YASKAWA

VIPA SPEED7

SP7-OPC | SPEED7 OPC UA Configurator | Manual

HB50 | SP7-OPC | SPEED7 OPC UA Configurator | en | 19-24

Software manual SPEED7 OPC UA Configurator - V1.8.6

www.vipa.com/en/service-support/manuals

VIPA CONTROLS

VIPA GmbH
Ohmstraße 4
91074 Herzogenaurach
Telephone: 09132-744-0
Fax: 09132-744-1864
Email: info@vipa.com
Internet: www.vipa.com

1 General information on this documentation

1.1 Copyright © VIPA GmbH

All Rights Reserved

This document contains proprietary information of VIPA and is not to be disclosed or used except in accordance with applicable agreements.

This material is protected by the copyright laws. It may not be reproduced, distributed, or altered in any fashion by any entity (either internal or external to VIPA), except in accordance with applicable agreements, contracts or licensing, without the express written consent of VIPA and the business management owner of the material.

For permission to reproduce or distribute, please contact: VIPA, Gesellschaft für Visualisierung und Prozessautomatisierung mbH Ohmstraße 4, D-91074 Herzogenaurach, Germany

Tel.: +49 9132 744 -0

Fax.: +49 9132 744-1864

EMail: info@vipa.de

http://www.vipa.com

Every effort has been made to ensure that the information contained in this document was complete and accurate at the time of publishing. Nevertheless, the authors retain the right to modify the information.

This customer document describes all the hardware units and functions known at the present time. Descriptions may be included for units which are not present at the customer site. The exact scope of delivery is described in the respective purchase contract.

EC Conformity Declaration

Hereby, VIPA GmbH declares that the products and systems are in compliance with the essential requirements and other relevant provisions. Conformity is indicated by the CE marking affixed to the product.

Conformity Information

For more information regarding CE marking and Declaration of Conformity (DoC), please contact your local VIPA customer service organization.

Trademarks

VIPA, SLIO, System 100V, System 200V, System 300V, System 300S, System 400V, System 500S and Commander Compact are registered trademarks of VIPA Gesellschaft für Visualisierung und Prozessautomatisierung mbH.

SPEED7 is a registered trademark of profichip GmbH.

SIMATIC, STEP, SINEC, TIA Portal, S7-300, S7-400 and S7-1500 are registered trademarks of Siemens AG.

Microsoft and Windows are registered trademarks of Microsoft Inc., USA.

Portable Document Format (PDF) and Postscript are registered trademarks of Adobe Systems, Inc.

All other trademarks, logos and service or product marks specified herein are owned by their respective companies.

Information product support

Contact your local VIPA Customer Service Organization representative if you wish to report errors or questions regarding the contents of this document. If you are unable to locate a customer service centre, contact VIPA as follows:

VIPA GmbH, Ohmstraße 4, 91074 Herzogenaurach, Germany

Telefax: +49 9132 744-1204

EMail: documentation@vipa.de

Technical support

Contact your local VIPA Customer Service Organization representative if you encounter problems with the product or have questions regarding the product. If you are unable to locate a customer service centre, contact VIPA as follows:

VIPA GmbH, Ohmstraße 4, 91074 Herzogenaurach, Germany

Tel.: +49 9132 744-1150 (Hotline)

EMail: support@vipa.de

1.2 Purpose of the documentation

This documentation describes the VIPA *SPEED7 OPC UA Configurator* software package.

The manual is intended for persons who implement control functions automation systems.

1.3 Validity of the documentation

This software description is valid for the *SPEED7 OPC UA Configurator* software package from version 1.8.6

1.4 Presentation and tags

Tips, recommendations and operating instructions are presented in this documentation as follows:

Tips and recommendations

ℹ *This icon refers to information which will facilitate the use of the system.*

Operating instructions

This documentation includes operating instructions for many functions which you can follow step by step. Operating instructions include the following elements:

→ Every operating step tells you what to do. The individual steps of any operating instruction consisting of several steps will be successively numbered.

⇨ Here, the result of the operating step is presented.

2 OPC UA

2.1 General

Term definitions

- OPC - **O**pen **P**latform **C**ommunications
  - *OPC* is an interoperability standard for secure and reliable data exchange in industrial automation.
  - *OPC* is platform-independent and ensures a seamless flow of information between devices from different manufacturers.
- UA - **U**nified **A**rchitecture
  - *UA* specifies security features and data modeling based on a service-oriented architecture (SOA).

Precondition

- VIPA *SPEED7 Studio* from Version V1.8.6
  - The functionality for the *OPC UA* configuration is integrated in the *SPEED7 Studio*.
- Siemens SIMATIC Manager from version V5.5 and VIPA *SPEED7 Studio* from version V1.8.6
  - The *OPC UA* configuration is done with the *OPC UA Configurator*. This is part of the *SPEED7 Studio* from VIPA from version V1.8.6.
  - When calling the *OPC UA Configurator*, the *SPEED7 Studio* opens with functionality limited to *OPC UA* configuration.
  - The *OPC UA Configurator* is to be called from the Siemens SIMATIC Manager as external device tool.
  - To be able to call the *OPC UA Configurator* as an external device tool, you must first register it in the Siemens SIMATIC Manager. This is done with *SPEED7 Tools Integration*, which is automatically installed during the installation of the *SPEED7 Studio*.
  - The *OPC UA Configurator* is to be called from the Siemens SIMATIC Manager after project creation and online configuration.
  - The *OPC UA Configurator* automatically imports the data for the *OPC UA* configuration from the project data of the Siemens SIMATIC Manager.
  - The *OPC UA* configuration is transferred online from the *OPC UA Configurator*. The *OPC UA Configurator* automatically imports the data for the *OPC UA* configuration from the project data of the Siemens SIMATIC Manager.
- Siemens TIA Portal from version V15.0 and VIPA *SPEED7 Studio* from version V1.8.6
  - The *OPC UA* configuration is done with the *OPC UA Configurator*. This is part of the *SPEED7 Studio* from VIPA from version V1.8.6.
  - When calling the *OPC UA Configurator*, the *SPEED7 Studio* opens with functionality limited to *OPC UA* configuration.
  - The *OPC UA Configurator* is to be called from the Siemens TIA Portal as external device tool.
  - To be able to call the *OPC UA Configurator* as an external device tool, you must first register it in the Siemens TIA Portal. This is done with *SPEED7 Tools Integration*, which is automatically installed during the installation of the *SPEED7 Studio*.
  - The *OPC UA Configurator* is to be called from the Siemens TIA Portal after project creation and online configuration.
  - The *OPC UA Configurator* automatically imports the data for the *OPC UA* configuration from the project data of the Siemens TIA Portal.
  - The *OPC UA* configuration is transferred online from the *OPC UA Configurator*. The *OPC UA Configurator* automatically imports the data for the *OPC UA* configuration from the project data of the Siemens TIA Portal.

2.2 Basics OPC UA

2.2.1 OPC UA

Standard for data and information exchange

OPC UA defines a common standard for data and information exchange in an *'Industry 4.0'* environment. Due to the platform independence, the integrated security concept and the data type information supplied with the data, *OPC UA* provides the basis for machine-readable and cross-level communication.

OPC - Open Platform Communications

- Classic variant not scalable and exclusively for Microsoft Windows®
- For each type of data transfer, such as real-time data, history data, interrupts, events, etc., a separate solution with its own semantics is required, such as OPC DA, OPC HDA, OPC A&E, etc.
- Separate and complex effort required for security settings.
- *OPC* requires a complex *DCOM* configuration.
- *OPC* requires separate complex firewall settings.

OPC UA - Open Platform Communications Unified Architecture

- Scalable and platform-independent communication standard specified in IEC 62541.
- Standardization of classic OPC specifications with integrated security concept.
- The *OPC UA* security concept includes user and application authentication, message signing, and encryption of transmitted data.
- IP-based, optimized, binary protocol for Internet and firewall communication via one port (4840).
- With *OPC UA*, any type of information is available at any time and place for any authorized application and authorized person. For example, raw data and preprocessed information can be transported safely from the sensor and field level to the control system and into the production planning systems.
- SOA (**S**ervice **O**riented **A**rchitecture) replaces the Microsoft *DCOM* technology with open, platform-independent protocols with integrated security mechanisms.
  - Communication takes place via standardized services based on the *Information model* of OPC UA.
  - The services are divided into different task groups.
  - Based on a basic model, arbitrarily complex, object-oriented extensions of the services can be carried out, without affecting interoperability.

OPC UA server

- An *OPC UA server* provides information within a network that can be retrieved from an OPC UA client.
- The data exchange can take place via security certificates, which have to be stored accordingly in the server.
- The *OPC UA server* provides basic services such as data exchange or navigation through the address space.
- The OPC UA configuration is used to define the variables or contents that an OPC UA server should provide.
- The OPC UA configuration is done via an external tool such as for CPUs from VIPA the *OPC UA Configurator* from VIPA.

OPC UA client

OPC UA clients are programs with the following functionality:

- Read or write access to information of the OPC UA server.
- Access is controlled by access rights.
- Execute methods on the OPC UA server.

Communication types

- Client/Server
  - An *OPC UA client* accesses information from the *OPC UA server* via methods provided by the *OPC UA server*. Here a fix defined connection is used.
  - Example: *OPC UA client* retrieves status of an input in the CPU.
- Publisher/Subscriber
  - A *Publisher* sends to unknown *Subscriber* (clients) without a fixed connection.
  - Example: Sensors send data to the cloud.

2.2.2 Information modeling

Information model

- *Information models* are used to describe devices and their data.
- The basis is the *Core specification*. The *Core specification* describes the structure of the address range and of the services, such as the entry points for the clients in the address space of an *OPC UA* server.
- In an *information model*, the content of the address space of the *OPC UA* server is described.
- The *Information models* are structured in layers. Each higher-order type is based on certain basic rules. Thus, clients who only know the basic rules can still edit complex information models, e.g. navigate through the *address space* and read or write data variables.
- In the *address space*, all information is represented by *Nodes*, which are interconnected via *references*.
- A node is always an instance of a *NodeClass*.
- *OPC UA* offers basic services such as data exchange or navigation through the address space. The services are grouped in *Service Sets*.

Node classes

The following *NodeClasses* are defined in the *OPC UA* specification:

- Variable - class of variables
- Method - class of functions
- Object - class of objects
- View - Class of view of a subset of nodes
- DataType - Class of the data types of the value of a variable
- VariableType - Class of the data types of a variable
- ObjectType - class of object types
- ReferenceType - class of reference types

Node attributes

Each *node* consists of attributes and references. Some attributes may also be optional. The following attributes of each *NodeClass* must be published:

- NodeID - Unique identifier of a *nodes* in the *address space*
- NodeClass - class of *node* instance
- BrowseName - name of the *node* in plain text
- DisplayName - display name of the *Node* for the user
- Description - Description of the *node* (optional)

***OPC UA* services**

- *OPC UA* services are abstract descriptions defined by request and response messages.
- The available services of an *OPC UA* server are defined in the server profile and grouped together in service sets.

Basic service sets

- Discovery Service Set
  - Services for discovering existing servers and endpoints.
- SecureChannel Service Set
  - Services for opening and closing secure communication channels.
- Session Service Set
  - Services for the client to create and manage a session.
- NodeManagement Service Set
  - Services for creating and deleting nodes and references.
- View Service Set
  - Services for the client to navigate in the address space or in the view.
- Query Service Set
  - Services for search queries in the address space.
- Attribute Service Set
  - Services for accessing attributes of nodes.
- Method Service Set
  - Service for calling a method of an object.
- MonitoredItem Service Set
  - Services for the client to create and manage monitored items.
  - Monitored items are used to log in for data and event notifications.
- Subscription Service Set
  - Services for the client to create and manage subscriptions.
  - Subscriptions control the way of the data and event notification.

Access

- To access an *OPC UA* server, the *endpoint* must be known.
- You can navigate via the *endpoint* using the navigation function through the address space of the *OPC UA* server. Here you receive information about the *OPC UA* server and the CPU and have access to the objects created in the *OPC UA* configuration, such as tags, data blocks, etc.
- Lower network load through *'subscriptions'*
  - If variables are to be transmitted only if their value has changed, you have to use *subscriptions*.
  - To activate a *subscription*, enter the transmission interval "Publishing Interval" in the *OPC UA* client.
  - When the *subscription* is created, tell the server which variables to monitor. Among other things, you can specify the amount by which a value must change in order a transfer takes place.
  - Since only a change in value of a transmission takes place, the use of *subscriptions* leads to a reduced network load.
- Fast access through *'registration'*
  - Normally the addressing takes place by means of identifier strings. By using a numeric identifier access can be accelerated. For this reason, you should use the *registration* for regular access to certain variables.
  - During *registration*, the *OPC UA* client assigns the variable to the *OPC UA* server. The *OPC UA* server then generates a numeric identifier and sends it back to the *OPC UA* client.
  - The numerical identifier is valid for the duration of the session.
  - In the properties of the CPU, you can set the maximum number of registered nodes. This must be taken into account by the *OPC UA* clients.
  - Since the registration takes time, you should put them in the start-up phase of the *OPC UA* server.

Setting the sampling intervals (sampling interval, publishing interval) too short may cause too much network load. Always choose intervals that are still sufficient for your application. Specifying -1 as the interval will use the default setting of the OPC UA server for the interval.

2.2.3 *OPC UA* data types and their conversion

Siemens S7 data types do not always match the *OPC UA* data types. The CPU provides variables to the *OPC UA* server as an *OPC UA* data type so that *OPC UA* clients can access these variables with *OPC UA* data types via the server interface. A client can read the "DataType" attribute from such a variable and reconstruct the original data type.

Data type mapping

Siemens S7 data type		SPEED7 PLC *OPC UA* data type		*OPC UA* data type
BOOL	→	BOOL	→	Boolean
BYTE		BYTE		Byte
WORD		WORD		UInt16
DWORD		DWORD		UInt32
INT		INT		Int16
DINT		DINT		Int32
REAL		REAL		Float
S5TIME		S5TIME		UInt16
TIME		TIME		Int32
DATE		DATE		UInt16
TIME_OF_DAY (TOD)		TIME_OF_DAY		UInt32
CHAR		CHAR		Byte
COUNTER		COUNTER		UInt16 (Only valid values)
TIMER		TIMER		UInt16 (Only valid values)
STRING		STRING		String
DT		DT		Byte[8]

Particularities

- String
  - The data type *STRING* in Siemens S7 is a byte array in which the maximum length and the current length are stored in the first 2 bytes. The other bytes store the string.
  - The *OPC UA* data type *String* should be defined in the same way.
- Array
  - A read or write job in *OPC UA* is always an *Array* access, i.e. always provided with index and length.
  - A single variable is a special case of an *Arrays* (index 0 and length 1). On the line, the data type is simply sent several times in succession. For the variable, the *DataType* attribute points to the base data type. The *ValueRank* and *ArrayDimensions* attributes determine if it is an array and how large the array is.
- Structure
  - A structure describes a complex data type.
  - You can describe your own structures as a subtype of the abstract data type *Structures*, which inherits from the data type *BaseDataType*.
  - Since a client may not know user-specific structures, the variables of the data type of this structure are uniformly published in an *ExtensionObject*. The structure *ExtensionObject* can be read by any client and also publishes the *DataTypeId* of the user-specific structure.
  - All structures that are not described by structures of the basic data types are published on the server in a *TypeDictionary*.
  - With the description of the structure by the *TypeDictionary* and the *DataTypeId*, which is published by the *ExtensionObject*, the structure of the *ExtensionObject* can be decoded by a client.
  - If a client knows in advance the description of a user-specific structure, it can be decoded without reading the *TypeDictionary*. In this approach, a client needs to read and decode the entire tree to access individual elements.

2.2.4 Integrated security concept

Generals to data security

The topic of data security and access protection have become increasingly important in the industrial environment. The increased networking of entire industrial systems to the network levels within the company together with the functions of remote maintenance have all served to increase vulnerability. Threats can arise from internal manipulation like technical errors, operator and program errors respectively from external manipulation like software viruses and worms, trojans and password phishing.

The most important precautions to prevent manipulation and loss of data security in the industrial environment are:

- Encrypting the data traffic by means of certificates.
- Filtering and inspection of the traffic by means of VPN - "Virtual Private Networks".
- Identification of the nodes by "Authentication" via save channels.
- Segmenting in protected automation cells, so that only devices in the same group can exchange data.

Guidelines for information security

With the "VDI/VDE 2182 sheet 1", Information Security in the Industrial Automation - General procedural model, VDI guidelines, the VDI/VDE society for measuring and automation engineering has published a guide for implementing a security architecture in the industrial environment. The guideline can be found at www.vdi.de PROFIBUS & PROFINET International (PI) can support you in setting up security standards by means of the "PROFINET Security Guideline". More concerning this can be found at the corresponding web site such as www.profibus.com

Security mechanisms in *OPC UA*

- Verifying the identity of *OPC UA* servers and clients.
- Checking the identity of the users.
- Signed and encrypted data exchange between *OPC UA* server and clients.
- In the connection settings in the *OPC UA Configurator*, you can specify how a user of an *OPC UA* client must legitimize access to the *OPC UA* server.

Safety rules:

- Only activate *'Anonymous-Login'* or *'Unsecured data traffic'* in exceptional cases.
- Only in exceptional cases use the "guest authentication" of the user.
- Only allow access to variables and data blocks via *OPC UA* if it is actually required.

Activate only security guidelines that are compatible with the protection concept for your machine or Application. Deactivate all other security guidelines.

X.509 certificates

OPC UA has integrated security mechanisms in multiple layers. An important component here are X.509 certificates, which are also used in the PC world. When using certificates, the *OPC UA* server delivers data to the client only if the security certificate has been accepted as valid on both sides. An X.509 certificate includes the following information:

- Version and serial number of the certificate.
- Name of the certification authority.
- Information about the algorithm used by the certification authority to sign the certificate.
- Start and end of the validity of the certificate.
- Name of the program, person, or organization for which the certificate was signed by the certification authority.
- The public key of the program, person or organization.

OPC UA uses three types of X.509 certificates when establishing a client-to-server connection:

- *OPC UA* application certificates
- *OPC UA* software certificates
- *OPC UA* user certificates

- Check when establishing a connection
  - When establishing a connection between client and server, the participants check all information from the certificate that is required to establish integrity.
  - Among other things, the period of validity which is stored in the certificate is checked. Please ensure that the date and time are set correctly for the participants, otherwise no communication can take place.
- Sign and encrypt
  - To avoid tampering, certificates are signed.
  - Within the *OPC UA Configurator*, you can use the *'Server settings'* to import certificates or create and sign them yourself.

- Self-signed certificate
  - Each participant generates his own certificate and signs it.
  - Self-signed certificates are to be transferred to the CPU.
  - From a self-signed certificate no new certificates can be derived.
  - Sample applications: Static configuration with limited number of communication participants.
- *CA certificate:*
  - All certificates are created and signed by a certification authority.
  - It is only necessary to transfer the certificate of the certification authority to the CPU.
  - The certification authority can generate new certificates. Adding partner devices is possible at any time.
  - Sample applications: Dynamically growing plants.

Digital signature

The signature can be used to prove the integrity and origin of a message.

1. The sender forms a hash value as a check value from the clear message.
2. The hash value and a private key result in the digital signature.
3. The clear message is sent to the recipient together with the digital signature.
4. The recipient decrypts the received signature with the public key and thus gets back the original hash value.
5. The receiver also forms a hash value from the clear message and checks it with the original hash value. The public key and hash method are included in the X.509 certificate.

   ⇨
   - If both hash values are identical, sender and clear message were not manipulated.
   - If both hash values are not identical, the clear message was manipulated or falsified during transmission.

Encrypting

- X.509 certificates are not encrypted; they are public and anyone can see them.
- Encrypting data prevents unauthorized users from knowing the content.
- When encrypting, the sender encrypts the clear message with the recipient's public key from the X.509 certificate.
- The recipient decrypts the message with his private key. Each owner of the private key can decrypt a received message.

Secure Channel

- *OPC UA* uses private and public keys to establish secure channels between client and server
- Once a secure connection is established, the client and server generate a shared private key for signing and encrypting messages.

Security policies

OPC UA uses the following security policies to protect messages:

- *No security*

  All messages are unsecured. To use these security policies, connect to a "None" endpoint of a server.
- *Sign*

  All messages are signed. This allows the integrity of the received messages to be checked. Manipulations are detected. To use these security policies, connect to a "Sign" endpoint of a server.
- *Sign & encrypt*

  All messages are signed and encrypted. This allows the integrity of the received messages to be checked. Manipulations are detected. Due to the encryption, no attacker can read the content of the message. To use these security policies, connect to a "Sign & Encrypt" endpoint of a server.

  The security guidelines are additionally named according to the algorithms used. Example: "Basic256Sha256 - Sign & Encrypt" means: Secure Endpoint, supports a set of algorithms for 256-bit hashing and 256-bit encryption.

3 Configure OPC UA

3.1 Overview

- With the OPC UA configuration you can set up and configure the integrated OPC UA server of a target station (CPU respectively CP).
- You can either use the OPC UA configuration within the *SPEED7 Studio* or start it from the Siemens SIMATIC Manager respectively Siemens TIA Portal. When called, the *SPEED7 Studio* opens as *OPC UA Configurator* with to OPC UA configuration limited functions.
- If you create or change the OPC UA configuration, you must compile this configuration and transfer it from the *OPC UA Configurator* into the target station.

3.2 Usage in Siemens SIMATIC Manager

Precondition

Siemens SIMATIC Manager from V5.5 and VIPA *SPEED7 Studio* from V1.8.6

- The *OPC UA* configuration happens by the external *OPC UA Configurator* from VIPA.
- The *OPC UA Configurator* is the *SPEED7 Studio* reduced to *OPC UA* functionality.
- The *OPC UA Configurator* can be registered in the Siemens SIMATIC Manager by means of the *SPEED7 Tools Integration*.
- The *OPC UA Configurator* is to be called from the Siemens SIMATIC Manager after project creation and online configuration.
- The *OPC UA Configurator* automatically imports the data for the *OPC UA* configuration from the project data of the Siemens SIMATIC Manager.
- The *OPC UA* configuration is transferred online from the *OPC UA Configurator*. The *OPC UA Configurator* automatically imports the data for the *OPC UA* configuration from the project data of the Siemens SIMATIC Manager.

Please note that only the objects of the LD, FBD and IL languages can be transferred to the OPC UA Configurator.

3.2.1 Installation *OPC UA Configurator*

Proceeding

Installation and activation of *SPEED7 Studio*

The *OPC UA Configurator* is part of the *SPEED7 Studio* with *OPC UA* functionality. With the *SPEED7 Tools Integration*, which is also installed when installing the *SPEED7 Studio* the *OPC UA Configurator* is to be registered in the Siemens SIMATIC Manager as external tool.

1. The latest version of the *SPEED7 Studio* can be found in the download area of www.vipa.com. Double-click on the installation program an follow the instructions on the monitor.

   The use of the SPEED7 Studio requires that you agree with the license agreement. During installation, you must confirm this.

   Further components are required in order to operate *SPEED7 Studio*. If the following programs are not already present on your PC, they are automatically installed:

   - Microsoft .NET Framework 4.52
   - Microsoft SQL Server© 2014 SP1
   - WinPcap

2. You can use a 30-day demo version or activate a license.

   In order to use *SPEED7 Studio* without restrictions, you require a licence, which you can obtain from your local VIPA customer service organisation.

   If the PC, on which you would like to use the *SPEED7 Studio*, is connected to the Internet, you can activate the licence online. If no license is activated, the dialog box for activating the license opens with each new start of *SPEED7 Studio*.

   Click on *‘Yes’*.

   ⇨ The *‘Product activation’* dialog window will open.

3. Enter the serial number that you received with your order of *SPEED7 Studio* in the *‘Licence key’* input field.
4. Enter your name in the *‘Your name’* input field.
5. If you enter your e-mail address in the *‘E-mail address’* input field, you receive an e-mail confirmation regarding the product activation.
6. Click at *‘Activate’*.

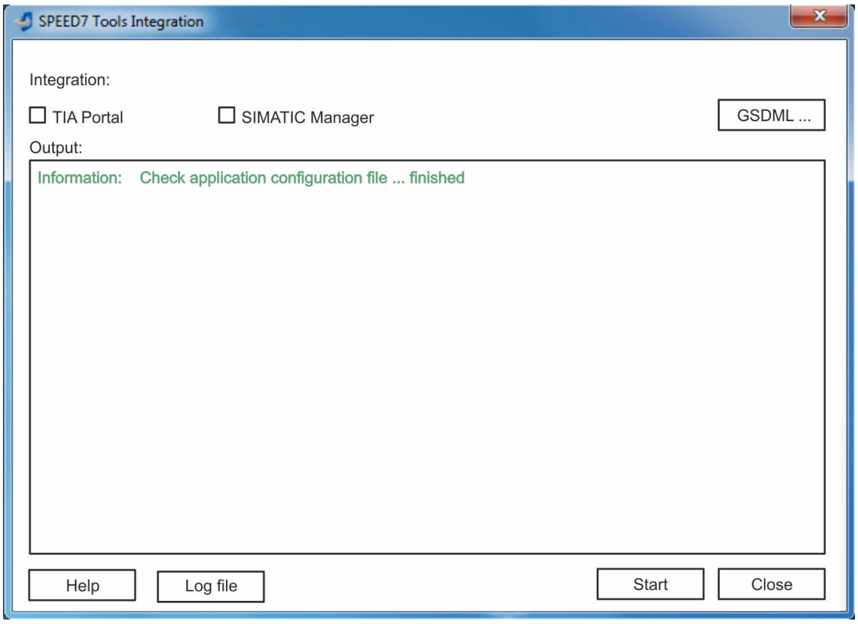
   ⇨ The licence is activated and the *SPEED7 Studio* is started.

Registration of SPEED7 Studio in the Siemens SIMATIC Manager as *OPC UA Configurator*.

SPEED7 Tools Integration is automatically listed in the Windows Start menu during the installation of the *SPEED7 Studio*.

1. To start the *SPEED7 Tools Integration*, click in the Windows Start menu on *'VIPA GmbH ➔ SPEED7 Tools Integration'*.

   ⇨ For *SPEED7 Tools Integration* can start, you have to acknowledge the security prompt to change the data on your computer with *'Yes'*. Afterwards *SPEED7 Tools Integration* will be started.



Information: Check application configuration file ... finished

2. Click at *'GSDML ...'*.
3. Navigate to your GSDML file of your VIPA-CPU, which you also use for your configuration in the Siemens *'SIMATIC Manager'*. Select these and click at *'Confirm'*. You can also select and use several GSDML files.

   ⇨ The identified GSDML files are listed and the selection for the configuration tools is enabled.

4. Select the Siemens *'SIMATIC Manager'*, in which the *SPEED7 Studio* is to be registered as *OPC UA Configurator*.
5. Click on *'Start'*.

   ⇨
   - *SPEED7 Studio* is registered in the Windows registry as *OPC UA Configurator*.
   - In the Siemens SIMATIC Manager the *OPC UA Configurator* is registered as externally callable program.
   - All changes are recorded in a log file, which you can output via *'Log file'*.

6. *'Close'* closes *SPEED7 Tools Integration*.

   ⇨ With the next start of the Siemens hardware configurator, the *SPEED7 Studio* can be called as *OPC UA Configurator* with to *OPC UA* configuration limited functions. More information about the usage can be found in the in the online help of the *OPC UA Configurator*.

3.2.2 Steps of the *OPC UA* configuration

Steps of configuration

When using the Siemens SIMATIC Manager, the *OPC UA* configuration happens by the following steps:

1. Create your project in the Siemens SIMATIC Manager with the corresponding hardware configuration.
2. Configure the corresponding Ethernet connection for PG/OP communication and establish an online connection.
3. Save translate and transfer your project.
4. Call the external *OPC UA Configurator* from the Siemens SIMATIC Manager. For this click in the hardware configurator on the CPU from VIPA and select *'Start Device Tool ➔ VIPA Framework ➔ OPC UA Configurator'*.
5. Confirm to start an external program with [YES].

> **NOTICE!**
>
> **Data exchange between platforms of different vendors**
>
> If you allow access, you permit the exchange of data between *OPC UA Configurator* and your project data of the Siemens SIMATIC Manager.
>
> – Ensure that the necessary security guidelines are complied with.

⇨ The *OPC UA Configurator* is started. For the *OPC UA* configuration, the data is taken from the Siemens SIMATIC Manager project and listed in the table for the *OPC UA* configuration.

> *Please note that only the objects of the LD, FBD and IL languages can be transferred to the OPC UA Configurator.*

6. Configure the *OPC UA* server and the data for the *OPC UA* communication.
7. In the *OPC UA Configurator* switch to the online dialog and transfer the *OPC UA* configuration. For communication the IP address data are taken from the Siemens SIMATIC Manager project.

⇨ The *OPC UA* configuration is now complete. For check you will find information about your *OPC UA* configuration on the device web page at *'OPC UA'*.

3.3 Usage in Siemens TIA Portal

Precondition

Siemens TIA Portal from version V15.0 and VIPA *SPEED7 Studio* from V1.8.6

- The *OPC UA* configuration happens by the external *OPC UA Configurator* from VIPA.
- The *OPC UA Configurator* is the *SPEED7 Studio* reduced to *OPC UA* functionality.
- The *OPC UA Configurator* can be registered in the Siemens TIA Portal by means of the *SPEED7 Tools Integration*.
- The *OPC UA Configurator* is to be called from the Siemens TIA Portal after project creation and online configuration.
- The *OPC UA Configurator* automatically imports the data for the *OPC UA* configuration from the project data of the Siemens TIA Portal.
- The *OPC UA* configuration is transferred online from the *OPC UA Configurator*. For the communication the *OPC UA Configurator* automatically uses the IP address data of the Siemens TIA Portal project.

Please note that only the objects of the LD, FBD and IL languages can be transferred to the OPC UA Configurator.

3.3.1 Installation *OPC UA Configurator*

Proceeding

Installation and activation of *SPEED7 Studio*

The *OPC UA Configurator* is part of the *SPEED7 Studio* with *OPC UA* functionality. With the *SPEED7 Tools Integration*, which is also installed when installing the *SPEED7 Studio* the *OPC UA Configurator* is to be registered in the Siemens TIA Portal as external tool.

1. The latest version of the *SPEED7 Studio* can be found in the download area of www.vipa.com. Double-click on the installation program an follow the instructions on the monitor.

   The use of the SPEED7 Studio requires that you agree with the license agreement. During installation, you must confirm this.

   Further components are required in order to operate *SPEED7 Studio*. If the following programs are not already present on your PC, they are automatically installed:

   - Microsoft .NET Framework 4.52
   - Microsoft SQL Server© 2014 SP1
   - WinPcap
2. You can use a 30-day demo version or activate a license.

   In order to use *SPEED7 Studio* without restrictions, you require a licence, which you can obtain from your local VIPA customer service organisation.

   If the PC, on which you would like to use the *SPEED7 Studio*, is connected to the Internet, you can activate the licence online. If no license is activated, the dialog box for activating the license opens with each new start of *SPEED7 Studio*.

   Click on *'Yes'*.

   ⇨ The *'Product activation'* dialog window will open.
3. Enter the serial number that you received with your order of *SPEED7 Studio* in the *'Licence key'* input field.
4. Enter your name in the *'Your name'* input field.
5. If you enter your e-mail address in the *'E-mail address'* input field, you receive an e-mail confirmation regarding the product activation.
6. Click at *'Activate'*.

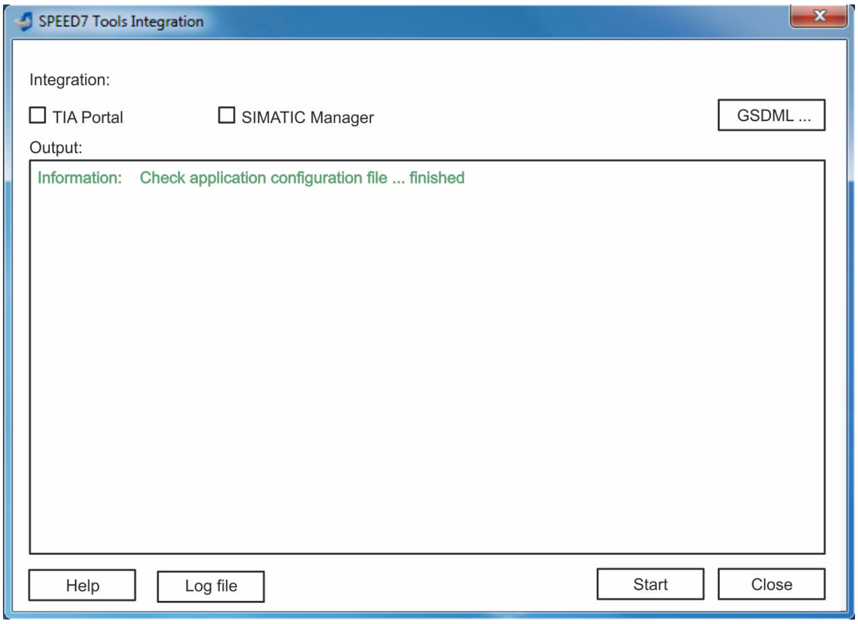
   ⇨ The licence is activated and the *SPEED7 Studio* is started.

Registration of SPEED7 Studio in the Siemens TIA Portal as *OPC UA Configurator*

SPEED7 Tools Integration is automatically listed in the Windows Start menu during the installation of the *SPEED7 Studio*.

1. To start the *SPEED7 Tools Integration*, click in the Windows Start menu on *'VIPA GmbH ➔ SPEED7 Tools Integration'*.

   ⇨ For *SPEED7 Tools Integration* can start, you have to acknowledge the security prompt to change the data on your computer with *'Yes'*. Afterwards *SPEED7 Tools Integration* will be started.



Information: Check application configuration file ... finished

2. Click at *'GSDML ...'*.
3. Navigate to your GSDML file of your VIPA-CPU, which you also use for your configuration in the Siemens *'TIA Portal'*. Select these and click at *'Confirm'*. You can also select and use several GSDML files.

   ⇨ The identified GSDML files are listed and the selection for the configuration tools is enabled.
4. Select *'TIA Portal'*, in which the *SPEED7 Studio* is to be registered as *OPC UA Configurator*.
5. Click on *'Start'*.

   ⇨
   - *SPEED7 Studio* is registered in the Windows registry as *OPC UA Configurator*.
   - In the Siemens TIA Portal the *OPC UA Configurator* is registered as externally callable program.
   - The current Windows user is registered in the user group *Siemens TIA Openness* of the Siemens TIA Portal.
   - All changes are recorded in a log file, which you can output via *'Log file'*.
6. *'Close'* closes *SPEED7 Tools Integration*.

   ⇨ With the next start of the Siemens TIA Portal, the *SPEED7 Studio* can be called as *OPC UA Configurator* with to *OPC UA* configuration limited functions. More information about the usage can be found in the in the online help of the *OPC UA Configurator*.

3.3.2 Steps of the *OPC UA* configuration

Steps of configuration

When using the Siemens TIA Portal, the *OPC UA* configuration happens by the following steps:

1. Create your project in the Siemens TIA Portal with the corresponding hardware configuration.
2. Configure the corresponding Ethernet connection for PG/OP communication and establish an online connection.
3. Save translate and transfer your project.
4. Call the external *OPC UA Configurator* from the Siemens TIA Portal. For this click at *'Devices & networks'* on the CPU of VIPA and select *'Start device tool'*.

   ⇨ A dialog window opens. Select *'OPC UA Configurator'* and click [Start].
5. Ignore the query *'Set interface'* with [OK]

   ⇨ The *OPC UA Configurator* is started.
6. If not yet confirmed, you will now receive an access request in the TIA Portal.

   > *Please note that due to the software the access request does not appear in the foreground. To show the access request, you must again bring the Siemens TIA Portal to the foreground. Once the access has been selected, you must again bring the 'OPC UA Configurator' to the foreground.*

   You have the following options for access:

   - *'No'*: Deny access - the *OPC UA Configurator* is not started.
   - *'Yes'*: Access is permitted once and the *OPC UA Configurator* is started.
   - *'Yes to all'*: Access is permitted and the *OPC UA Configurator* is started. At the next call, the access request is no longer shown.

   Allow access with *'Yes'* respectively *'Yes to all'*.

   ⇨

   > **NOTICE!**
   >
   > **Data exchange between platforms of different vendors**
   >
   > If you allow access, you permit the exchange of data between *OPC UA Configurator* and your project data of the Siemens TIA Portal.
   >
   > – Ensure that the necessary security guidelines are complied with.

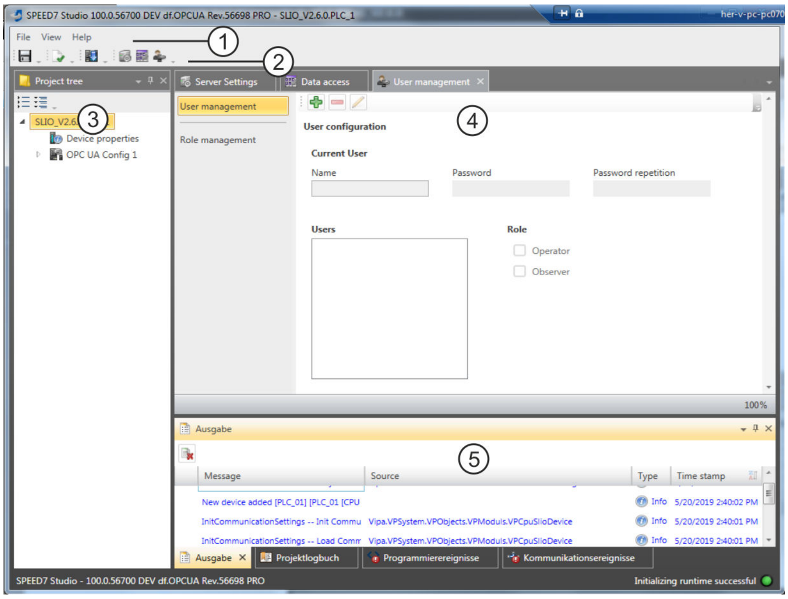
   For the *OPC UA* configuration, the data is taken from the Siemens TIA Portal project and listed in the table for the *OPC UA* configuration.

   > *Please note that only the objects of the LD, FBD and IL languages can be transferred to the OPC UA Configurator.*
7. Configure the *OPC UA* server and the data for the *OPC UA* communication.
8. In the *OPC UA Configurator* switch to the online dialog and transfer the *OPC UA* configuration. For communication the IP address data are taken from the TIA Portal project.

   ⇨ The *OPC UA* configuration is now complete. For check you will find information about your *OPC UA* configuration on the device web page at *'OPC UA'*.

3.4 *OPC UA Configurator*

The user interface of the *OPC UA Configurator* is divided into the following areas:



Message	Source	Type	Time stamp
New device added [PLC_01] [PLC_01 [CPU		Info	5/20/2019 2:40:02 PM
InitCommunicationSettings -- Init Commu	Vipa.VPSystem.VPObjects.VPModuls.VPCpuSlioDevice	Info	5/20/2019 2:40:01 PM
InitCommunicationSettings -- Load Comm	Vipa.VPSystem.VPObjects.VPModuls.VPCpuSlioDevice	Info	5/20/2019 2:40:01 PM

SPEED7 Studio - 100.0.56700 DEV df.OPCUA Rev.56698 PRO

Initializing runtime successful

1 Menu bar
2 Toolbar
3 Project tree
4 Workspace
5 Output area

Menu bar

In the menu bar you will find a few general commands on the *OPC UA Configurator*. Further commands can be called up via context menus with the right mouse button, e.g. functions for an object in the project tree.

Toolbar

Store *OPC UA* configuration

Compile *OPC UA* configuration

Transfer *OPC UA* configuration into the control

Project tree

The *Project tree* gives you access to the *'Device properties'* and to the following areas of the *'OPC UA configuration'*:

- Server settings
- Data access
- User management

Workspace

In the *Work space*, you can edit the settings in the following areas of the *OPC UA* configuration:

- Device properties - General
  - Information about the CPU such as device name, name and firmware version.
- Device properties - Communication
  - Configuration of the interface for data exchange.
  - The IP address data are automatically imported from the project when the *OPC UA Configurator* is called and can be viewed here.

- Device properties - Server configuration
  - Administration and interface assignment of the *OPC UA* server in the *Project tree*
- Server settings - Connection
  - Legitimation of the user for access to the *OPC UA* server.
  - Port for communication.
  - Security policy for encryption and corresponding exceptions.
- Server settings - Certificate
  - Create, view, import or export X.509 ITU-T standard certificate.
  - Re-creating or importing replaces an existing certificate.
- Data access
  - Selection of the variables that can be accessed via *OPC UA*.
  - Filter option to limit the selection.
- User management
  - Creation of a user list with password and role assignment.

Output area

The output area shows information about activities performed and background operations.

3.5 Project tree

You can edit the *OPC UA* configuration via the project tree. The project tree contains the *OPC UA* configurations, which you have created. You can create a maximum of two *OPC UA* configurations: One configuration for the CPU and one configuration for the CP (if exists).

Show project tree

If the project tree is not shown select *'View ➔ Project tree'* or press *[Strg]*+*[Shift]*+*[P]*.

Show/hide objects

The objects in the project tree are arranged in a tree structure. You can show or hide objects:

Hide all objects (*'Project ➔ Collapse project tree'*)

Show all objects (*'Project ➔ Expand project tree'*)

▸ Hide slave objects / close folder

▾ Show slave objects / open folder

Edit configurations and *OPC UA* configuration

Device properties	
Device properties	- Edit device name and comment 🔗 *Chap. 3.6.2 'General device properties' page 23* - Perform communication settings 🔗 *Chap. 3.6.3 'Communication settings' page 24* - Create *OPC UA* configuration 🔗 *Chap. 3.6.4 'Server configuration' page 25*
OPC UA	
Server settings	🔗 *Chap. 3.7 'Server settings - Connection ' page 25* 🔗 *Chap. 3.8 'Server settings - Certificate ' page 26*
Data access	🔗 *Chap. 3.9 'Data access ' page 28*
User management	🔗 *Chap. 3.10 'User management ' page 29* 🔗 *Chap. 3.11 'Role management ' page 29*

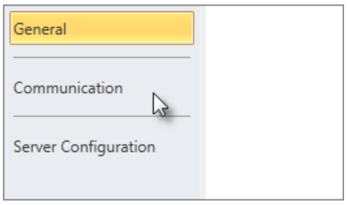
3.6 Device properties

3.6.1 Overview

Here you can edit the device name and the comment, perform the communication settings as well as create the *OPC UA* configuration.

→ Click in the project tree at *'Device properties'*.

⇨ The *'Device properties'* editor opens.



The *'Device properties'* editor is divided into several sections:

- 🔗 *Chap. 3.6.2 'General device properties' page 23*
- 🔗 *Chap. 3.6.3 'Communication settings' page 24*
- 🔗 *Chap. 3.6.4 'Server configuration' page 25*

3.6.2 General device properties

To show or change the device properties, proceed as follows:

1. Click in the project tree at *'Device properties'*.

   ⇨ The editor of the *'Device properties'* opens.

2. Select the area *'General'*.

'Device type'	-	Name of the CPU
'Firmware'	-	Firmware version of the CPU
'Name'	-	Device name: This name is shown in the project tree.
'Author'	-	Name of the responsible person who created the device
'Comment'	-	Any comment, e.g. an annotation or explanation

→ Click on the input field and enter any comment, e.g. an annotation or explanation. With the *[Enter]* key, you can add a new line to the input field.

3.6.3 Communication settings

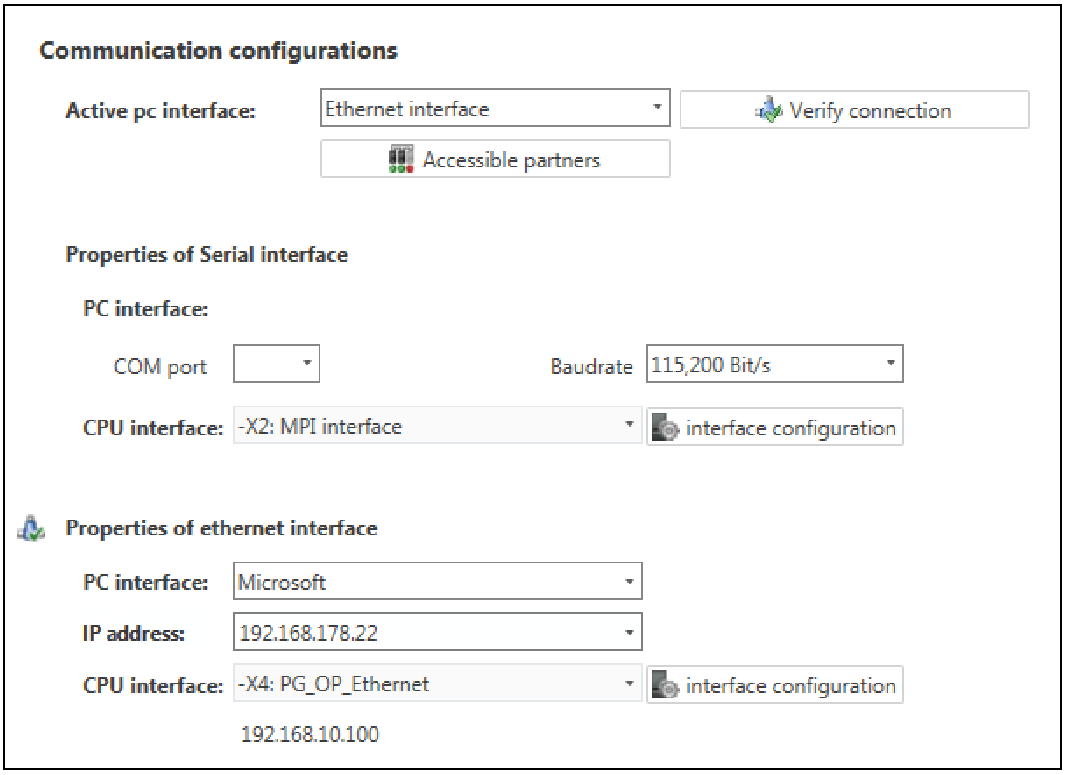
The communication settings are used to configure the interface for the data exchange between programming device and destination station. Since the IP address parameters for the *OPC UA* configuration are imported from the project, you simply have to set the interface via which you are connected to the destination station.

Device properties

1. Click in the project tree at *'Device properties'*.

   ⇨ The editor of the *'Device properties'* opens.

2. Select the area *'Communication settings'*.



Communication configurations

Active pc interface: Ethernet interface

Properties of Serial interface

PC interface:

COM port

Baudrate 115,200 Bit/s

CPU interface: -X2: MPI interface

Properties of ethernet interface

PC interface: Microsoft

IP address: 192.168.178.22

CPU interface: -X4: PG_OP_Ethernet

192.168.10.100

Setting the Ethernet interface

1. *'Active PC interface'*: Select *'Ethernet interface'*.
2. *'PC interface'*: Select the network adapter for the communication connection from the list.

   ⇨ If an IP address is already configured in the network adapter, it is shown under the input field *'IP address'*. If necessary, select a different IP address.

3. *'CPU interface'*: Select the interface of the control from the list.

   ⇨ Since the IP address is imported from the project, it is shown below the input field.

4. To configure further settings of the interface, click on *'Interface configuration'*.

   ⇨ The *'Interface properties'* dialog window will open.

5. In order to check whether a connection between the programming device and the control can be established with the selected communication settings, click on *'Verify connection'*.

   ⇨ You can see in the status line, whether the connection could be established successfully.

6. In order to check whether your programming device is connected with the correct control, you can retrieve information from the connected control. For this click on *‘Accessible partners’*.

   ⇨ The *‘Search for accessible partners’* dialog window will open.

3.6.4 Server configuration

Here you can create the *OPC UA* configurations.

1. Click in the project tree at *‘Device properties’*.

   ⇨ The editor of the *‘Device properties’* opens.

2. Select the area *‘Server configuration’*.

You can create a maximum of two *OPC UA* configurations: One configuration for the CPU and one configuration for the CP (if exists).

Create configuration

1. Click at on *‘Add Server’*.

   ⇨ A new *OPC UA* configuration is created and listed in the project tree.

2. Click in the selection field *‘Active server CP’* or *‘Active server CP’* and choose which configuration is to be assigned. With the selection *‘None’* the configuration remains saved in the project. However, it is not transferred to the device.

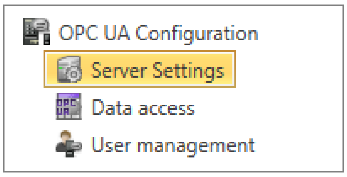
To swap the two configurations for CP and CPU, click on the button .

You can create a maximum of two *OPC UA* configurations.

Remove server

Right-click on the *OPC UA* configuration (PLC) in the project tree and select *‘Remove OPC UA server’*.

3.7 Server settings - Connection



Here you can perform the connection settings of the *OPC UA* server.

1. Under *Project tree* at *‘OPC UA configuration’* click on *‘Server settings’*.

   ⇨ The *‘Server settings editor’* editor opens.

2. Select the area *‘Connection’*.

General

You can set for the *OPC UA* server how a user of an *OPC UA* client must prove their identity for access to the server. Select at least one of the following login methods. You can also combine the two login methods with each other.

- *‘Activate anonymous login’*
  – The *OPC UA* server does not check the authorisation of the *OPC UA* client.
- *‘Activate user/password login’*
  – The *OPC UA* server checks using the user name and password whether the access of the *OPC UA* client is authorised. To do this, the server evaluates the role assigned to the user. *Chap. 3.11 ‘Role management’ page 29*

- *'Allow obsolete security guideline'*
  - Allows the selection of the two obsolete security guidelines *'Basic128Rsa15'* and *'Basic256'* (not recommended)
- *'Application name'*
  - Clear identification of the application in the OPC name space.

Network

- *'End point port'*
  - TCP port for binary data exchange (standard: 4840).

Security

> *Activate only security guidelines that are compatible with the protection concept for your machine or system. Deactivate all other security guidelines.*

- *'None'*
  - Insecure data traffic between server and client.
- *'Basic128Rsa15'*
  - Secured data traffic, hash algorithm RSA-15, 128-bit encoding (allow option with *'Allow obsolete security guideline'* see above).
- *'Basic256'*
  - Secured data traffic, 256-bit encoding (allow option with *'Allow obsolete security guideline'* see above).
- *'Basic256Sha256'*
  - Secured data traffic, hash algorithm SHA-256, 256-bit encoding (recommended).

Encoding:

- *'Sign'*
  - Endpoint secures the integrity of the data through signing.
- *'SignAndEncrypt'*
  - Endpoint secures the integrity and confidentiality of the data through signing and encoding.
- *'Both'*
  - The *OPC UA* server offers both encryption methods *'Sign'* and *'SignAndEncrypt'*. The *OPC UA* client can use one of the two encoding methods.

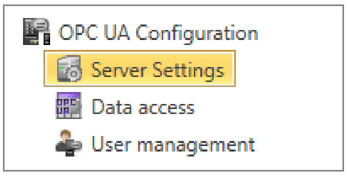
Security Check Overrides

Here you can allow various exceptions in the security check, in order to increase the error tolerance.

3.8 Server settings - Certificate

A secure connection between the *OPC UA* client and the server can only be established if the server classifies and accepts the client's digital certificate as trusted. Currently, the server accepts every valid client certificate. The server accepts self-signed certificates. In addition, the client also checks the server's certificate.

Here you can create, show, import or export an ITU-T standardized X.509 certificate for the *OPC UA* server. The certificate shown here is transferred into the *OPC UA* server.



OPC UA Configuration
Server Settings
Data access
User management

1. Under *Project tree* at *‘OPC UA configuration’* click on *‘Server settings’*.
   ⇨ The *‘Server settings’* editor opens.
2. Select the area *‘Certificate’*.

The current X.509 certificate is shown in the work space. If you create or import a new certificate, the previously shown certificate is replaced.

Toolbar

Create new certificate: Opens the dialog window *‘Create new certificate’*

Display certificate: Shows information on the current certificate

Export certificate: Opens the dialog window *‘Save certificate’*

Import certificate: Opens the dialog window *‘Open certificate’*

Create new certificate

1. Click on to create a new certificate.
   ⇨ The dialog window *‘Create new certificate’* opens.
2. Enter the data for the certificate and click on *‘OK’*.
   ⇨ The previously shown certificate is replaced by the new certificate.

Display certificate

- Click on to show information about the current certificate.
  ⇨ The dialog window *‘Certificate’* opens.

Export certificate

You can export the current certificate e.g. to use it on different computers.

1. Click on .
   ⇨ The dialog window *‘Save certificate’* opens.
2. Select a directory and enter a file name.
3. Click on *‘Save’*.
   ⇨ The current certificate is saved in the export file (pfx file format).

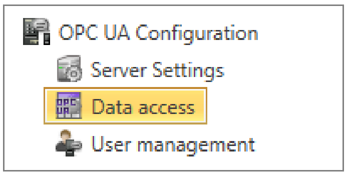
Import certificate

You can import a certificate, e.g. to use it for the current *OPC UA* configuration.

1. Click on .
   ⇨ The dialog window *‘Open certificate’* opens.
2. Select the desired certificate (pfx file format).
3. Click on *‘Open’*.
   ⇨ The previously shown certificate is replaced by the imported certificate.

3.9 Data access

Here you can select the variables belonging to the CPU or CP (if exists) that can be accessed via *OPC UA*.



OPC UA Configuration
Server Settings
Data access
User management

- Under *Project tree* at *'OPC UA configuration'* click on *'Data access'*.
  ⇨ The editor for *'Data access'* opens.

Toolbar

🔄 **Refresh variables:** Apply changed filter settings to the result table.

Filter settings

Here you can select the operands and address ranges that will be shown in the results table.

1. Activate ☑ *'All operand areas'* or individual operand ranges to be shown in the result table.
2. To limit the addresses of an operand range, enter the start and end byte addresses in the two adjacent fields, e.g. `0` to `1000`.
3. Click on 🔄 or activate ☑ *'Apply filter changes immediately'*.
   ⇨ The result table is updated with the filter settings.

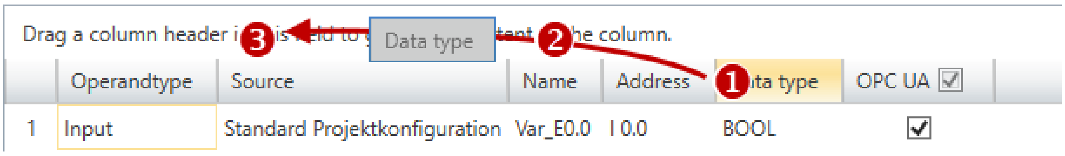
Result

In the results table, select the variables that are to be used in the *OPC UA* configuration. *OPC UA* clients may access these variables.

- Activate ☑ *'OPC UA'* of the desired variables.

Group operands

For a better overview, you can sort the table entries by groups.



	Operandtype	Source	Name	Address	ta type	OPC UA ☑
1	Input	Standard Projektkonfiguration	Var_E0.0	I 0.0	BOOL	☑

(1) Select column (hold left mouse button down)
(2) Drag the column
(3) Drop column in the field (release mouse button)

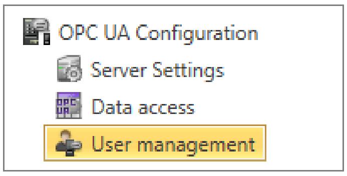
1. Drag the desired column header into the field above the table.
   ⇨ The contents of the column will be grouped. The number of lines is shown for each group.
2. Click on ▸ to open the group. Click on ▾ to close the group.

You can repeat steps 1 to 2 in order to structure the group into further sub-groups.

In order to cancel a grouping, click on the close icon next to the group name.

3.10 User management

The user management allows you to create a user list. For each user, you can define a password and a role.



OPC UA Configuration
- Server Settings
- Data access
- User management

1. Under *Project tree* at *‘OPC UA configuration’* click on *‘User management’*.

   ⇨ The editor for *‘User management’* opens.

2. Select the area *‘User management’*.

Toolbar

✚ **Add new user:** Input mode for new user

▬ **Remove user:** Deletes the selected user

✎ **Edit current user:** Input mode for selected user

💾 **Save input:** Save input Save user settings

✖ **Cancel input:** Cancel user settings without saving

Adding a user

1. Click on ✚.
2. Enter the desired user name in the input field *‘Name’*.
3. Enter the password in the input field *‘Password’* and repeat the input under *‘Re-enter password’*.
4. Select a role for the user. With this role, the access rights to the *OPC UA* server are established.
5. Click on 💾.

   ⇨ The user will be entered in the user list.

Edit user

1. In the user list, select the user whose data you want to change.
2. Click on ✎.
3. Enter the desired changes and click on 💾.

Removing a user

1. In the user list, select the user you want to delete.
2. Click on ▬.

   ⇨ A dialog box opens where you can choose whether the user should be deleted or not.

3.11 Role management

Here you establish the roles and access rights that you can assign to the users. When you activate the authentication via User/password login ↳ *Chap. 3.7 ‘Server settings - Connection* ⚙*’ page 25*, the access rights to the *OPC UA* server are issued using the logged-in user and the assigned role.

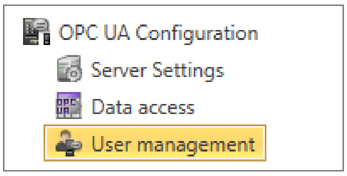
Example:

Role: Operator

Username: "I myself"

Server settings: User/password login activated

The user "Me Self" receives write permission and reading rights to the *OPC UA* server when he has successfully logged in with the password.



1. Under Project tree at *'OPC UA configuration'* click on *'User management'*.
   ⇨ The editor for *'User management'* opens.
2. Select the area *'Role management'*.

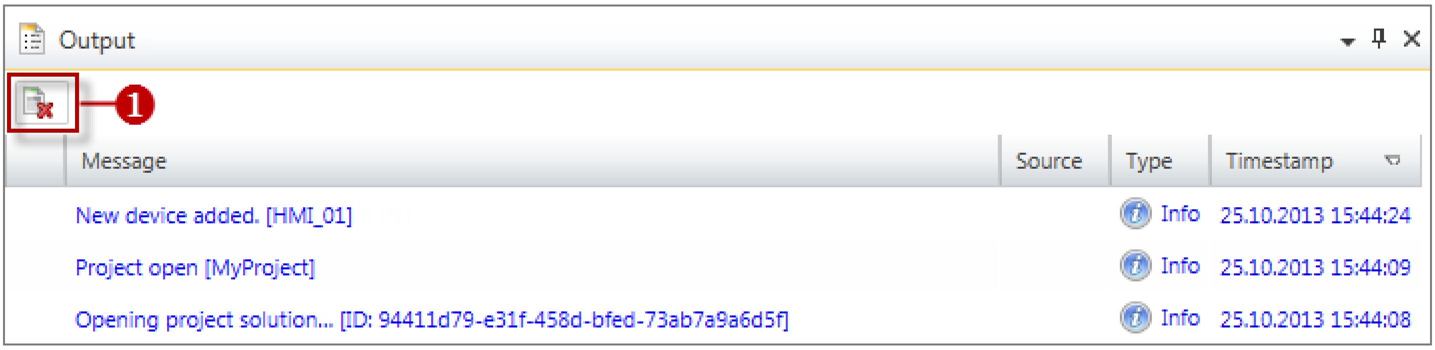
Configure roles

The following two roles are currently available for selection; further roles can not be added at the moment.

- Operator: Write permission and reading rights
- Observer: Reading rights only

3.12 Output

Information on executed activities and background operations are displayed in the "Output" window.

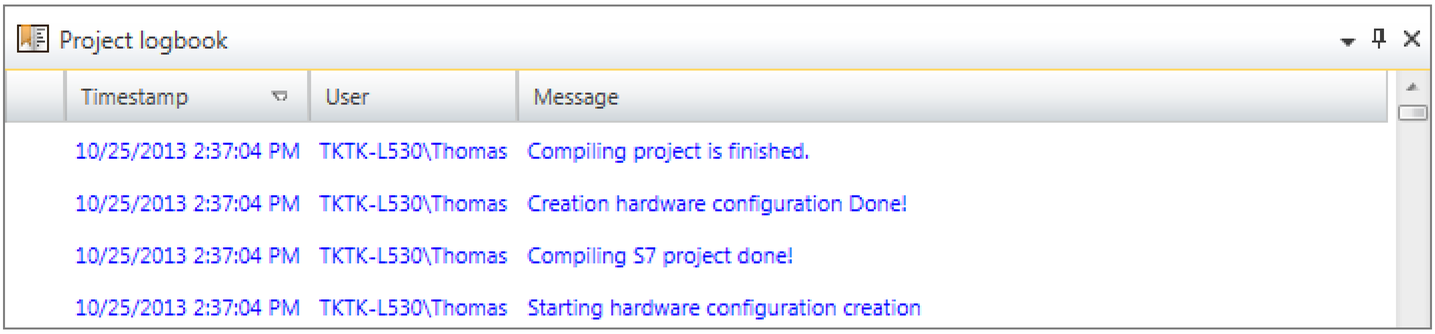


Message	Source	Type	Timestamp
New device added. [HMI_01]		Info	25.10.2013 15:44:24
Project open [MyProject]		Info	25.10.2013 15:44:09
Opening project solution... [ID: 94411d79-e31f-458d-bfed-73ab7a9a6d5f]		Info	25.10.2013 15:44:08

(1) Delete all messages in the output window

3.13 Project logbook

All activities are chronologically listed in the "Project logbook" window.

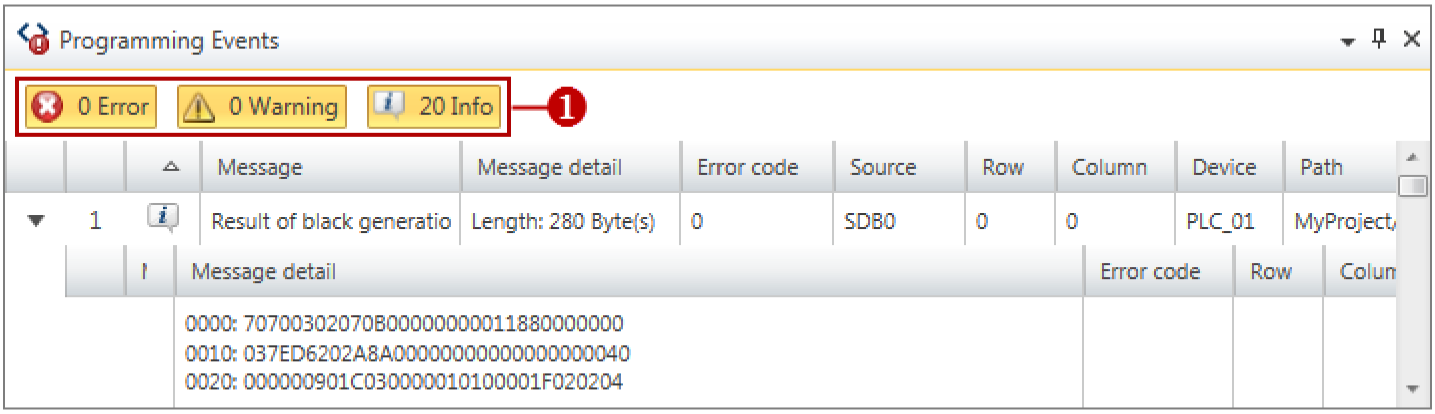


Timestamp	User	Message
10/25/2013 2:37:04 PM	TKTK-L530\Thomas	Compiling project is finished.
10/25/2013 2:37:04 PM	TKTK-L530\Thomas	Creation hardware configuration Done!
10/25/2013 2:37:04 PM	TKTK-L530\Thomas	Compiling S7 project done!
10/25/2013 2:37:04 PM	TKTK-L530\Thomas	Starting hardware configuration creation

Fig. 1: Project logbook

3.14 Programming events

Information on events in the PLC program are provided in the "Programming events" window.



			Message	Message detail	Error code	Source	Row	Column	Device	Path
▼	1	ℹ	Result of black generatio	Length: 280 Byte(s)	0	SDB0	0	0	PLC_01	MyProject/

	Message detail	Error code	Row	Colum
	0000: 70700302070B00000000011880000000 0010: 037ED6202A8A00000000000000000040 0020: 000000901C030000010100001F020204			

Fig. 2: Programming events

(1) Show/hide messages

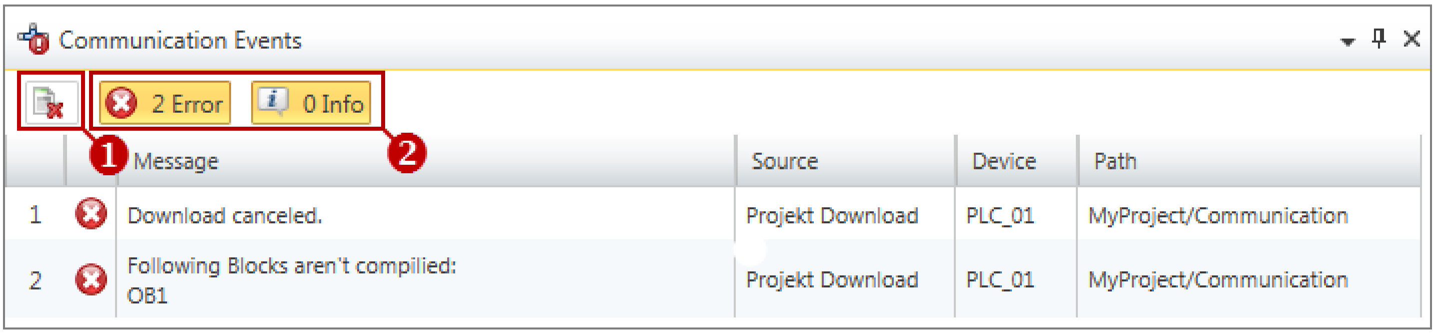
Show/hide details

You can show or hide further details on a message:

- ▸ Hide message details
- ▾ Show message details

3.15 Communication events

Information on communication events between the programming device and the connected devices are provided in the "Communication events" window.



		Message	Source	Device	Path
1	⊗	Download canceled.	Projekt Download	PLC_01	MyProject/Communication
2	⊗	Following Blocks aren't compilied: OB1	Projekt Download	PLC_01	MyProject/Communication

Fig. 3: Communication events

(1) Delete all messages in the output window
(2) Show/hide messages