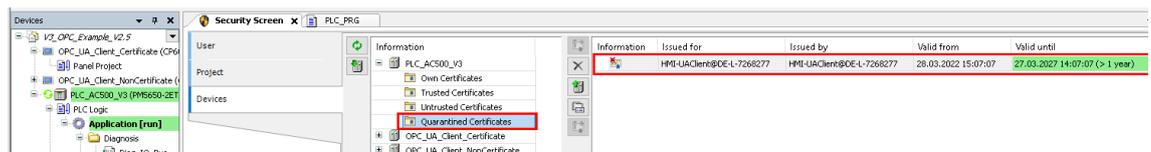
Now we have to tell the OPC UA server (V3 PLC) to trust these certificates.

In Automation builder development system, reopen the Security Screen and click the  symbol button. The view is refreshed.



Select the certificate folder Quarantined Certificates.

The client certificate HMI-UAClient@DE-L-7268277(PC Name) is displayed in the right view, including the valid time information, etc.



Drag this certificate to the certificate folder Trusted Certificates

Information	Issued for	Issued by	Valid from	Valid until
HMI-UAClient@DE-L-7268277	HMI-UAClient@DE-L-7268277	HMI-UAClient@DE-L-7268277	28.03.2022 15:07:07	27.03.2027 14:07:07 (> 1 year)
HMI-UAClient@HMI-7480	HMI-UAClient@HMI-7480	HMI-UAClient@HMI-7480	25.04.2022 16:09:14	24.04.2027 16:09:14 (> 1 year)

Now the communication will start

HMISimulator 05/03/22 - 15:12:58

Application example for AC500 V3/CP600 - OPC UA with Certificate (Protocol:OPC UA Client)

Data Input:	Data Output:
CP600[1] = 210371.2969	AC500[1] = 58.6500
CP600[2] = 113783.7188	AC500[2] = 0.0000
CP600[3] = 315058.5625	AC500[3] = 547.0000
CP600[4] = 358822.9375	AC500[4] = 0.0000
CP600[5] = 498866.9062	AC500[5] = 45.2100
CP600[6] = 717654.0625	AC500[6] = 555.0000
CP600[7] = 797608.9375	AC500[7] = 0.0000
CP600[8] = 288989.4062	AC500[8] = 0.0000
CP600[9] = 218987.5000	AC500[9] = 52.3600
CP600[10] = 210373.2031	AC500[10] = 55.3000

ABB Attention: Please note that this example is designed to show the general usage of the functionality. They are not designed for a safe and complete implementation in a field application.

Stand: V4.0.1.462 (Release) Date: 28.04.2022

Online with Automation Builder to observe the data value.

Expression	Type	Value	Prepared
val_Real	ARRAY [1..10] OF REAL		
val_Real[1]	REAL	222534.891	
val_Real[2]	REAL	120339.422	
val_Real[3]	REAL	333224.969	
val_Real[4]	REAL	379516.844	
val_Real[5]	REAL	527617.25	
val_Real[6]	REAL	759033.7	
val_Real[7]	REAL	843726.7	
val_Real[8]	REAL	305730.781	
val_Real[9]	REAL	231622.5	
val_Real[10]	REAL	222536.8	
val_CP600	ARRAY [1..10] OF REAL		
val_CP600[1]	REAL	58.65	
val_CP600[2]	REAL	0	
val_CP600[3]	REAL	547	
val_CP600[4]	REAL	0	
val_CP600[5]	REAL	45.21	
val_CP600[6]	REAL	555	
val_CP600[7]	REAL	0	
val_CP600[8]	REAL	0	
val_CP600[9]	REAL	52.36	
val_CP600[10]	REAL	55.3	

**NOTICE**

When an OPC UA Client attempts a connection with the OPC UA Server, the server checks if the client certificate is available inside its own trusted certificate list. If it is not found, the communication will be rejected and the certificate will be stored in a list of unreliable certificates.

10 Connection AC500 V3 High Availability and OPC DA Client



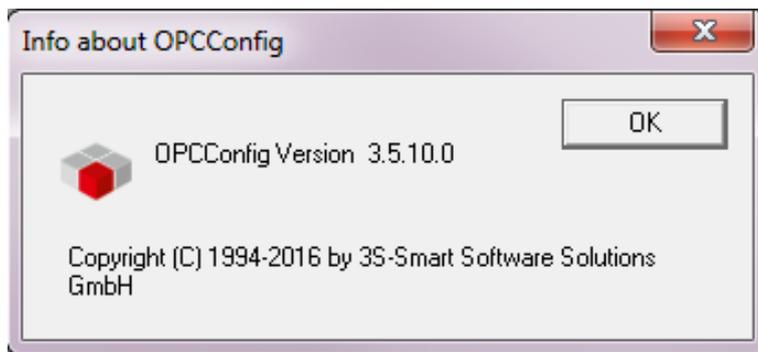
NOTICE

For more details how to program High Availability (HA) functionalities using Modbus TCP, refer to “AC500 High Availability - HA-Modbus TCP V3 Library Example Description 3ADR025289M0208.pdf”

Start Automation Builder and open the V3 HA Example project “V3_HA_ModbusTCP_Example_AB220.project” (location as with all AC500 library examples folder: C:\Users\Public\Documents\AutomationBuilder\Examples\PS5601-HA-MTCP\AC500_V3

10.1 Requirements

The OPC server V3 (version 3.5.10.0 or higher) must be installed from the Automation Builder 2.2.0. Please refer to the installation description.



By default, the OPC Server (WinCoDeSysOPC.exe) runs in the same user account of the process that launched the OPC Server. Depending on the user account this could lead to unpredictable behavior. Thus, it's required to configure the user account for the OPC server.



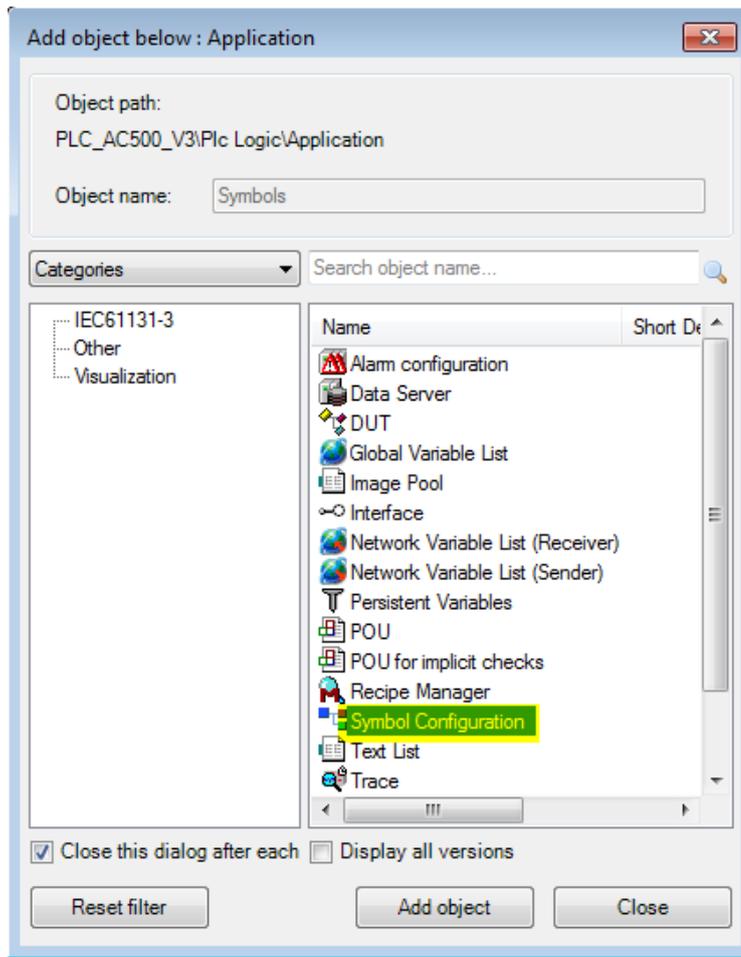
NOTICE

The OPC server can generally also be registered as service.

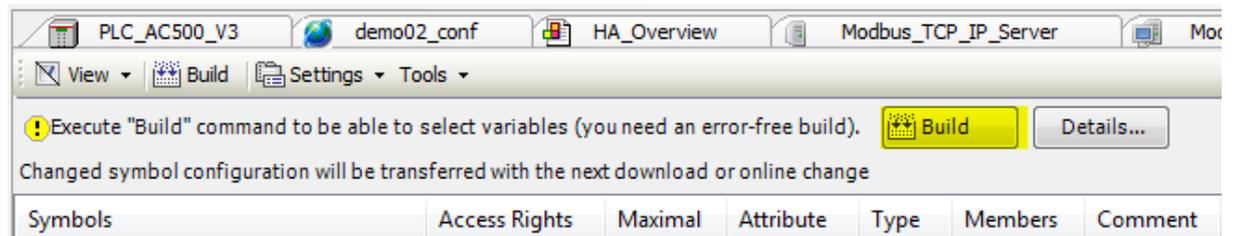
10.2 Configure and download symbol file

Symbol includes the items (variables) which exchanges data with PLC, this is needed for OPC communication. The symbol file can be used to check if it is generated correctly.

To add Symbol Configuration, Right click on 'Application' node in the Automation Builder device tree and select add object and select Symbol configuration (Project-> Add Object->Symbol configuration). The symbol configuration editor appears.

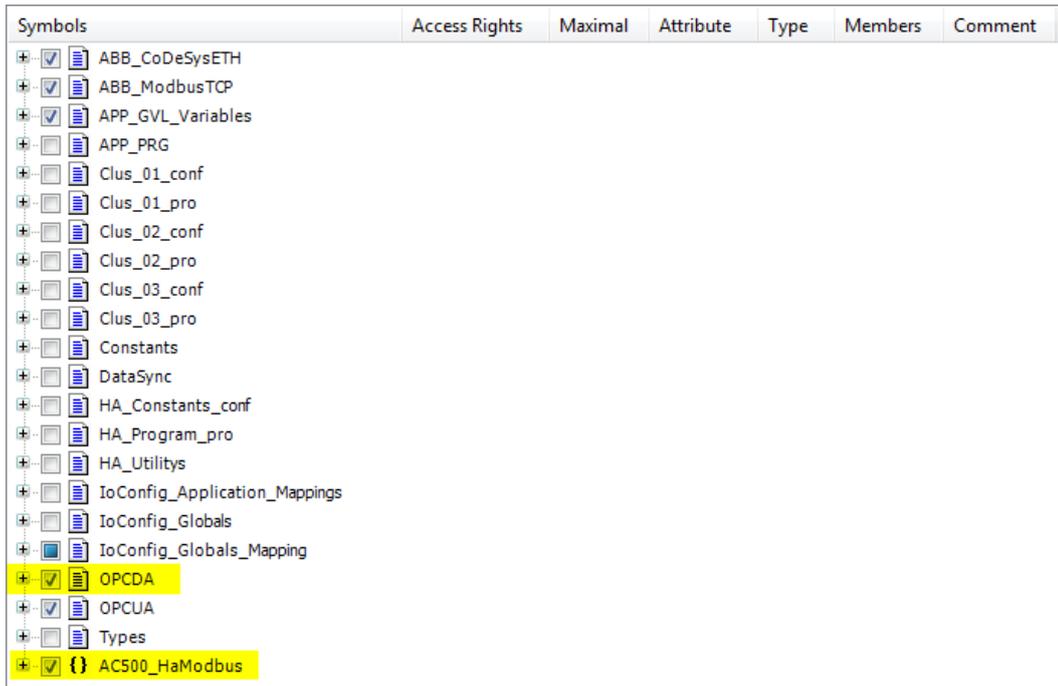


In Symbol Configuration editor, execute 'Build' command to enable variables selection (you need an error-free build).



Then you will see all Items (variables) in the window and select the desired Items which exchanges data with PLC, these are needed for OPC communication.

Only choose the variables which should be communicated and the HA Global variables from HA lib as symbol. e.g. In the example, we choose name 'OPCDA', 'APP_GVL_Variables' and 'HA_Global_Variables' for the OPC Items dedicated communication.

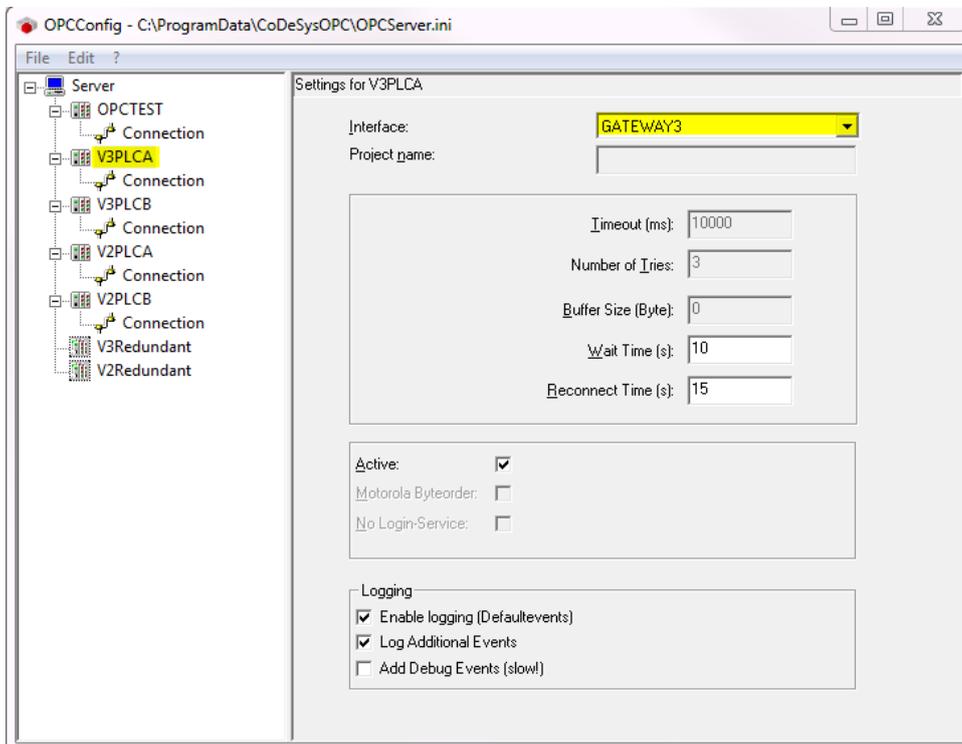


Go to Online-> Login, with commanding Login the application will be compiled and downloaded.

10.3 OPC Config

To start the OPC configuration start the application “OPCConfig.exe” from the installation directory (Windows ->Start ->3S CODESYS -> CODESYS OPC Server V3 -> OPC Configurator).

The inserted PLC will be presented in the left window. The configuration will be done at the PLC node and the underlying Connection node.



For creating a new communication setup or for modifying an existing one, use button ‘Edit’ to open the dialog ‘CoDeSys address of PLC’:

The 'LifeCounter Variable' must only be set once to the value < Application.AC500_HAModbus.HA_GLOBAL_VARIABLES.dwHaModServerAlive >.



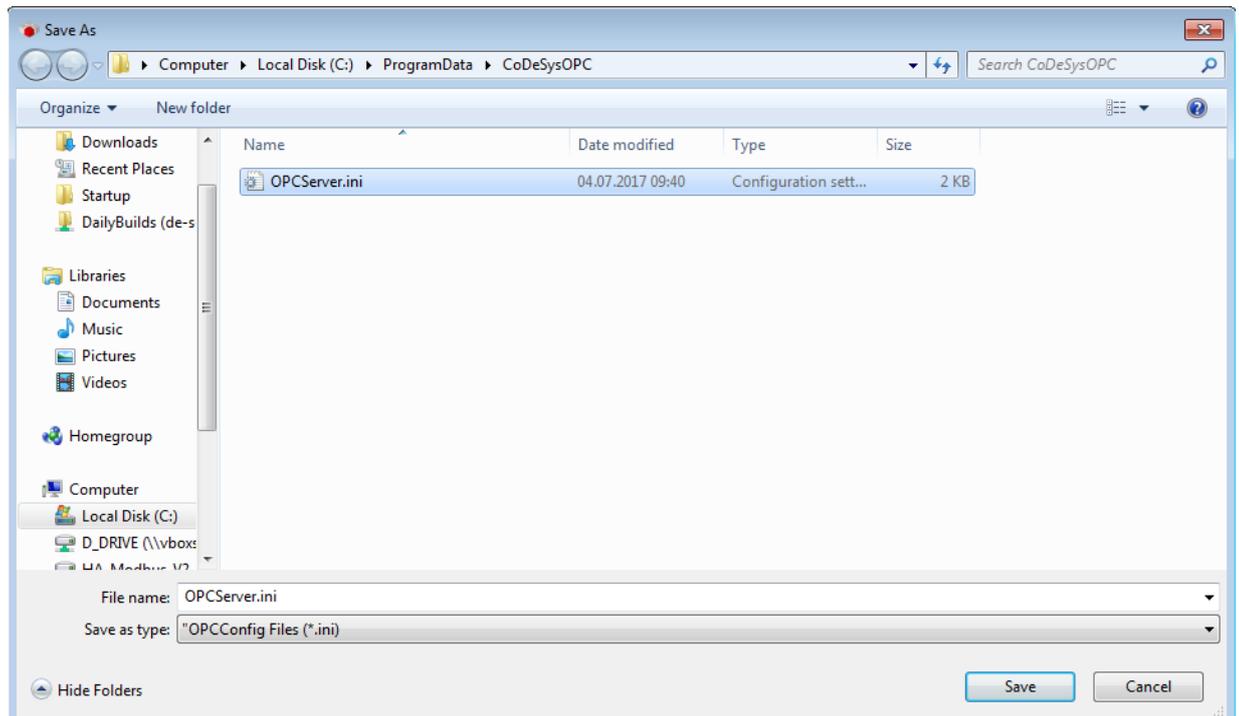
NOTICE

The 'Master Variable' must be set for both PLC's to the value:
Application.AC500_HAModbus.HA_GLOBAL_VARIABLES.xHaModPrimary.

The 'LifeCounter Variable' must be set the value:
Application.AC500_HAModbus.HA_GLOBAL_VARIABLES.dwHaModServerAlive.

If one setting is missing or misspelled (case sensitive) the OPC Server will not work properly.

After OPC configuration, please save as "OPCServer.ini" into folder "C:\ProgramData\CoDeSysOPC".



NOTICE

The OPC configuration must be saved as <OPCServer.ini> at the OPC server execution directory.



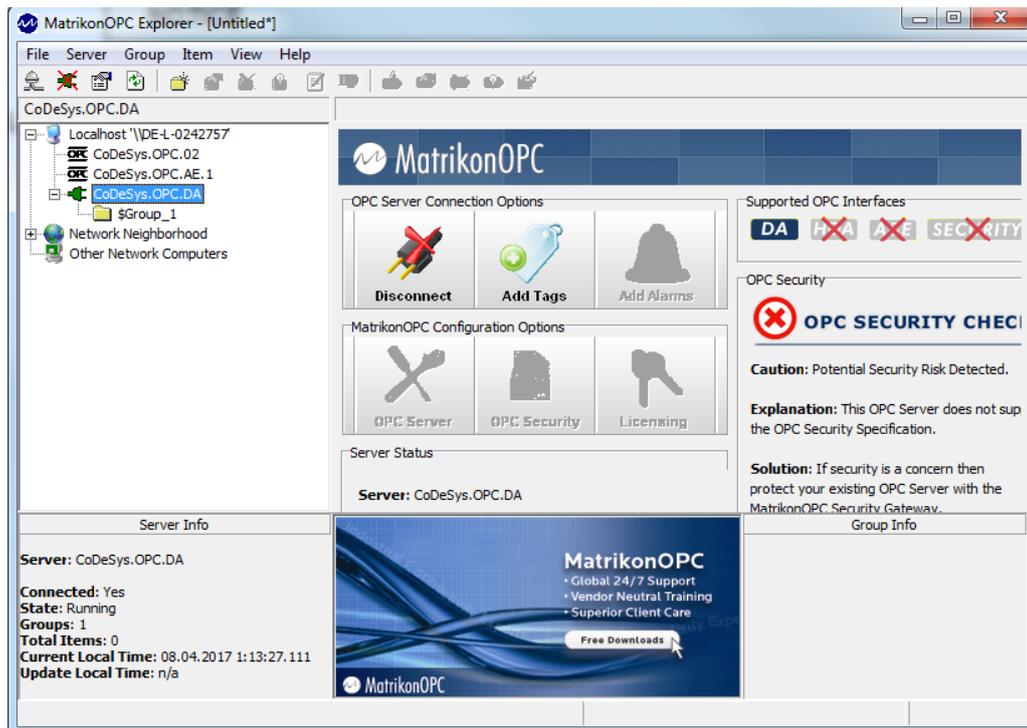
NOTICE

Each change of the OPC configuration will only come to action if the OPC server is restarted after saving the configuration as <OPCServer.ini> at the OPC server execution directory.

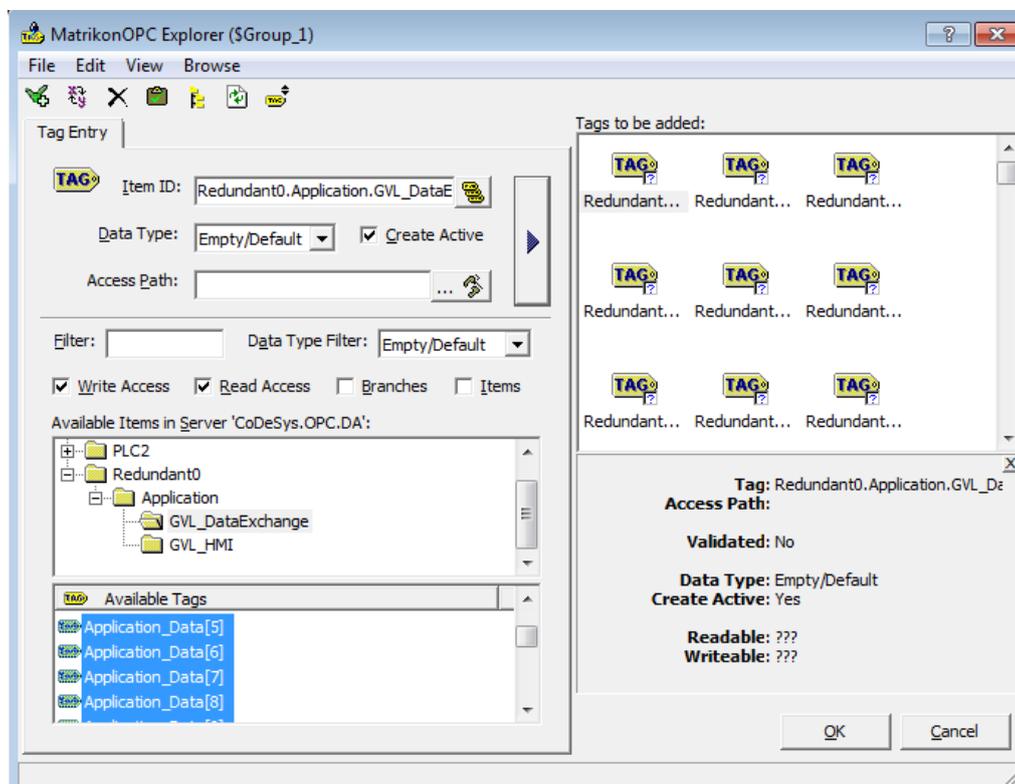
For more details on the configuration of OPC DA, refer to the online help file or AC500 Application description.

10.4 Check OPC Server with MatrikonOPCEXplorer

The OPC configuration can be made and tested with any OPC test Client, e.g. MatrikonOPCEXplorer.



Connect CoDeSys.OPC.DA, Add Group→Add Tags→Select Available Tags→Add to Tag list...



If the OPC Server V3 (CoDeSys.OPC.DA) is connected and running, the quality is good. One OPC Client can read/write the values of the items.

The screenshot shows the MatrikonOPC Explorer interface. The main window displays a tree view on the left with the following structure:

- Localhost \DE-1-0242757
 - OPC CoDeSys.OPC.02
 - OPC CoDeSys.OPC.AE.1
 - OPC CoDeSys.OPC.DA
 - \$Group_1
 - Network Neighborhood
 - Other Network Computers

The main pane shows the 'Contents of '\$Group_1'' with the following table:

Item ID	Access ...	Value	Quality	Timestamp	Status
Redundant0...App_Data[10]		7045634	Good, non-specific	08.04.201...	Active
Redundant0...App_Data[11]		7045634	Good, non-specific	08.04.201...	Active
Redundant0...App_Data[12]		7045634	Good, non-specific	08.04.201...	Active
Redundant0...App_Data[13]		7045634	Good, non-specific	08.04.201...	Active
Redundant0...App_Data[14]		7045634	Good, non-specific	08.04.201...	Active
Redundant0...App_Data[15]		7045634	Good, non-specific	08.04.201...	Active
Redundant0...App_Data[16]		7045634	Good, non-specific	08.04.201...	Active
Redundant0...App_Data[17]		7045634	Good, non-specific	08.04.201...	Active
Redundant0...App_Data[18]		7045634	Good, non-specific	08.04.201...	Active
Redundant0...App_Data[19]		7045634	Good, non-specific	08.04.201...	Active
Redundant0...App_Data[1]		7045634	Good, non-specific	08.04.201...	Active
Redundant0...App_Data[20]		7045634	Good, non-specific	08.04.201...	Active
Redundant0...App_Data[2]		7045634	Good, non-specific	08.04.201...	Active
Redundant0...App_Data[3]		7045634	Good, non-specific	08.04.201...	Active
Redundant0...App_Data[4]		7045634	Good, non-specific	08.04.201...	Active
Redundant0...App_Data[5]		7045634	Good, non-specific	08.04.201...	Active
Redundant0...App_Data[6]		7045634	Good, non-specific	08.04.201...	Active

A 'Write Values' dialog box is open, showing the 'Multiple Value' tab. It contains the following table:

Item ID	Current Value	Data Type	New Value
<input checked="" type="checkbox"/> Redundant0...CP6 0		Single Float	45

The status bar at the bottom shows the following information:

- Server: CoDeS
- Connected: Ye
- State: Running
- Groups: 1
- Total Items: 144
- Current Local
- Update Local

Group Info for '\$Group_1':

- Connected (Async I/O): Yes (2.0)
- Active: Yes
- Items: 144
- Current Update Rate: 1000 ms
- Percent Deadband: 0,00%
- Data Change Rate: 32,06 Items/Sec

11 Connection AC500 V3 High Availability and OPC UA Client



NOTICE

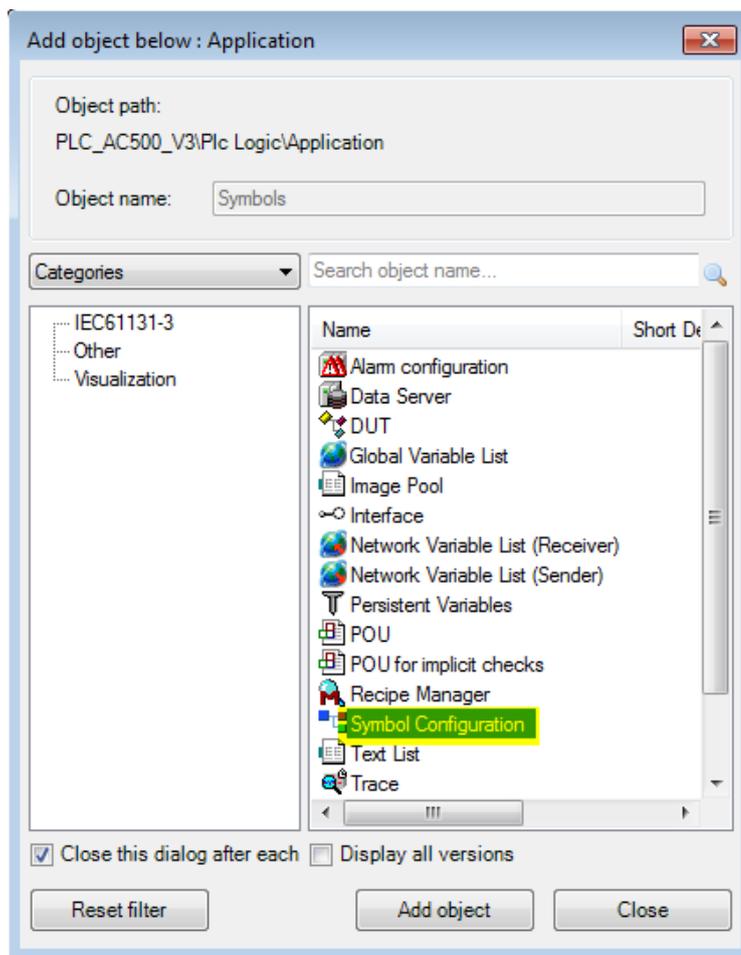
For more details how to program High Availability (HA) functionalities using Modbus TCP (including the CP600 project), refer to “AC500 High Availability - HA-Modbus TCP V3 Library Example Description 3ADR025289M0205.pdf”

Start Automation Builder and open the V3 HA Example project “V3_HA_ModbusTCP_Example_AB220.project” (location as with all AC500 library examples folder: C:\Users\Public\Documents\AutomationBuilder\Examples\PS5601-HA-MTCP\AC500_V3

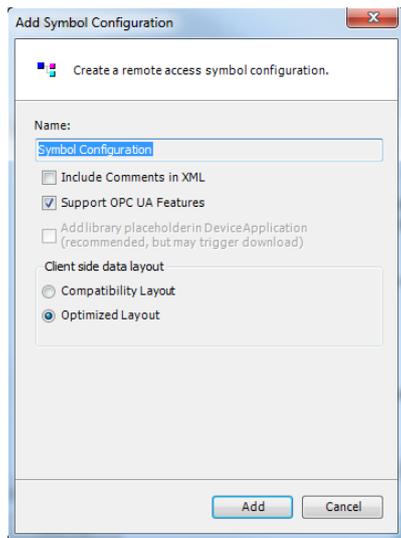
11.1 Configure and download symbol file

Symbol includes the items (variables) which exchanges data with PLC, this is needed for OPC communication. The symbol file can be used to check if it is generated correctly.

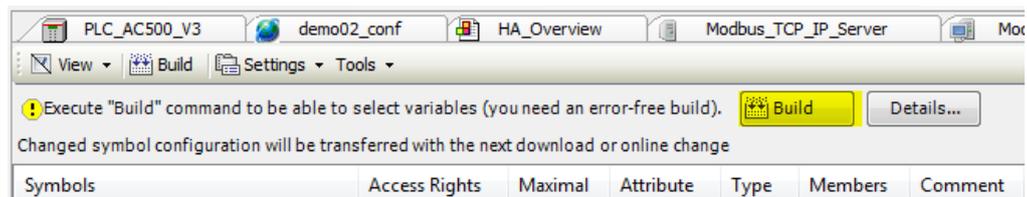
To add Symbol Configuration, Right click on ‘Application’ node in the Automation Builder device tree and select add object and select Symbol configuration (Project-> Add Object->Symbol configuration). The symbol configuration editor appears.



A new “Add Symbol Configuration” windows will pop up.
 Check the “Support OPC UA Features” box and click “Add” button to continue.

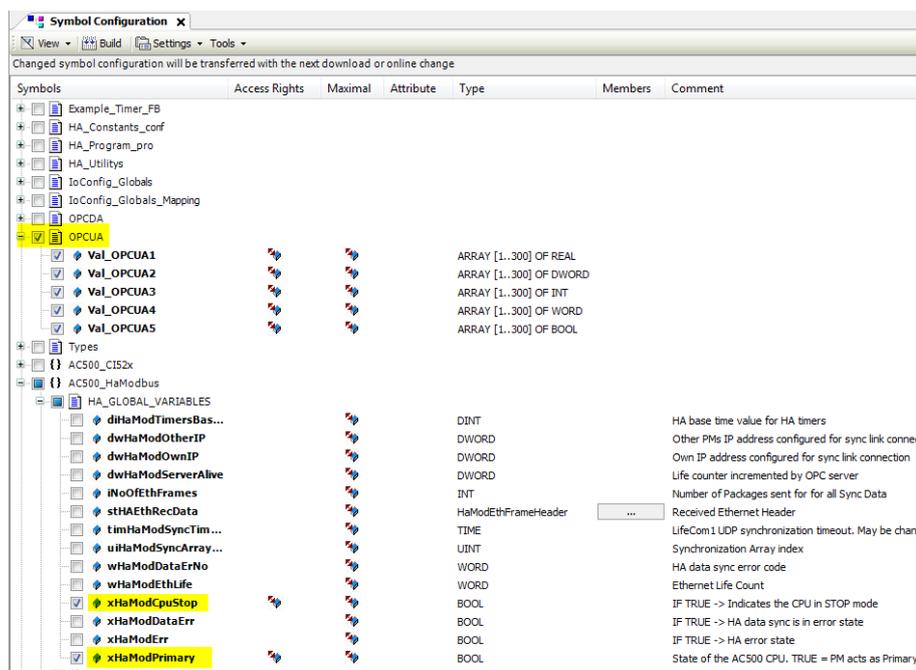


In Symbol Configuration editor, execute ‘Build’ command to enable variables selection (you need an error-free build).



Then you will see all Items (variables) in the window and select the desired Items which exchanges data with PLC, these are needed for OPC communication.

ONLY choose the variables which should be communicated as symbol. e.g. In the example, we choose name ‘OPCUA for the OPC Items dedicated communication.

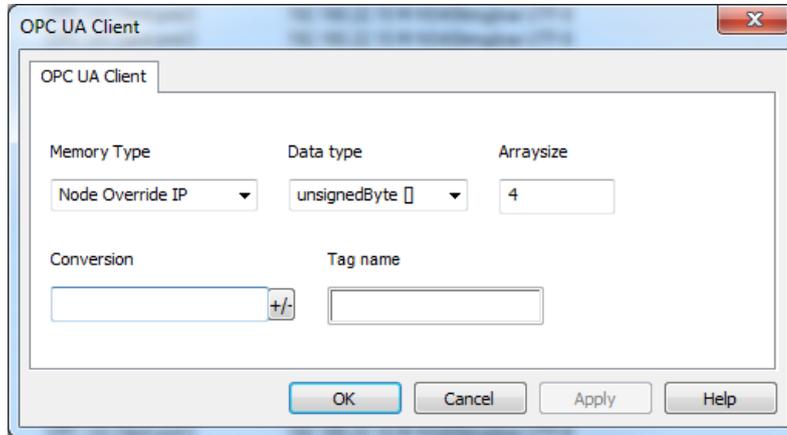


Go to Online-> Login, with commanding Login the application will be compiled and downloaded.

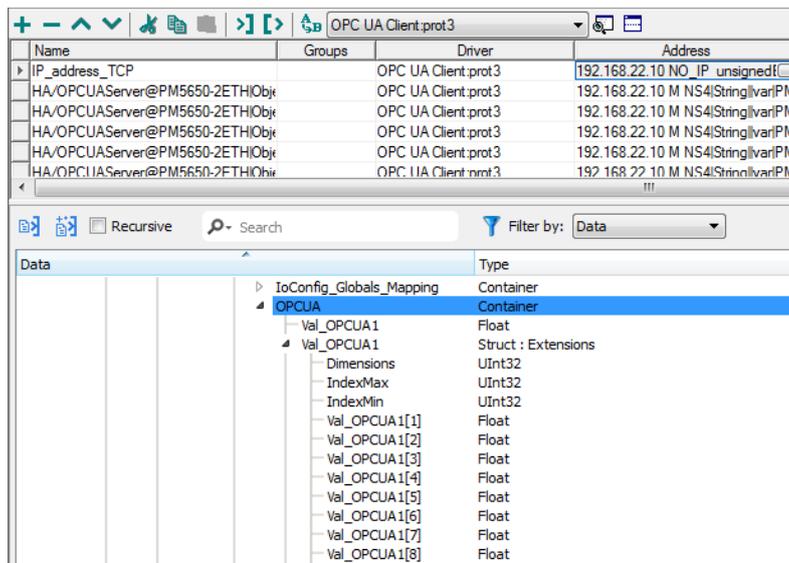
11.2 Check OPC Server with ABB Panel CP600

The CP600 panel configured with OPC UA Client provides the special data type Node Override IP which allows you to change the IP address of the target controller at runtime.

This memory type is an array of 4 unsigned bytes, one per each byte of the IP address. The Node Override IP is initialized with the value of the controller IP specified in the project at programming time.



Select the tags you want and click on the “Import Tags” button.



When complete, download the project to the panel.





ABB AG
Eppelheimer Straße 82
69123 Heidelberg, Germany
Phone: +49 62 21 701 1444
Fax: +49 62 21 701 1382
E-Mail: plc.support@de.abb.com
www.abb.com/plc

We reserve the right to make technical changes or modify the contents of this document without prior notice. With regard to purchase orders, the agreed particulars shall prevail. ABB AG does not accept any responsibility whatsoever for potential errors or possible lack of information in this document.

We reserve all rights in this document and in the subject matter and illustrations contained therein. Any reproduction, disclosure to third parties or utilization of its contents – in whole or in parts – is forbidden without prior written consent of ABB AG.
Copyright© 2023 ABB. All rights reserved