

SIMATIC S7 Connector Configurator V1.2


Operating Manual


General Data Protection Regulation (GDPR)	1
Security information	2
Security Information for Industrial Edge App	3
Introduction to SIMATIC S7 Connector	4
User Interface for SIMATIC S7 Connector Configurator Home Page	5
Working with SIMATIC S7 Connector Configurator	6
Additional Information	7


Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

 DANGER
indicates that death or severe personal injury will result if proper precautions are not taken.

 WARNING
indicates that death or severe personal injury may result if proper precautions are not taken.

 CAUTION
indicates that minor personal injury can result if proper precautions are not taken.

NOTICE
indicates that property damage can result if proper precautions are not taken.


If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

 WARNING
Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Table of contents

1	General Data Protection Regulation (GDPR)	5
2	Security information	7
3	Security Information for Industrial Edge App	9
4	Introduction to SIMATIC S7 Connector	11
5	User Interface for SIMATIC S7 Connector Configurator Home Page	13
6	Working with SIMATIC S7 Connector Configurator	15
6.1	Managing Data Sources	15
6.1.1	Configure S7-Protocol (S7-300/400/1200/1500) Data Source	15
6.1.2	Configure OPC-UA Data Source	18
6.1.3	Configure Optimized S7-Protocol (S7-1200/1500) Data Source.....	21
6.1.4	Edit Data Source	24
6.1.5	Delete Data Source	24
6.2	Managing Tags/Data Points	25
6.2.1	Add Tags.....	25
6.2.2	Edit Tags.....	31
6.2.3	Import Tags	32
6.2.4	Browse Tags.....	35
6.2.5	Bulk Publish Tags	37
6.2.6	Delete Tags	39
6.3	Managing Project.....	40
6.3.1	Configure Settings.....	40
6.3.2	Deploy Project.....	41
6.3.3	Start Project.....	43
6.3.4	Stop Project	44
6.3.5	Verify Configuration	44
6.4	Import/Export Configuration.....	45
7	Additional Information	47
7.1	How to Write Tags?.....	47
7.2	Topic Structure for Data and Metadata.....	48
7.2.1	JSON structure for Alarms and Tags Data	48
7.2.2	JSON structure for Alarms and Tags Metadata	54
7.2.3	Connection and Disconnection Status	56
7.3	How to Export Tags from TIA Portal?.....	58
7.3.1	Export Tags for S7-Protocol (S7-300/400/1200/1500) and Optimized S7-Protocol (S7-1200/1500) Connection.....	58
7.3.2	Export Tags for OPC-UA Connection.....	63
7.4	How to Configure String Datatype in S7-Protocol (S7-300/400/1200/1500)?	64
7.5	How to Generate OPC-UA Client and Server Certificates?	65

7.6	How to Configure Access Password?	73
7.7	How to Configure S7-1200 PLC FW version 4.5?	76
7.8	Configuration Version 1.1 vs 1.2.....	77
7.9	Data Types	77
7.10	Notes on use.....	95
7.11	Known issues.....	98

General Data Protection Regulation (GDPR)

Siemens adheres to the principles of data protection, in particular the principles of data minimization (Privacy by Design).

For this product, SIMATIC S7 Connector Configurator, this means:

Personal data

There is no personal data* collected but following data is stored to allow machine to machine communication:

- Industrial Edge Databus credentials
- S7+ legitimization credentials
- OPC-UA server login credentials
- OPC-UA server certificates for encryption and signing
- Tags data and metadata from field devices
- Timestamp
- Smart device information and app usage data

If the customer links the data mentioned above to other data (e.g., shift plans) or if the customer saves personal information on the same medium (e.g. hard disk) and thus creates a personal reference, the customer has to ensure that the guidelines regarding data protection are observed.

Note

* This section refers to any personal data processed by the Application other than the personal data contained in log-files / tracking data (if any). "Personal data" are any information relating to an identified or identifiable natural person. Please note that IP-addresses, device identifiers such as IMEI, UDID, IMSI, MAC-address, MSISDN, location data or machine data (if machine data tracks events triggered by user interaction with the machine) usually qualify as personal data.

Purposes

The data mentioned above is required for the following purposes:

- Access protection and security measures
- Message system for traceability and availability
- For app diagnosis

Storage of the data is affected for a suitable purpose and is limited to what is strictly necessary, as the information is indispensable in order to identify the authorized operators.

Securing of data

The above data will not be stored anonymously or pseudonymized, as the purpose (identification of the operating personnel) cannot be achieved otherwise.

The following data will be used only within the product and within the Edge eco-system and will not be automatically passed on to third parties or unauthorized persons:

- Industrial Edge Databus credentials
- S7+ legitimization credentials
- OPC-UA server login credentials
- OPC-UA server certificates

The above data is secured by adequate technical measures, such as storing and encryption of process data in databases.

The tags data and metadata from field devices data will be used only within the product and will not be automatically passed on to third parties or unauthorized persons.

The customer must ensure the access protection as part of his process configuration.

Deletion policy

This product does not provide an automatic deletion for the databus or PLC credentials already provided by the user. In case the user provides a different databus or PLC credentials, the previous credentials will be overwritten. Since there is no explicit delete option, the user could provide junk databus or PLC credentials to delete the existing valid credentials.

If the user deletes a connection or tag, then the credentials information and other details would also be deleted.

In short, the collected log data will be automatically deleted once the limit is reached (oldest entries first).

Data configuration

The customer can configure the data collected via the product as follows:

- Using the App Configurator

Security information

Siemens provides products and solutions with industrial security functions that support the secure operation of plants, systems, machines, and networks.

In order to protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial security concept. Siemens' products and solutions form one element of such a concept.

Customers are responsible for preventing unauthorized access to their plants, systems, machines and networks. These systems, machines and components should only be connected to the enterprise network or the Internet if and only to the extent necessary and with appropriate security measures (firewalls and/or network segmentation) in place.

You can find more information on protective measures in the area of industrial security by visiting:

<https://www.siemens.com/industrialsecurity>.

Siemens' products and solutions undergo continuous development to make them more secure. Siemens strongly recommends performing product updates as soon as they are available and using only the latest product versions. Use of product versions that are no longer supported, and failure to apply latest updates may increase customer's exposure to cyber threats.

To stay informed about product updates, subscribe to the Siemens Industrial Security RSS Feed under

<https://www.siemens.com/industrialsecurity>.

Security Information for Industrial Edge App

Security information (assumption/constraint) for Industrial Edge Apps are as follows:

- Only authorized internal operators will have access to Industrial Edge Device with-in secure network using VPN connection.
- Perimeter firewall configuration responsibility lies with end customer.
- Security guidelines for usage of USB sticks within shop floor are applied.
- Creating users with appropriate access rights needs to be done during commissioning and it is the responsibility of the operator.
- Customer is responsible for configuring the application as per the installation/user manual, based on system requirements and technical capabilities of app documented so that the Automation System performance is not impacted.
- The system is installed in an environment that ensures physical access is limited to authorized maintenance personnel only. Managing unauthorized attachment of removable devices is the responsibility of the operator.
- The platform including hardware, firmware and operating system is securely configured and maintained by the operator.
- The operator is capable of protecting the environment from malware infection.
- Centralized IT security components (Active Directory, Centralized IT Logging Server) are provided and well secured by the operator and can be trusted.
- The operator personnel accessing the system is well trained in the usage of the system and general information security aspects like password handling, removable media, etc. are in place.
- Operator is responsible for the CIA of data stored outside the Industrial Edge Device.
- Operator is responsible for configuring the PLC's with appropriate read/write access levels (Legitimization) and configure Industrial Edge Apps with appropriate passwords for data collection from PLC's.
- Customer takes care about time sync of Industrial Edge Management and Industrial Edge Device.

Introduction to SIMATIC S7 Connector

The SIMATIC S7 Connector connects your Industrial Edge Device to a S7-Protocol (S7-300/400/1200/1500) controller, Optimized S7-Protocol (S7-1200/1500) controller, or an OPC UA (OPC Server). You can configure these connections using the SIMATIC S7 Connector Configurator. You can use the configured connections to transfer the measured value series of selected data points to the Industrial Edge Runtime of the respective Industrial Edge Device. The Industrial Edge Runtime sends this data to the Industrial Edge Databus. You can then use the data collected via the Industrial Edge Databus for your Industrial Edge App or other applications.

The SIMATIC S7 Connector provides easy connectivity with SIMATIC PLCs (S7-300, S7-400, S7-1200, S7-1500) via communication channel for tags and alarms data acquisition.

The SIMATIC S7 Connector consists of the following two components:

1. SIMATIC S7 Connector Configurator in Industrial Edge Management
The SIMATIC S7 Connector Configurator in the Industrial Edge Management provides its own user interface that you can use to manage the SIMATIC S7 Connector on the respective Industrial Edge Device.
2. SIMATIC S7 Connector
The SIMATIC S7 Connector is an application that runs on the individual Industrial Edge Device. You can install the SIMATIC S7 Connector on all Industrial Edge Devices that you want to connect to your controllers. The SIMATIC S7 Connector sends the data point values imported from the controllers to the Industrial Edge Databus through the Industrial Edge Runtime.

User Interface for SIMATIC S7 Connector Configurator Home Page

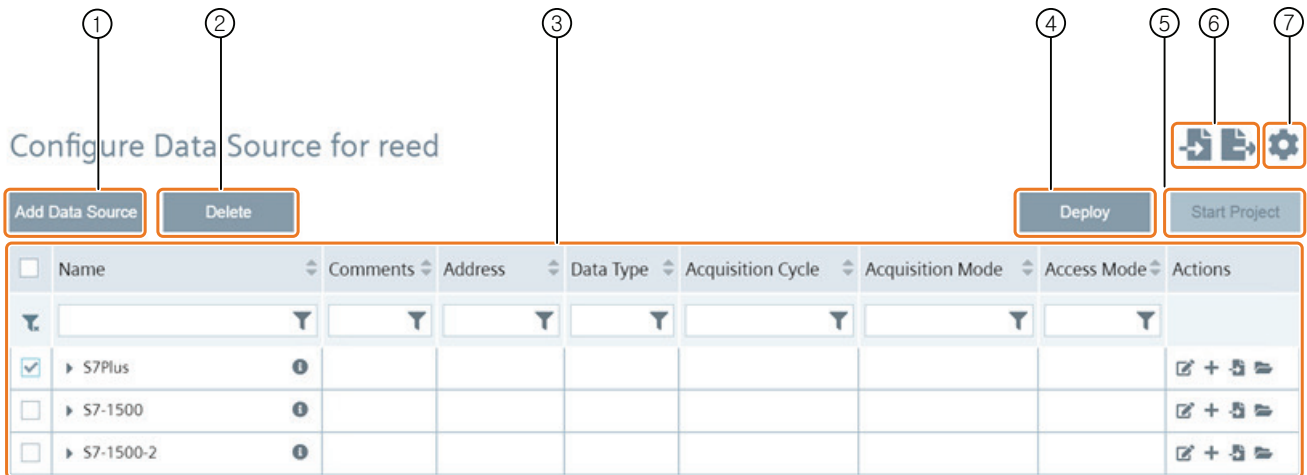
5

Prerequisite

- SIMATIC S7 Connector app must be installed and running on the Industrial Edge Device.
- SIMATIC S7 Connector Configurator must be installed and running on the Industrial Edge Management.

Home page

When you launch SIMATIC S7 Connector Configurator, the home page is displayed as follows:



- ① Add Data Source Button
- ② Delete Button
- ③ Data Source Table
- ④ Deploy Button
- ⑤ Start/Stop Project Button
- ⑥ Import/Export Icons
- ⑦ Settings Icon

UI Elements

The following table lists the different UI elements in the "Configure" section page:

Symbol	Description
Add Data Source Button	Enables to add a new data source.
Delete Button	Enables to delete the data source and data points.

Symbol	Description
Data Source Table	<p>Displays the data source and data points and their information as follows:</p> <ul style="list-style-type: none"> • "Name": Displays the name of the data point. • "Comments": Displays the given comments. • "Address": Displays the address of the data point in the controller or on the server. • "Data Type": Displays the data point type. • "Acquisition Cycle": Displays the acquisition cycle with which the data is sent to the Databus. • "Acquisition Mode": Displays the acquisition mode with which the data is sent to the Databus. • "Access Mode": Displays the access permission. • "Actions": Displays the "Edit Data Source", "Add Tags", and "Import Tags" options.
Deploy Button	<p>Enables to deploy the project. If you close the SIMATIC S7 Connector Configurator, then all the unsaved configurations are lost. The "Deploy" button saves the configuration on the Industrial Edge Runtime of the SIMATIC S7 Connector.</p>
Start/Stop Project Button	<p>Enables to start and stop the project.</p>
Import/Export Icons	<p>Enable to import and export the configuration. For more information, refer Import/Export Configuration (Page 45).</p>
Settings Icon	<p>Enables to perform the following tasks:</p> <ol style="list-style-type: none"> 1. You can specify SIMATIC S7 Connector user credentials which you define in Industrial Edge Databus Configurator. 2. You can specify configuration version of the SIMATIC S7 Connector Configurator. <p>For more information, refer Configure Settings (Page 40).</p>

Working with SIMATIC S7 Connector Configurator

The SIMATIC S7 Connector Configurator allows you to add field devices to Industrial Edge Device and create data point lists. It supports multiple user access with the following scenarios:

- One user can access SIMATIC S7 Connector Configurator on multiple Industrial Edge Device (IED).
But the user cannot access SIMATIC S7 Connector Configurator on the same IED in multiple tabs of same browser or multiple browsers.
- Two users can access SIMATIC S7 Connector Configurator on two different IEDs simultaneously.
When the two users access SIMATIC S7 Connector Configurator on the same IED, a message is displayed to the second user that the IED is in use already. Therefore, the second user cannot access SIMATIC S7 Connector Configurator.

Note

When you perform

- add, edit, or delete of a data source,
- add, edit, import, or delete of a tag,

you must click "Deploy" button to save the changes on the Industrial Edge Runtime of the SIMATIC S7 Connector.

6.1 Managing Data Sources

A data source is a field device which provides the data. You can configure the SIMATIC S7 Connector to the field devices to consume the acquired data in the Industrial Edge Device for value creation.

The SIMATIC S7 Connector Configurator supports the following three communication channels:

1. S7-Protocol (S7-300/400/1200/1500)
2. Optimized S7-Protocol (S7-1200/1500)
3. OPC UA (OPC Server)

6.1.1 Configure S7-Protocol (S7-300/400/1200/1500) Data Source

You can configure S7-Protocol (S7-300/400/1200/1500) controller data source in the SIMATIC S7 Connector Configurator. The Configurator allows you to configure the S7-Protocol (S7-300/400/1200/1500) communication channel to the SIMATIC PLCs for data acquisition.

Example Scenario

A Plant Administrator or Industrial Edge Box Operator would like to configure the data acquisition from the S7-Protocol (S7-300/400/1200/1500) controller, and subsequently would like to create value from the acquired data.

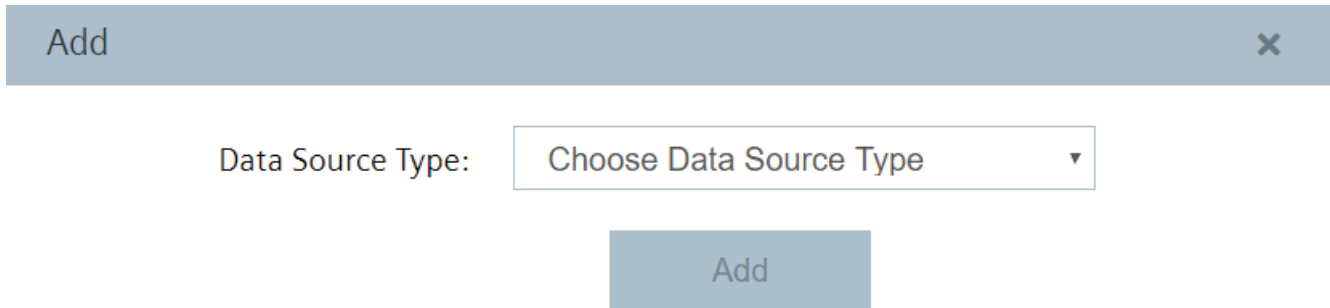
Prerequisite

The SIMATIC S7 Connector Configurator must be running.

Procedure

To configure S7-Protocol (S7-300/400/1200/1500) controller data source, follow these steps:

1. Launch the SIMATIC S7 Connector Configurator.
The Configurator home page is displayed.
2. Click "Add Data Source" in the upper-left corner.
The "Add" dialog box is displayed as follows:



- Select the S7-Protocol (S7-300/400/1200/1500) data source type from the "Data Source Type" drop-down.
The fields are displayed as follows:

Add
✕

Data Source Type:

Name:

IP Address:

Rack Number:

Slot Number:

PLC Type:

- Complete the following fields:

Field Name	Definition
Name	Defines the name of the data source.
IP Address	Defines the IP address of the S7-Protocol (S7-300/400/1200/1500) controller with the desired data points.
Rack Number	Defines the rack of the S7-Protocol (S7-300/400/1200/1500) controller. The default value is 0.
Slot Number	Defines the slot of the S7-Protocol (S7-300/400/1200/1500) controller. The default value is 1.
PLC Type	Specifies the PLC type.
Full Text Alarms	Enables the full text alarms. This field is displayed when you select "PLC Type" as 300/400. This checkbox is enabled only when project runtime is not running.

*All the fields are mandatory.

- Click "Add".
The data source is added and displayed in the "Data Source" table.

6.1.2 Configure OPC-UA Data Source

You can configure OPC-UA server data source in the SIMATIC S7 Connector Configurator. The Configurator allows you to configure the OPC-UA communication channel for data acquisition using different OPC-UA messaging modes.

Example Scenario

A Plant Administrator or Industrial Edge Box Operator would like to configure the data acquisition from the PLC's OPC-UA Server, and subsequently would like to create value from the acquired data.

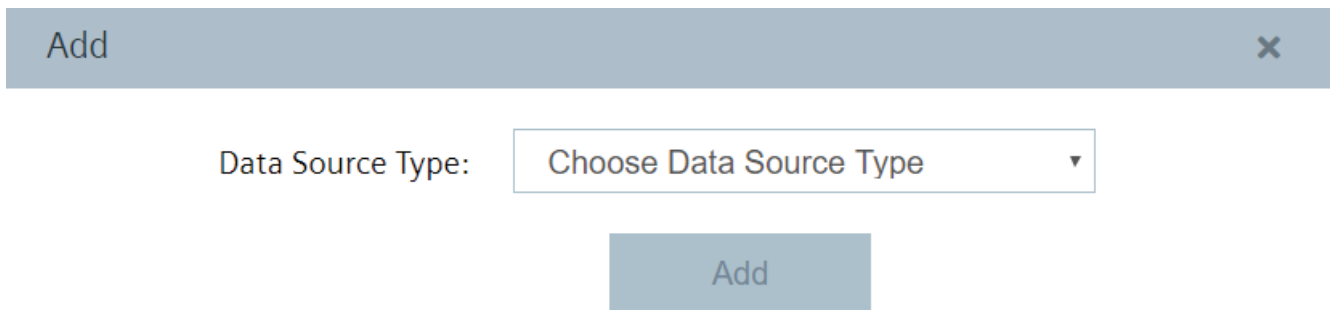
Prerequisite

The SIMATIC S7 Connector Configurator must be running.

Procedure

To configure OPC-UA server data source, follow these steps:

1. Launch the SIMATIC S7 Connector Configurator.
The Configurator home page is displayed.
2. Click "Add Data Source" in the upper-left corner.
The "Add" dialog box is displayed as follows:



3. Select the OPC UA (OPC Server) data source type from the "Data Source Type" drop-down.
The fields are displayed as follows:

Add×

Data Source Type:	<input type="text" value="OPC UA (OPC Server)"/>
Name:	<input type="text" value="Enter PLC Name"/>
OPC-UA URL:	<input type="text" value="opc.tcp://"/>
Port number:	<input type="text" value="Enter Port Number"/>
Messaging Mode:	<input type="text" value="Choose Messaging Mode"/>
Authentication Mode:	<input type="text" value="Choose Authentication Mode"/>

6.1 Managing Data Sources

4. Complete the following fields:

Field Name	Definition
Name	Defines the name of the data source.
OPC-UA URL	Defines the IP address of the OPC-UA server with the desired data points.
Port number	Defines the port number of the IP address of the OPC UA server.
Authentication Mode	Specifies the authentication. The available options are as follows: 1. "Anonymous": On selecting this option, you do not need any authentication. 2. "UserID & Password": On selecting this option, you must enter the username and password that you used when configuring the OPC-UA server in the TIA Portal.
User ID	Defines the username that you used when configuring the OPC-UA server in the TIA Portal. This field appears when authentication mode is selected as "UserID & Password".
Password	Defines the password that you used when configuring the OPC-UA server in the TIA Portal. This field appears when authentication mode is selected as "UserID & Password".
Messaging Mode	Specifies the messaging mode. The available options are as follows: 1. "None": It specifies no security. 2. "Sign": It specifies secure communication with signed client and server certificates. 3. "Sign and Encrypt": It specifies secure communication with signed client and server certificates with additional security where the data is encrypted from server application and sent to the client.
Security Policy	Specifies the security policies. The available options are as follows: 1. Basic128Rsa15 2. Basic256 3. Basic256Sha256 You must ensure that your selected security policy matches the algorithm that was used to generate the certificates from the TIA portal. This field appears when messaging mode is selected as "Sign" or "Sign and Encrypt".
Client PKCS12 file	Specifies the client certificate which is exported from the TIA portal. This field appears when a security policy is selected in "Security Policy" field. You can protect the Project by selecting the "Project Protection" option in the TIA portal. Subsequently, the "Certificate Manager" is enabled to export the certificate in '.p12' format. When you export the "Client PKCS12 file" from the TIA portal, a password window is displayed. You can define your own password. The same password is used in "PKCS12 Import Password" field in the SIMATIC S7 Connector Configurator.

Field Name	Definition
PKCS12 Import Password	Defines the password that you used when exporting the "Client PKCS12 file" from the TIA portal. This is used to extract the client certificate data in the backend. This field appears when a file is selected in "Client PKCS12 file" field.
Server DER certificate	Specifies the server certificate which is exported from the TIA portal. This field appears when a security policy is selected in "Security Policy" field. You can protect the Project by selecting the "Project Protection" option in the TIA portal. Subsequently, the "Certificate Manager" is enabled to export the certificate in '.der' format.

*All the fields are mandatory.

5. Click "Add".

The data source is added and displayed in the "Data Source" table.

For more information on client and server certificates, refer How to Generate OPC-UA Client and Server Certificates? (Page 65)

Note

- It is not mandatory to sync time between IED and PLC with UTC. By default, the IED time is published in case of OPC-UA Tags.
- If you configure secure OPC-UA connection using certificates, you must ensure that the certificates generated in the PLC must not be of future time. (PLC/OPC-UA server time should not be future time.)

Note

OPC-UA Support for Non-Siemens PLC

SIMATIC S7 Connector Configurator supports reading of the tag values from non-Siemens PLC for OPC-UA connection. It supports PLC Type: **OMRON NX102-1200** with Firmware version: V1.31.

When you configure non-Siemens PLCs for example OMRON, you must provide the address of the tag in following format:

```
ns=urn:OMRON:NxOpcUaServer:FactoryAutomation;s="<Tag_name>"
```

OPC-UA connection with authentication and messaging mode is not supported for OMRON PLC.

6.1.3 Configure Optimized S7-Protocol (S7-1200/1500) Data Source

You can configure Optimized S7-Protocol (S7-1200/1500) controller data source in the SIMATIC S7 Connector Configurator. The Configurator allows you to configure the Optimized S7-Protocol (S7-1200/1500) communication channel to the SIMATIC PLCs for data acquisition.

Example Scenario

A Plant Administrator or Industrial Edge Box Operator would like to configure the data acquisition from the Optimized S7-Protocol (S7-1200/1500) controller, and subsequently would like to create value from the acquired data.

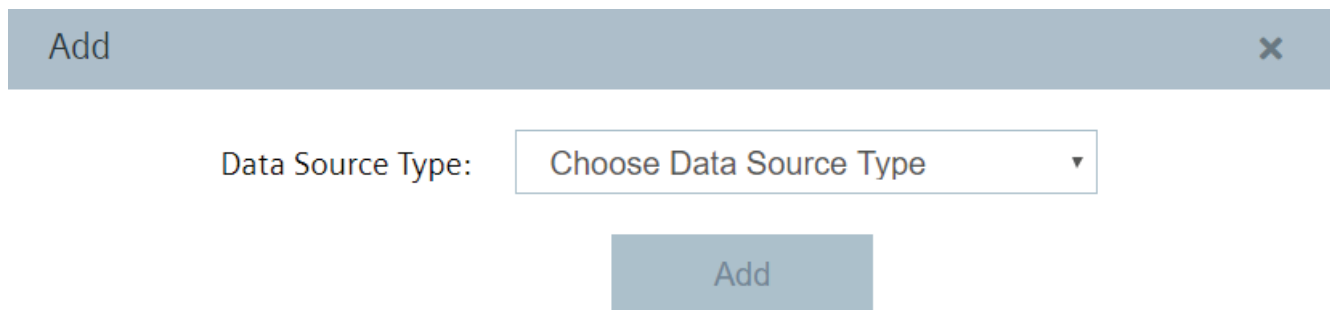
Prerequisite

The SIMATIC S7 Connector Configurator must be running.

Procedure

To configure Optimized S7-Protocol (S7-1200/1500) controller data source, follow these steps:

1. Launch the SIMATIC S7 Connector Configurator.
The Configurator home page is displayed.
2. Click "Add Data Source" in the upper-left corner.
The "Add" dialog box is displayed as follows:



- Select the Optimized S7-Protocol (S7-1200/1500) data source type from the "Data Source Type" drop-down.
The fields are displayed as follows:

Add
✕

Data Source Type:

Name:

PLC Type:

IP Address:

Access Level:

Access Password:

- Complete the following fields:

Field Name	Definition
Name	Defines the name of the data source.
PLC Type	Specifies the PLC type.
IP Address	Defines the IP address of the Optimized S7-Protocol (S7-1200/1500) controller with the desired data points.
Access Level	Specifies the access level. The available options are as follows: 1. No Access(complete protection) 2. Full Access(no protection)
Access Password	Defines the access password. This field is enabled when you select "Access Level" as No Access(complete protection). For more information, refer How to Configure Access Password? (Page 73)

*All the fields are mandatory.

- Click "Add".
The data source is added and displayed in the "Data Source" table.

6.1.4 Edit Data Source


You can edit a data source in the SIMATIC S7 Connector Configurator and update the required details. The updated details are configured for the data source.

Prerequisite

- The SIMATIC S7 Connector Configurator must be running.
- A data source must be available.

Procedure

To edit a data source, follow these steps:

1. Launch the SIMATIC S7 Connector Configurator.
The Configurator home page is displayed.
2. Click  under the "Actions" column of the data source that you want to edit.
The "Edit Data Source" dialog box is displayed.
3. Modify the relevant details.

Note

You cannot edit the "Connection Name" and "IP Address" for a data source once it is at Industrial Edge Runtime.

4. Click "Save".
The data source is modified and displayed in the "Data Source" table.

6.1.5 Delete Data Source

You can delete a data source in the SIMATIC S7 Connector Configurator. The "Data Source" table is updated with the updated list of the data sources. You must deploy the project using "Deploy" button to reflect the deleted configuration.

Prerequisite

- The SIMATIC S7 Connector Configurator must be running.
- A data source must be available.

Procedure

To delete a data source, follow these steps:

1. Launch the SIMATIC S7 Connector Configurator.
The Configurator home page is displayed.
2. Select the data source records that you want to delete.

3. Click "Delete" in the upper-left corner.
A confirmation message is displayed.
4. Click "OK".
The data sources are deleted and removed from the "Data Source" table.
The deleted data is hidden and only after click on "Deploy" it is deleted from the Industrial Edge Runtime.

6.2 Managing Tags/Data Points

6.2.1 Add Tags

You can add a data point or tag from a data source in the SIMATIC S7 Connector Configurator.

Prerequisite

- The SIMATIC S7 Connector Configurator must be running.
- A data source must be available.

Procedure

To add data points from a data source, follow these steps:

1. Launch the SIMATIC S7 Connector Configurator.
The Configurator home page is displayed.
2. Click **+** under the "Actions" column of the data source to which you want to add a tag.
The "Add Tags" dialog box is displayed based on the data source as follows:
For S7-Protocol (S7-300/400/1200/1500):

Add Tags ×

Name	Comments	Address	Data Type	String Length	Array	Array Size	Acquisition Cycle	Acquisition Mode	Access Mode	Actions
<input type="text" value="Name"/>	<input type="text" value="Enter Com"/>	<input type="text" value="Enter Ad"/>	<input style="border: none; background-color: #f0f0f0; text-align: center; font-size: small; font-weight: bold; color: #666; padding: 2px 5px; border: 1px solid #ccc; border-radius: 2px; width: 100%;" type="text" value="Choc"/>	<input type="text" value="Enter Strin"/>	<input type="checkbox"/>	<input type="text" value="Array Siz"/>	<input style="border: none; background-color: #f0f0f0; text-align: center; font-size: small; font-weight: bold; color: #666; padding: 2px 5px; border: 1px solid #ccc; border-radius: 2px; width: 100%;" type="text" value="Choose"/>	<input style="border: none; background-color: #f0f0f0; text-align: center; font-size: small; font-weight: bold; color: #666; padding: 2px 5px; border: 1px solid #ccc; border-radius: 2px; width: 100%;" type="text" value="Choose"/>	<input style="border: none; background-color: #f0f0f0; text-align: center; font-size: small; font-weight: bold; color: #666; padding: 2px 5px; border: 1px solid #ccc; border-radius: 2px; width: 100%;" type="text" value="Choose"/>	<input style="border: none; background-color: #f0f0f0; text-align: center; font-size: small; font-weight: bold; color: #666; padding: 2px 5px; border: 1px solid #ccc; border-radius: 2px; width: 100%;" type="button" value="+"/>

Note

- Only the 'Char' data type supports the array feature for S7-Protocol (S7-300/400/1200/1500) data source in the configurator, but the data is published as simple scalar values for each array elements.
- If you have configured string datatype in TIA Portal, then by default the string length is set to 210 for S7-Protocol (S7-300/400/1200/1500).

With "Configurator Version" as 1.2, all the array elements are published as scalar Char value. Any value change notification of one child element results in publishing of only that child tag value.

```
ie/m/j/simatic/v1/s7c1/dp : msg.payload : Object
  ▼ object
    seq: 1
    ▼ connections: array[1]
      ▼ 0: object
        name: "S7"
        type: "S7"
        ▼ dataPoints: array[1]
          ▼ 0: object
            name: "default"
            topic: "ie/d/j/simatic/v1/s7c1/dp/r/S7/default"
            publishType: "bulk"
            ▼ dataPointDefinitions: array[4]
              ▼ 0: object
                name: "CharArr/0"
                id: "104"
                dataType: "Char"
              ▼ 1: object
                name: "CharArr/1"
                id: "105"
                dataType: "Char"
              ▶ 2: object
              ▶ 3: object
```

6.2 Managing Tags/Data Points

```
ie/dj/simatic/v1/s7c1/dp/r/S7/default : msg.payload : Object
  ▼ object
    seq: 1
  ▼ vals: array[4]
    ▼ 0: object
      id: "104"
      qc: 0
      qx: 28
      ts: "2021-04-01T06:20:06.011Z"
      val: "A"
    ▼ 1: object
      id: "105"
      qc: 0
      qx: 28
      ts: "2021-04-01T06:20:06.011Z"
      val: "B"
    ▼ 2: object
      id: "106"
      qc: 0
      qx: 28
      ts: "2021-04-01T06:20:06.011Z"
      val: "C"
    ▶ 3: object
```

For Optimized S7-Protocol (S7-1200/1500):

Add Tags
✕

Name	Comments	Address	Data Type	String Length	Acquisition Cycle	Acquisition Mode	Access Mode	Actions
Name	Enter Com	Enter Ad	Choo ▾	Enter Strin	Choose ▾	Choose ▾	Choose ▾	+

Add Tags

Note

- Optimized S7-Protocol (S7-1200/1500) data source does not support the array feature.
- If you have configured string datatype in TIA Portal, then by default the string length is set to 254 for Optimized S7-Protocol (S7-1200/1500).

For OPC UA (OPC Server):

Add Tags ×

Name	Comments	Address	Data Type	Array	Acquisition Cycle	Acquisition Mode	Access Mode	Actions
Name	Enter Comments	Enter Address	Cho ▾	<input type="checkbox"/>	Choose ▾	Choose ▾	Choose ▾	+

Add Tags

Note


- The array data for OPC UA (OPC Server) data source is published as simple scalar value as String and Pipe separated.
- OPC UA (OPC Server) data source does not support the "String Length" field.

3. Complete the following fields:

Field Name	Definition
Name	Defines the name of the data point.
Comments	Defines the comments.
Address	Defines the address of the data point in the controller or on the server. For an OPC-UA server connection, the address must have the following syntax: ns=3; s=<DATA_BLOCK>.<TAG> Replace the "<DATA_BLOCK>" and "<TAG>" placeholders with the corresponding address of the data point.
Data Type	Specifies the data point type. For more information on the data types, refer Data Types (Page 77). The "Data Type" drop-down displays the data type options based on the data source as follows: <ul style="list-style-type: none"> S7-Protocol (S7-300/400/1200/1500) supports Bool, Int, Byte, Dint, Real, String, Word, USInt, UInt, UDInt, Dword, Date, Time, Char, DateTime, and Char Array. For more information on String data type, refer How to Configure String Datatype in S7-Protocol (S7-300/400/1200/1500)? (Page 64) Optimized S7-Protocol (S7-1200/1500) supports Bool, Int, Byte, Dint, Real, String, Word, Lint, Sint, USInt, UInt, UDInt, ULInt, Lreal, Dword, Lword, Date, Time, TOD, Ltime, LTOD, Char, and DateTime. OPC UA (OPC Server) supports Bool, Int, Byte, Dint, Real, String, Word, Lint, Sint, USInt, UInt, UDInt, ULInt, Lreal, Dword, Lword, Char, Bool Array, Int Array, Dint Array, String Array, Word Array, Lint Array, Sint Array, UInt Array, Udint Array, ULInt Array, Real Array, Dword Array, LWord Array, and Lreal Array.
String Length	Defines the string length to read the correct tag value. If you have configured fixed length string datatype in TIA Portal, then you must enter the correct string length in this field to read the correct tag value.
Array	Enables the array feature. Array feature is used to accumulate different PLC values in one array as required.
Array Size	Defines the size of the array.
Acquisition Cycle	Specifies the acquisition cycle with which the data is sent to the Databus. The available options are as follows: <ul style="list-style-type: none"> 10 milliseconds 50 milliseconds 100 milliseconds 250 milliseconds 500 milliseconds 1 second 2 second 5 second 10 second As a rule, the '1 second' acquisition cycle is used. Only OPC-UA supports highspeed acquisition cycle support of 10 milliseconds and 50 milliseconds.

Field Name	Definition
Acquisition Mode	Specifies the acquisition mode of the data. The available options are as follows: <ul style="list-style-type: none"> CyclicContinuous CyclicOnChange OPC-UA and Optimized S7 Protocol (S7-1200/1500) data source type support only "CyclicOnChange".
Access Mode	Specifies the access mode of the tags. You can configure the tags as readable, writable, or both readable-writable. The available options are as follows: <ul style="list-style-type: none"> "Read": This access mode is applicable only for 'Read' configured tags. "Read and Write": With this access mode, the value can be 'Read' and 'Write'.

*All the fields are mandatory.

You can add multiple data points as required by clicking  under the "Actions" column.

- Click "Add Tags".
The data points are added below the data source.

6.2.2 Edit Tags


You can edit a data point or tag in the SIMATIC S7 Connector Configurator and update the required details. The updated details are configured for the data source.

Prerequisite

- The SIMATIC S7 Connector Configurator must be running.
- A data source and a data point must be available.

Procedure

To edit a data point, follow these steps:

- Launch the SIMATIC S7 Connector Configurator.
The Configurator home page is displayed.
- Click  under the "Actions" column of the data point that you want to edit.
The "Edit Tag" dialog box is displayed.

Note

You cannot edit the "Tag Name", "Acquisition Mode", "Data Type", and "Access Mode" for a tag once it is at Industrial Edge Runtime.

- Modify the relevant details.
- Click "Save".
The data point is modified and displayed in the "Data Source" table.

6.2.3 Import Tags

The SIMATIC S7 Connector Configurator allows you to import multiple data points or tags simultaneously from a TIA Portal project. You can select a file with the datapoints from the TIA Portal and export it from the TIA Portal and import it to the SIMATIC S7 Connector Configurator. The file must correspond to the '.xml' file type.

In addition, S7-Protocol (S7-300/400/1200/1500) data source supports import of '.xls' file. The '.xls' format support is not from TIA Portal but from TagConverter Tool.

For more information, refer How to Export Tags from TIA Portal? (Page 58)

If tag names, that you have in the imported tag file, contain specific special characters, then these are replaced by the characters as mentioned in the following table:

Character in Tag name in TIA	Mapped to SIMATIC S7 Connector
"	(It is removed.)
.	_
#	_1
[_2
]	_3
:	_4
<	_5
>	_6
+	_7
/	_8
\	_9
\$	_0


This replacement is done to handle download failures, since these special characters are not supported in SIMATIC S7 Connector.

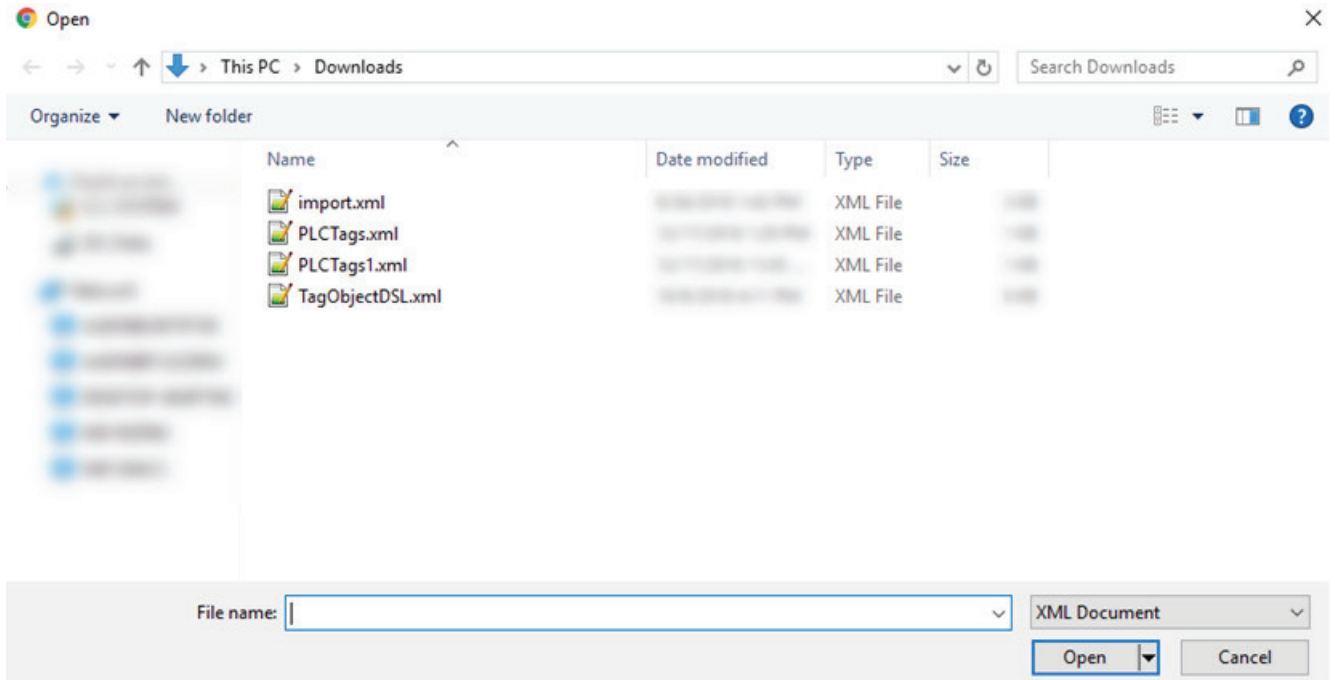
Prerequisite

- The SIMATIC S7 Connector Configurator must be running.
- The required data points file must be available.

Procedure

To import the data points, follow these steps:

1. Launch the SIMATIC S7 Connector Configurator.
The Configurator home page is displayed.
2. Click  under the "Actions" column of the data source to which you want to import the data points.
The "Open" dialog box is displayed as follows:



6.2 Managing Tags/Data Points

- 3. Select the required file and click "Open".
The "Import" dialog box is displayed as follows:



- ✓ 18 tags found.
- ✗ 1 tags have duplicate tagnames. 5 tags have invalid characters.

Details



It displays the valid tags, duplicate tags, existing tags already loaded in the configurator, tags having invalid data types, and tags with invalid characters. You can click "Details" hyperlink to view the invalid tags details.

- 4. Click "OK".
The "Import Tags" dialog box is displayed as follows:

A screenshot of the 'Import Tags (128)' dialog box. It features three drop-down menus at the top: 'Acquisition Cycle' set to '1 second', 'Acquisition Mode' set to 'CyclicOnChange', and 'Access Mode' set to 'Read'. Below these is a table with columns for 'Name', 'Address', and 'Data Type'. Each row has a checkbox on the left. The table lists several tags, including 'AlwaysFALSE', 'AlwaysTRUE', and various 'Clock' tags. At the bottom of the dialog is an 'Add Tags' button.

<input type="checkbox"/>	Name	Address	Data Type
<input type="checkbox"/>	AlwaysFALSE	%M1.3	Bool
<input type="checkbox"/>	AlwaysTRUE	%M1.2	Bool
<input type="checkbox"/>	Clock_0_5Hz	%M0.7	Bool
<input type="checkbox"/>	Clock_0_625Hz	%M0.6	Bool
<input type="checkbox"/>	Clock_10Hz	%M0.0	Bool
<input type="checkbox"/>	Clock_1Hz	%M0.5	Bool
<input type="checkbox"/>	Clock_1_25Hz	%M0.4	Bool

- 5. Specify the acquisition cycle, acquisition mode, and access mode using "Acquisition Cycle", "Acquisition Mode", and "Access Mode" drop-down respectively.
- 6. Mark true against the tags that you want to import and click "Add Tags".
The tags are added below the data source.

6.2.4 Browse Tags

The SIMATIC S7 Connector Configurator allows you to browse the data points or tags from a PLC and add them to the data source. Only OPC-UA and Optimized S7-Protocol (S7-1200/1500) data source types support the Browse Tags functionality.

If tag names, that you have browsed, contain specific special characters, then these are replaced by the characters as mentioned in the following table:

Character in Tag name in TIA	Mapped to SIMATIC S7 Connector
"	(It is removed.)
.	_
#	_1
[_2
]	_3
:	_4
<	_5
>	_6
+	_7
/	_8
\	_9
\$	_0


This replacement is done to handle download failures, since these special characters are not supported in SIMATIC S7 Connector.

Prerequisite

- The SIMATIC S7 Connector Configurator must be running.
- A data source must be available.


Procedure

To browse the data points, follow these steps:

1. Launch the SIMATIC S7 Connector Configurator.
The Configurator home page is displayed.
2. Click  under the "Actions" column of the data source of which you want to browse the data points.
The "Browse" dialog box is displayed as follows:



 750 tags found.

 4 tags have duplicate tagnames. 100 tags already exist. 3 tags have invalid characters.

[Details](#)




It displays the valid tags, duplicate tags, existing tags already loaded in the configurator, tags having invalid data types, and tags with invalid characters. You can click "Details" hyperlink to view the invalid tags details.

3. Click "OK".

The "Browsed Tags" dialog box is displayed as follows:

Name	Address	Data Type
Tag_1	Tag_1	Int
Tag_10	Tag_10	Int
Tag_100	Tag_100	Int

You can use Datablock and tag filters to filter the required tags. A datablock is an array of tags. The "Datablock Filter" drop-down field lists all the datablocks available in the PLC. You can use the "Datablock Filter" drop-down to select the required datablocks. Once you select the datablocks and click , the tags that are present in the selected datablocks are displayed. In addition, you can use "Tag Filter" field to further filter the tags.

Note

- Only Optimized S7-Protocol (S7-1200/1500) supports "Datablock Filter" and "Tag Filter" fields.
- Browsing data is cached for already browsed Optimized S7-Protocol (S7-1200/1500) connection. But only the last browsed data depending on filters is cached.

4. Specify the acquisition cycle, acquisition mode, and access mode using "Acquisition Cycle", "Acquisition Mode", and "Access Mode" drop-down respectively.
5. Mark true against the tags that you want to add and click "Add Tags". The tags are added below the data source.

6.2.5 Bulk Publish Tags

The SIMATIC S7 Connector Configurator allows you to group the data points or tags from the PLCs and publish them together as a group using the "Bulk Publish" functionality. When any tag value is updated in a group, then only the updated tag is published. All tags in same connection fall under one group and the group name is 'default'.

The "Bulk Publish" feature is enabled by default. Click  in the upper-right corner. The "Settings" dialog box is displayed as follows:

Settings
✕

Databus ServiceName:

UserName:

Password:

Configuration Version:

Bulk Publish:

Save

You cannot update the "Bulk Publish" checkbox as single publish is not supported.

Supported Quantity Structure

The Supported Quantity Structure for SIMATIC S7 Connector is as follows:

Connection Type	Acquisition Cycle (ms)	Publish Mode	Maximum Configurable Tags
S7-Protocol (S7-300/400/1200/1500)	1000	Bulk	6000
Optimized S7-Protocol (S7-1200/1500)	1000	Bulk	6000
OPC UA (OPC Server)	1000	Bulk	4000
S7-Protocol (S7-300/400/1200/1500)	100	Bulk	600
Optimized S7-Protocol (S7-1200/1500)	100	Bulk	600
OPC UA (OPC Server)	10	Bulk	50

The maximum connection supported by each driver is as follows:

- S7-Protocol (S7-300/400/1200/1500): 20
- Optimized S7-Protocol (S7-1200/1500): 8
- OPC UA (OPC Server): 20

Note**Test conditions**

Mqtt Clients: The benchmarking result was obtained using one MQTT publisher client (S7 Connector) and one lightweight MQTT subscriber client (open source MQTT Node.js client) in a test run of five hours. For a higher number of publishers or subscribers, the supported tags will be lower.

Connections: The benchmarking result was obtained with below mentioned number of connections:

- S7-Protocol (S7-300/400/1200/1500): 4
- Optimized S7-Protocol (S7-1200/1500): 8
- OPC UA (OPC Server): 6

Above quantity structure can be achieved by keeping number of connections less in the Bulk Publish mode. For higher number of connections, the supported tags will be lower.

Optimized S7-Protocol (S7-1200/1500) NFR was performed with all unique PLCs to get the optimal result.

6.2.6 Delete Tags

You can delete a tag/data point in the SIMATIC S7 Connector Configurator. The "Data Source" table is updated with the updated list of the data points. You must deploy the project using "Deploy" button to reflect the deleted configuration.

Prerequisite

- The SIMATIC S7 Connector Configurator must be running.
- A data source and data point must be available.


Procedure

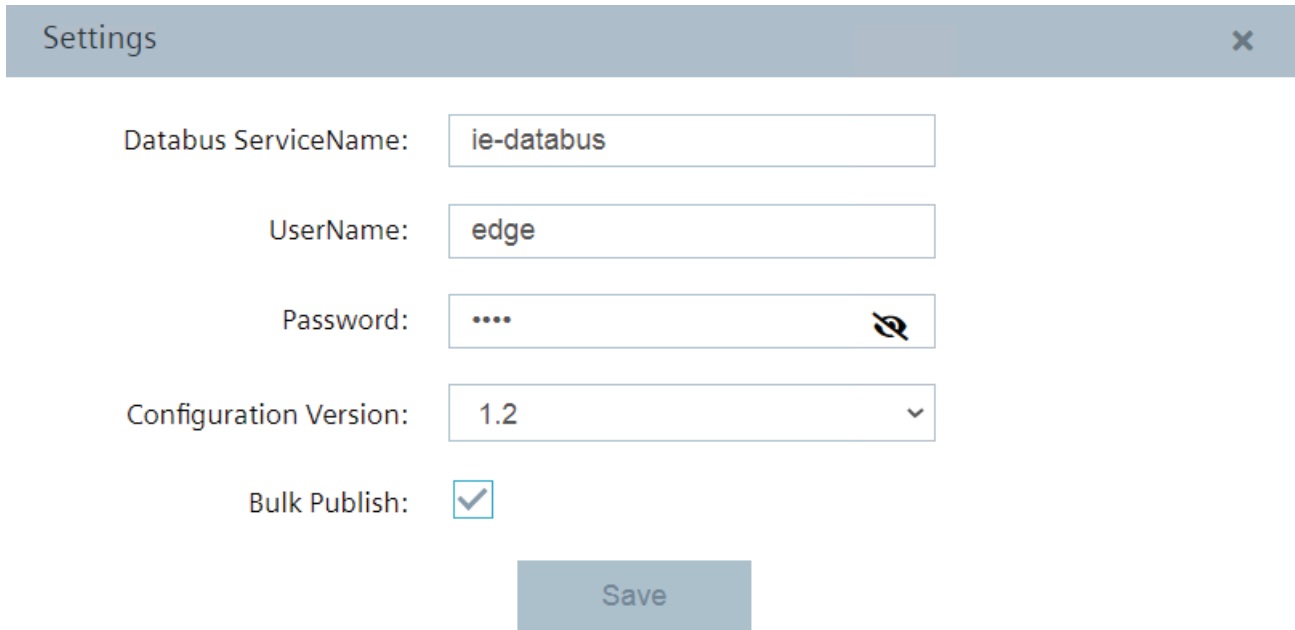
To delete a data point, follow these steps:

1. Launch the SIMATIC S7 Connector Configurator.
The Configurator home page is displayed.
2. Select the tag/data point records that you want to delete.
3. Click "Delete" in the upper-left corner.
A confirmation message is displayed.
4. Click "OK".
The data points are deleted and removed from the "Data Source" table.
The deleted data is hidden and only after click on "Deploy" it is deleted from the Industrial Edge Runtime.

6.3 Managing Project

6.3.1 Configure Settings


When you click  in the upper-right corner. The "Settings" dialog box is displayed as follows:




Settings

Databus ServiceName:

UserName:

Password: 

Configuration Version: 

Bulk Publish:

Save

You can perform the following tasks:

Define IE Databus Service Name

You can define the IE Databus service name in this field.

Specify Username and Password

You can connect the SIMATIC S7 Connector Configurator to Industrial Edge Databus to publish the data. The Industrial Edge Databus allows you to create a user and topics to this user. You must create topics for alarms and tags data to the user in the Industrial Edge Databus. The topic name must be as follows:

- "ie/m/j/simatic/v1/s7c1/dp" for tags metadata
where ie: industrial edge, m: metadata, j: json, s7c1: SIMATIC S7 Connector instance 1, dp: datapoint.
- "ie/d/j/simatic/v1/s7c1/dp/r/#" for tags data
where ie: industrial edge, d: data, j: json, s7c1: SIMATIC S7 Connector instance 1, dp: datapoint
- "ie/m/j/simatic/v1/s7c1/ev" for alarms metadata
where ie: industrial edge, m: metadata, j: json, s7c1: SIMATIC S7 Connector instance 1, ev: event.

- `ie/d/j/simatic/v1/s7c1/ev/#` for alarms data
where `ie`: industrial edge, `d`: data, `j`: json, `s7c1`: SIMATIC S7 Connector instance 1, `dp`: datapoint
- `ie/m/j/simatic/v1/s7c1/status` for metadata client status
where `ie`: industrial edge, `m`: metadata, `j`: json, `s7c1`: SIMATIC S7 Connector instance 1
- `ie/d/j/simatic/v1/s7c1/dp/r/status` for tag data provider client status
where `ie`: industrial edge, `d`: data, `j`: json, `s7c1`: SIMATIC S7 Connector instance 1, `dp`: datapoint
- `ie/d/j/simatic/v1/s7c1/ev/status` for alarm data provider client status
where `ie`: industrial edge, `d`: data, `j`: json, `s7c1`: SIMATIC S7 Connector instance 1, `ev`: event

When you create a user in Industrial Edge Databus, you define a username and password for the user. You must specify this username and password combination in the "UserName" and "Password" fields. Using this credential, the SIMATIC S7 Connector Configurator establishes connection to Industrial Edge Databus.

Specify Configuration Version

You can specify the configuration version of the SIMATIC S7 Connector Configurator as required. The available options are as follows:

- 1.1 (Deprecated)
- 1.2

For more information, refer Configuration Version 1.1 vs 1.2 (Page 77).

Bulk Publish

By default, the bulk publish is always enabled. You cannot update the "Bulk Publish" checkbox as single publish is not supported. For more information, refer Bulk Publish Functionality (Page 37).

6.3.2 Deploy Project

Once you configure the data sources and tags/data points, you can deploy the project to the Industrial Edge Runtime. You can select the required data points of each data source that you want to deploy, and only the selected data points are deployed. When you deploy the data points to the Industrial Edge Runtime, the configuration is saved and loaded to the Industrial Edge Runtime.

The SIMATIC S7 Connector Configurator allows you to edit or delete the data points or import the new data points without stopping the Industrial Edge Runtime. When you deploy and start the project using "Deploy" and "Start Project" buttons respectively, you can again use the "Deploy" button to deploy the changes while the Industrial Edge Runtime is still running.

When you deploy the changes while the Industrial Edge Runtime is running, the metadata and metadata client 'birth' message are also published. For more information on metadata publish, refer JSON structure for Alarms and Tags Metadata? (Page 54)

While the Industrial Edge Runtime is running, you cannot perform the following tasks:

- Configuring OPC UA (OPC Server) data source with Authentication and Messaging Mode connection
- Configuring Full Text Alarms

6.3 Managing Project

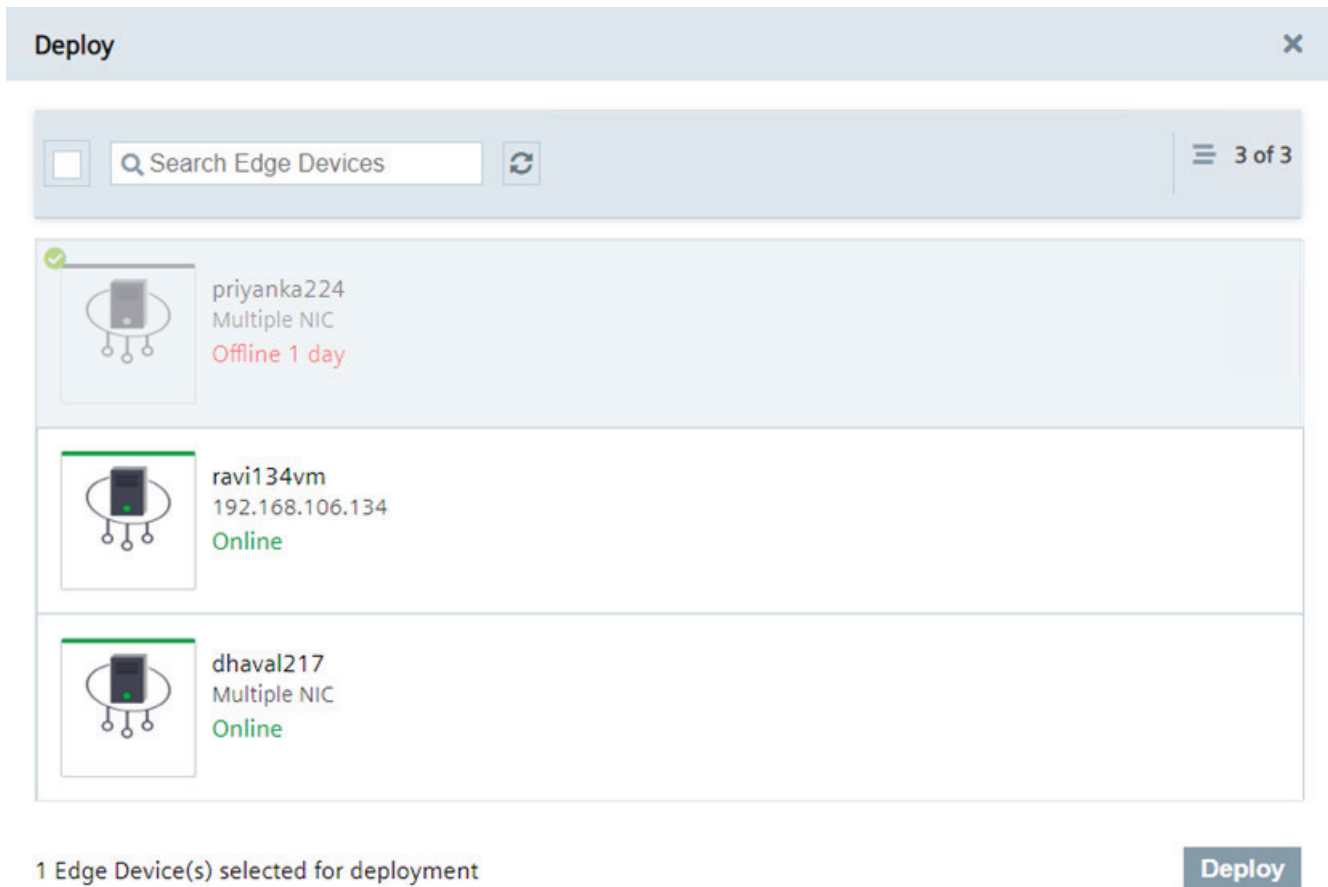
Prerequisite

- The SIMATIC S7 Connector Configurator must be running.
- A data source and a data point must be available.

Procedure

To deploy a data point, follow these steps:

1. Launch the SIMATIC S7 Connector Configurator.
The Configurator home page is displayed.
2. Mark true against the data points that you want to deploy.
3. Click "Deploy" in the upper-right corner.
The "Deploy" dialog box is displayed as follows:



4. Select the required Industrial Edge Devices on which you want to deploy the changes.
5. Click "Deploy".
The Configurator takes a while to deploy the project, and a success message is displayed subsequently. The data points are deployed and downloaded, and a green check mark is displayed next to the selected data points.

Note

1. You can deploy the changes only when the Industrial Edge Devices are in "Online" state.
 2. Mass deployment feature (deploying to multiple Industrial Edge Devices) is available only when Industrial Edge Runtime for the selected Industrial Edge Devices is not running.
 3. You must manually start, configure project when Industrial Edge Runtime is started, and stop the project by launching the UI of that Industrial Edge Device.
 4. Before performing mass deployment, you must ensure that same user credentials and topic are already added in the IE Databus Configurator for all the selected Industrial Edge Devices where you want to perform the mass deployment.
-

6.3.3 Start Project

Once you successfully deploy the project to the Industrial Edge Runtime, the "Start Project" button is enabled. You can start the project using this button. When you start the project, you start the Industrial Edge Runtime. The Industrial Edge Runtime reads the data point values and sends this data to the Industrial Edge Databus.

Once you start the Industrial Edge Runtime, the applications that have access to the Industrial Edge Databus can evaluate this data and use it for their own purposes.

When Industrial Edge Runtime is started, the 'birth' message from Tag and Alarm Provider client is published to the broker (IE Databus) indicating clients are connected to the broker.

Prerequisite

- The SIMATIC S7 Connector Configurator must be running.
- A data source and a data point must be available.
- A data point must be deployed to the Industrial Edge Runtime as described in Deploy Project (Page 41).

Procedure

To start a project, follow these steps:

1. Launch the SIMATIC S7 Connector Configurator.
The Configurator home page is displayed.
2. Click "Start Project" in the upper-right corner.
The Configurator takes a while to start the project and a success message is displayed subsequently.

6.3.4 Stop Project

Once you successfully start the project, and the Industrial Edge Runtime starts the data transfer to the Industrial Edge Databus, the SIMATIC S7 Connector Configurator converts the "Start Project" button to "Stop Project" button.

The "Stop Project" button allows you to stop the data transfer to the Industrial Edge Databus. You can use the "Stop Project" button anytime to stop the Industrial Edge Runtime.

When Industrial Edge Runtime is stopped, the 'will' message from Metadata, Tag, and Alarm Provider client is published to the broker (IE Databus) indicating that clients are disconnected from the broker.

A new start of the project is only possible after new deploy.

Prerequisite

- The SIMATIC S7 Connector Configurator must be running.
- A data source and a data point must be available.
- The Industrial Edge Runtime must be running as described in Start Project (Page 43).

Procedure

To stop a project, follow these steps:

1. Launch the SIMATIC S7 Connector Configurator.
The Configurator home page is displayed.
2. Click "Stop Project" in the upper-right corner.
The Configurator takes a while to stop the project, and a success message is displayed subsequently. The Industrial Edge Runtime stops the data transfer to the Industrial Edge Databus.

6.3.5 Verify Configuration



You can create a data flow using SIMATIC Flow Creator application to verify the configurations of the SIMATIC S7 Connector Configurator. The values can be displayed using the MQTT Broker Node. SIMATIC S7 Connector sends the data to the IE Databus using the following topic:

- For tags data: `ie/d/j/simatic/v1/s7c1/dp/r/#`
- For tags metadata: `ie/m/j/simatic/v1/s7c1/dp`
- For alarms data: `ie/d/j/simatic/v1/s7c1/ev/#`
- For alarms metadata: `ie/m/j/simatic/v1/s7c1/ev`
- For metadata client status: `ie/m/j/simatic/v1/s7c1/status`
- For tag data provider client status: `ie/d/j/simatic/v1/s7c1/dp/r/status`
- For alarm data provider client status: `ie/d/j/simatic/v1/s7c1/ev/status`

6.4 Import/Export Configuration

The SIMATIC S7 Connector Configurator allows you to import and export the configuration. This functionality is helpful when you want to use an existing configuration of one Industrial Edge Device (IED) in other IEDs. This enables to take the back-up of the configuration and restore it as required. You can export the configuration file from the required IED and import this configuration file in the desired IED where you want to use this configuration. The configuration file is a JSON file, and is exported in .json format.

The exported configuration file must not be modified externally. If the exported .json file is modified externally, then the file cannot be imported. This provides a unique means of verifying that the integrity of the file has been maintained.

To import and export the configuration, you can use  and  icons respectively present in the upper-right corner.

Note

You can import configuration files of V1.1 and V1.2. When you import V1.1 configuration file in V1.2, the payload is applied as per V1.1 settings. Single mode is changed to Bulk and tag Id is published in tag data payload.

Additional Information

7.1 How to Write Tags?

SIMATIC S7 Connector supports the 'Tag Write' functionality. This enables you to write a tag. You can add a topic in IE Databus in following format to write a tag:

```
ie/d/j/simatic/v1/s7c1/dp/w/<connection-name>
```

For example, `ie/d/j/simatic/v1/s7c1/dp/w/s7plus`

The JSON payload structure for Bulk subscribe is as follows:

```
{
  "seq": 1,
  "vals": [
    {
      "id": "105",
      "qc": 3,
      "ts": "2020-07-15T17:04:24.404Z",
      "val": 56
    },
    {
      "id": "106",
      "qc": 3,
      "ts": "2020-07-15T17:04:24.404Z",
      "val": "hello"
    }
  ]
}
```

Where, `seq`: unique sequence number of the payload.

`vals`: array of data points published in the payload.

`id`: unique identification of data point. You must fetch the tag ID from metadata payload based on the tag name.

`qc`: quality code. It is an optional field. It provides specific integer value to indicate the quality of the data point value.

`ts`: timestamp of the data point. It is an optional field. it is in ISO 8601 Zulu format.

`val`: value of the Tag. Based on the data type of the data point, the value can be simple scalar value.

Note

You can use the metadata to get the tag id published from the SIMATIC S7 connector for 'Tag Write' functionality. You can use this tag Id in the Publisher (MQTT) to write value for that tag.

For more information on Bulk Publish, refer Bulk Publish Tags. (Page 37)

For more information on topic format structure, refer Topic Structure for Data and Metadata (Page 48).

7.2 Topic Structure for Data and Metadata

This section describes the topic structure for Data and Metadata. The recommended structure of topic naming is as follows:

```
ie/{payloadType}/{encoding}/{msgStructureScheme}/
{msgStructureSchemeVersion}/{provideAppInstanceId}/{payloadMsgType}
```

and it is made of the following elements:

```
ie/{payloadType}/{encoding}/{msgStructureScheme}/
{msgStructureSchemeVersion}/{provideAppInstanceId}/{payloadMsgType}/
{accessmode}/{connectionname}/{collectionname}
```

The following table explains the above elements:

Element	Description	Possible Values
payloadType	This indicates what the payload contains.	'd' for data 'm' for metadata
encoding	This indicates payload encoding.	'j' for JSON
msgStructureScheme	This indicates payload format schema model.	simatic
msgStructureSchemeVersion	This indicates payload format schema version number	v1
provideAppInstanceId	This indicates unique id of provider app.	's7c1' for SIMATIC S7 Connector instance 1
payloadMsgType	This indicates payload message content.	'dp': DataPoints for PLC Variables (process image) 'ev': Events for PLC Fulltext Alarms, HMI Alarms
accessmode	This is an app specific element. This indicates the purpose of the payload for SIMATIC S7 Connector as it would support both read and write of Data Points.	'r' for Read from Connectors 'w' for write to Connectors
connectionname	This is an app specific element. This indicates unique name provided in Connector for a connection to a PLC in SIMATIC S7 Connector.	Paintshop1PLC
collectionname	This is an app specific element. This indicates the collection name.	'default' for Tags published in bulk mode

7.2.1 JSON structure for Alarms and Tags Data

You can create a topic in IE Databus to publish the alarms and tags data.

Tags Data

With the "Bulk Publish" feature, all tags data is published under single group with topic name as:

ie/d/j/simatic/v1/s7c1/dp/r/<connection-name>/default

The JSON structure is as follows:

```
ie/d/j/simatic/v1/s7c1/dp/r/opc/default : msg.payload : Object
  ▼ object
    seq: 1
    ▼ vals: array[4]
      ▼ 0: object
        id: "110"
        qc: 3
        ts: "2021-04-01T08:25:09.214Z"
        val: 0
      ▼ 1: object
        id: "111"
        qc: 3
        ts: "2021-04-01T08:25:09.214Z"
        val: "this is string"
      ▼ 2: object
        id: "112"
        qc: 3
        ts: "2021-04-01T08:25:09.214Z"
        val: 3.4000001549684603e+38
      ▼ 3: object
        id: "113"
        qc: 3
        ts: "2021-04-01T08:25:09.214Z"
        val: "100 | 200"
```

Where, `seq`: unique sequence number of the payload.

`vals`: array of data points published in the payload.

`id`: unique identification of data point. You must fetch the tag ID from metadata payload based on the tag name.

`qc`: quality code. It provides specific integer value to indicate the quality of the data point value.

With "Configurator Version" as 1.2, along with `qc`, `qx` is published which holds all the bits data: quality code, sub status, extended sub status, flags, and limit. `qc` is published in decimal value, for example, 192. You must convert it into binary to get the bit information.

Only if when any of the bits '0 to 5' and '8 to 15' is 1, the `qx` field is published.

Limits (bits 0,1)

Value	Meaning	Description
0	Ok	Data quality unrelated to limits.
1	Low limit violation	The value has exceeded its low limit.
2	High limit violation	The value has exceeded its high limit.
3	Constant	The value cannot move, no matter what the process does.

Sub-status "BAD" (sub-status bits 2..5)

Value	Meaning	Description
0	Non-specific	There is no specific reason why the value is BAD.
1	Configuration error	The value is not useful because of some inconsistency regarding the configuration.
2	Not connected	The value is not reliable because the connection to the provider has been disconnected at consumer-side. For example, a communication driver actively disconnects from a PLC on user request or by design.
4	Sensor failure	The value is not useful because it cannot be converted. A value from the device (PLC) cannot be converted to the corresponding HMI tag.
5	No communication, with last usable value	The value is not useful because the communication to the data source failed, however a last known value is available.
6	No communication, no usable value	The value is not useful because the communication to the data source failed or has never been established since it was last out of service and a last known value is not available.
7	Out of service	The value is not reliable because the provider side has been disabled or shutdown. For example, a PLC is in stop mode or a tag is disabled for maintenance purposes.

Sub-status "UNCERTAIN" (sub-status bits 2..5)

Value	Meaning	Description
0	Non-specific	There is no specific reason why the value is UNCERTAIN.
1	Last usable value	The connection to the data source is still established, however, the data source stopped updating the value for some reason.
2	Substitute value	A predefined value is used in case of an invalid value due to communication issues with the data source or a range violation. The reason for providing substitute values is configurable.
3	Initial value	A predefined value intended for the startup of the HMI system (or a subordinate device) is used while not being able to provide values from the data source.
5	Range violation	The value lies outside the range defined by minimum value and maximum value. The limits define which direction (min or max) has been exceeded.
6	Sub-normal	A value derived from multiple values has less than the required number of good sources. This includes data aggregation by means of data compression algorithms.

Sub-status "GOOD (cascade)" (sub-status bits 2..5)

Value	Meaning	Description
0	Non-specific	No error or special condition is associated with this value.
6	Local override	The value has been overridden by the user or some logic in to continue operation. Typically, the input has been disconnected and a manually entered value has been 'forced', or a value has been corrected.

Quality (bits 6,7)

Value	Meaning	Description
0	BAD	The value is not useful for reasons indicated by the sub-status.
1	UNCERTAIN	The quality of the value is less than normal, but the value may still be useful. The reason is indicated by the sub-status.
2	GOOD (non-cascade)	The quality of the value is good.
3	GOOD (cascade)	The quality of the value is good and may be used in control.

Extended sub-status "BAD" (sub-status (bits 8..11))

Value	Meaning	Description
0	Non-specific	No CHROM specific extended bad sub-status is associated with this value.
1	Aggregated value	The value has been calculated out of multiple values with less than the required number of good sources. This includes data aggregation by means of data compression algorithms. The corresponding sub-status is set to 'non-specific'.
3	Unusable value	A (logged) value has been identified to be incorrect, but a respective correction value is not available. The corresponding sub-status is set to 'non-specific'.
7	Disabled	The provider of the value (logging tag for logged value) has been disabled and the previous value was BAD. The corresponding sub-status is taken from the last (previous) sub-status.

Extended sub-status "UNCERTAIN" (sub-status (bits 8..11))

Value	Meaning	Description
0	Non-specific	No CHROM specific extended uncertain sub-status is associated with this value.
1	Aggregated value	The value has been calculated out of multiple values with less than the required number of good sources to be GOOD as well as less than required number of bad sources to be BAD. This includes data aggregation by means of data compression algorithms. The corresponding sub-status is set to 'non-specific'.
7	Disabled	The provider of the value (logging tag for logged value) has been disabled and the previous value was GOOD or UNCERTAIN. In case of GOOD, the corresponding sub-status is set to 'last usable value'. In case of UNCERTAIN, the corresponding sub-status is taken from the last (previous) sub-status.

Extended sub-status "GOOD (cascade)" (sub-status (bits 8..11))

Value	Meaning	Description
0	Non-specific	No CHROM specific extended good sub-status is associated with this value.
1	Aggregated value	The value has been calculated out of multiple (good) values. This includes data aggregation by means of data compression algorithms. The corresponding sub-status is set to 'non-specific'.
2	Manual input	A (logged) value has been created manually. The corresponding sub-status is set to 'non-specific'.
3	Corrected value	A (logged) value has been corrected. The corresponding sub-status is set to 'non-specific'.
4	Last usable value	The local data source has been initialized with the last usable value if present inside a local persistency. The corresponding sub-status is set to 'non-specific'.
6	Initial value	The local data source has been initialized with the configured initial value. The corresponding sub-status is set to 'non-specific'.

Flags (bit 12..15)

Value	Meaning	Description
Bit 12	Source quality	The data quality has been determined and assigned by external data source.
Bit 13	Source time	The data timestamp has been produced and assigned by external data source.
Bit 14	Time corrected	The data timestamp applied by external data source has been corrected by the system. Thus, Bit 13 "Source time" is not set. Time correction happens if the external timestamp is older than the timestamp of the last known value.

`ts`: timestamp of the data point. it is in ISO 8601 Zulu format.

`val`: value of the Tag. Based on the data type of the data point, the value can be simple scalar value.

Alarms Data

On the other hand, the 'Bulk Publish Functionality' does not apply for alarms data. The topic format for alarms data is as follows:

```
ie/d/j/simatic/v1/s7c1/ev/<connection-name>/FullText
```

The JSON structure is as follows:

```

8/5/2020, 7:25:07 PM node: 710d31ec.1aa2b
ie/dj/simatic/v1/s7c1/ev/S7300/FullText : msg.payload : Object
  ▼ object
    ▼ evs: array[1]
      ▼ 0: object
        area: "System/HMI/DriverCommunication"
        clsName: "SystemAlarm"
        ▼ evTxt: array[1]
          0: "Host1 (iowa): Partner is not fully operational"
        ▼ evTxtExt: array[1]
          0: ""
          id: "1"
          modificationTime: "2020-08-05T13:55:06.000Z"
          name: "PlcInStopAlarm"
          origin: "iowa:S71_108"
          ▶ params: array[1]
          raisedTime: "2020-08-05T13:55:06.000Z"
          state: 1
      seq: 1

```

Where, `evs`: array of events published in the payload.

`area`: string containing the configured area information of the event.

`clsName`: name of the class of the event.

`evTxt`: event text. This may contain array of objects based on the event source.

`evTxtExt`: array of string containing additional texts of the events.

`id`: unique identification of the event.

`modificationTime`: event's last modified timestamp in UTC format.

`name`: name of the event.

`origin`: string containing the origin of the event.

`params`: array of parameters of the event.

`raisedTime`: event's raised timestamp in UTC format.

`state`: integer that indicates the state of the event (provider specific).

Value	Description
0	Normal
1	Alarm is raised.
2	Raised and cleared.

`seq`: unique sequence number of the payload.

7.2.2 JSON structure for Alarms and Tags Metadata

When you deploy a project as described in Deploy Project (Page 41), the metadata is also published.

Note

You must discover the tag information from the metadata payload.

Tags Metadata

The topic format for tags metadata is as follows:

`ie/m/j/simatic/v1/s7c1/dp`

where `ie`: industrial edge, `m`: metadata, `j`: json, `s7c1`: SIMATIC S7 Connector instance 1, `dp`: datapoint.

With the "Bulk Publish" feature, the JSON structure for tags metadata is as follows:

```

ie/m/j/simatic/v1/s7c1/dp: msg.payload: Object
  ▼ object
    seq: 1
    ▼ connections: array[1]
      ▼ 0: object
        name: "opc"
        type: "OPCUA"
        ▼ dataPoints: array[1]
          ▼ 0: object
            name: "default"
            topic: "ie/d/j/simatic/v1/s7c1/dp/r/opc/default"
            publishType: "bulk"
          ▼ dataPointDefinitions: array[4]
            ▼ 0: object
              name: "FirstScan"
              id: "110"
              dataType: "Bool"
            ▼ 1: object
              name: "Data_block_2_String"
              id: "111"
              dataType: "String"
            ▼ 2: object
              name: "Data_block_2_Real"
              id: "112"
              dataType: "Real"
            ▶ 3: object

```

Alarms Metadata

The topic format for alarms metadata is as follows:

```
ie/m/j/simatic/v1/s7c1/ev
```

where *ie*: industrial edge, *m*: metadata, *j*: json, *s7c1*: SIMATIC S7 Connector instance 1, *ev*: event.

The JSON structure for alarms metadata is as follows:

```

7/1/2020, 1:30:21 PM node: 560cf8a7.531af8
ie/m/j/simatic/v1/s7c1/ev : msg.payload : Object
  ▼ object
    seq: 1
  ▼ connections: array[1]
    ▼ 0: object
      name: "S7-300"
      type: "S7"
    ▼ events: array[1]
      ▼ 0: object
        type: "FullText"
        topic: "ie/d/j/simatic/v1/s7c1/ev/S7-300/FullText"

```

Note

Alarms metadata is published only when you deploy S7 Connection with 'Full Text Alarm' checkbox enabled before starting the project.

Once you start the Industrial Edge Runtime, you cannot configure 'Full Text Alarms' checkbox. Therefore, the alarms metadata is not published in this case.

7.2.3 Connection and Disconnection Status

When you successfully deploy a project, the metadata packet and metadata client status are published as follows:

```

12/11/2020, 1:35:55 PM node: 4c5cf855.ffbf38
ie/m/j/simatic/v1/s7c1/status : msg.payload : Object
  ▼ object
    id: 0
    msg: "Connected"

```

```

12/11/2020, 1:35:56 PM node: 944a308a.706dc
ie/m/j/simatic/v1/s7c1/dp : msg.payload : Object
  ▶ { seq: 1, connections: array[1] }

```

When you successfully start a project, the tag and alarm client statuses are published with tag and alarm data packet as follows:


```
12/11/2020, 2:47:01 PM node: 41c0897c.2ed9e8
ie/dlj/simatic/v1/s7c1/dp/r/status : msg.payload : Object
▼ object
  id: 0
  msg: "Connected"
```

```
12/11/2020, 2:47:03 PM node: 726bdd82.258394
ie/dlj/simatic/v1/s7c1/ev/status : msg.payload : Object
▼ object
  id: 0
  msg: "Connected"
```

When you successfully stop a project, the disconnected state from clients is published to broker as follows:

```
12/11/2020, 2:50:05 PM node: 4c5cf855.ffbf38
ie/mlj/simatic/v1/s7c1/status : msg.payload : Object
▼ object
  id: 1
  msg: "Disconnected"
```

```
12/11/2020, 2:50:05 PM node: 726bdd82.258394
ie/dlj/simatic/v1/s7c1/ev/status : msg.payload : Object
▼ object
  id: 1
  msg: "Disconnected"
```

```
12/11/2020, 2:50:07 PM node: 41c0897c.2ed9e8
ie/dlj/simatic/v1/s7c1/dp/r/status : msg.payload : Object
▼ object
  id: 1
  msg: "Disconnected"
```

Note

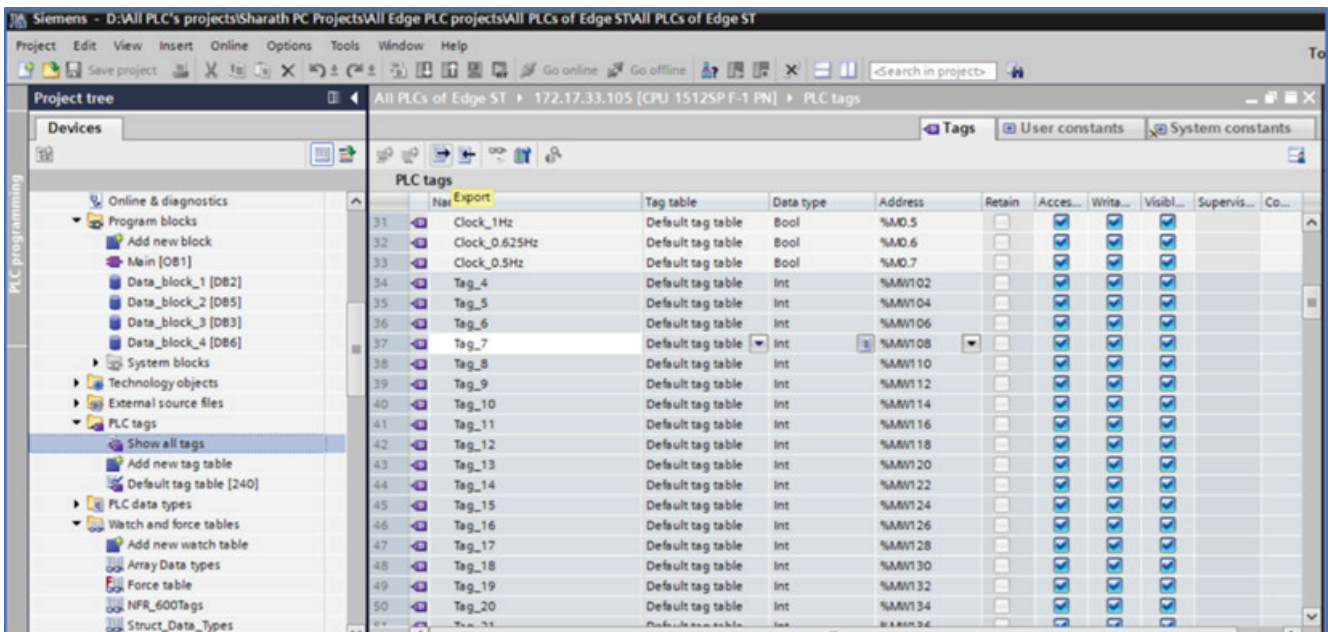
- "ie/m/j/simatic/v1/s7c1/status" topic must be configured in IE Databus to get metadata client status.
- Metadata client status data is not retained.
- Tag data client status is published in "ie/m/j/simatic/v1/s7c1/dp/r/status" topic.
- Alarm data client status is published in "ie/m/j/simatic/v1/s7c1/ev/sstatus" topic.
- Tag data and alarm data client status data is retained.
- 'id: 0' indicates metadata, tag, and alarm MQTT Publisher client are successfully connected to the broker, and 'id: 1' indicates these are not connected.

7.3 How to Export Tags from TIA Portal?

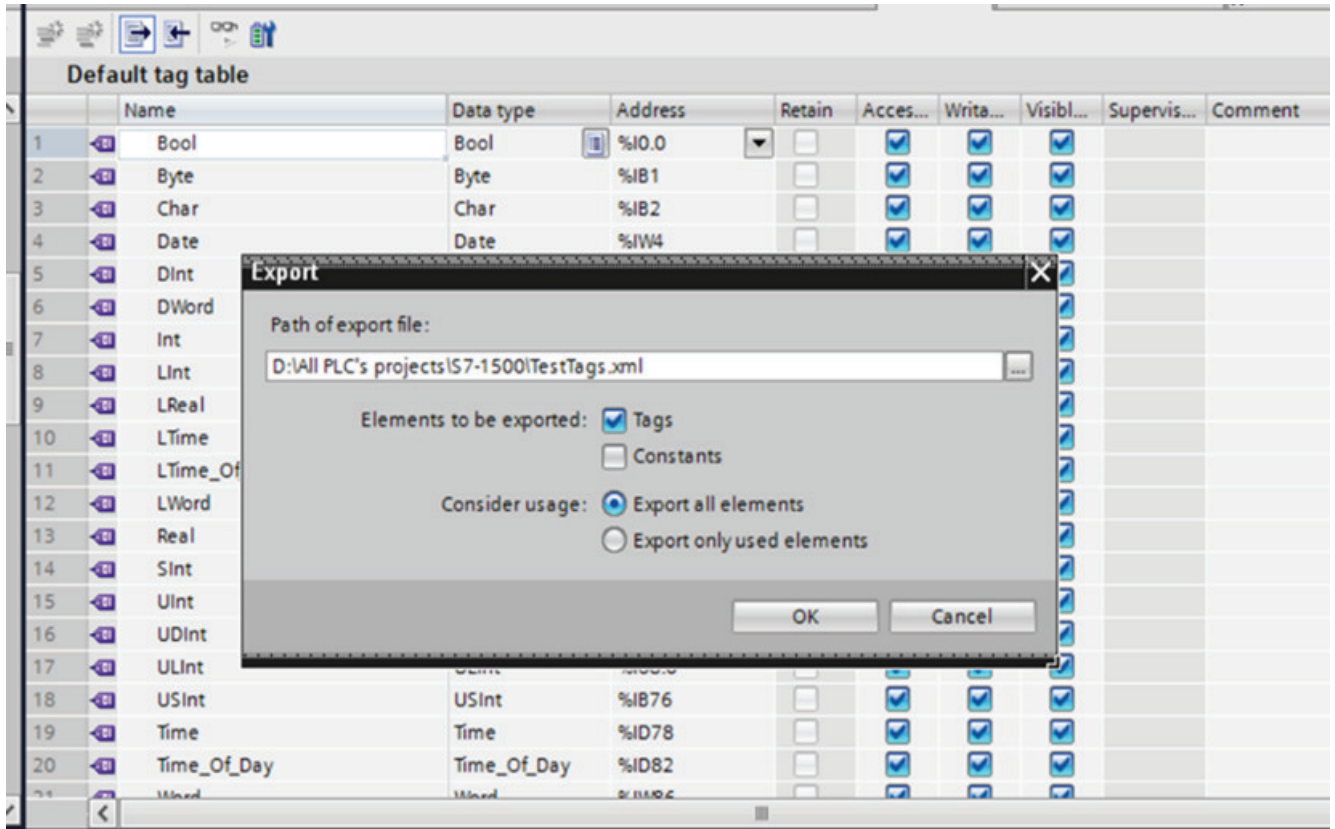
7.3.1 Export Tags for S7-Protocol (S7-300/400/1200/1500) and Optimized S7-Protocol (S7-1200/1500) Connection

PLC Tag Tables can be exported from TIA Portal for S7-Protocol (S7-300/400/1200/1500) and Optimized S7-Protocol (S7-1200/1500) connection. To export the XML file, follow these steps:

1. Go to TIA Portal where PLC is configured.
2. Go to PLC Tags folder.
3. Click on Tag Table.



- Click on "Export" icon.
The "Export" dialog box is displayed. Specify the file name with .xml extension as follows:



- Click "OK".
The XML file is exported.

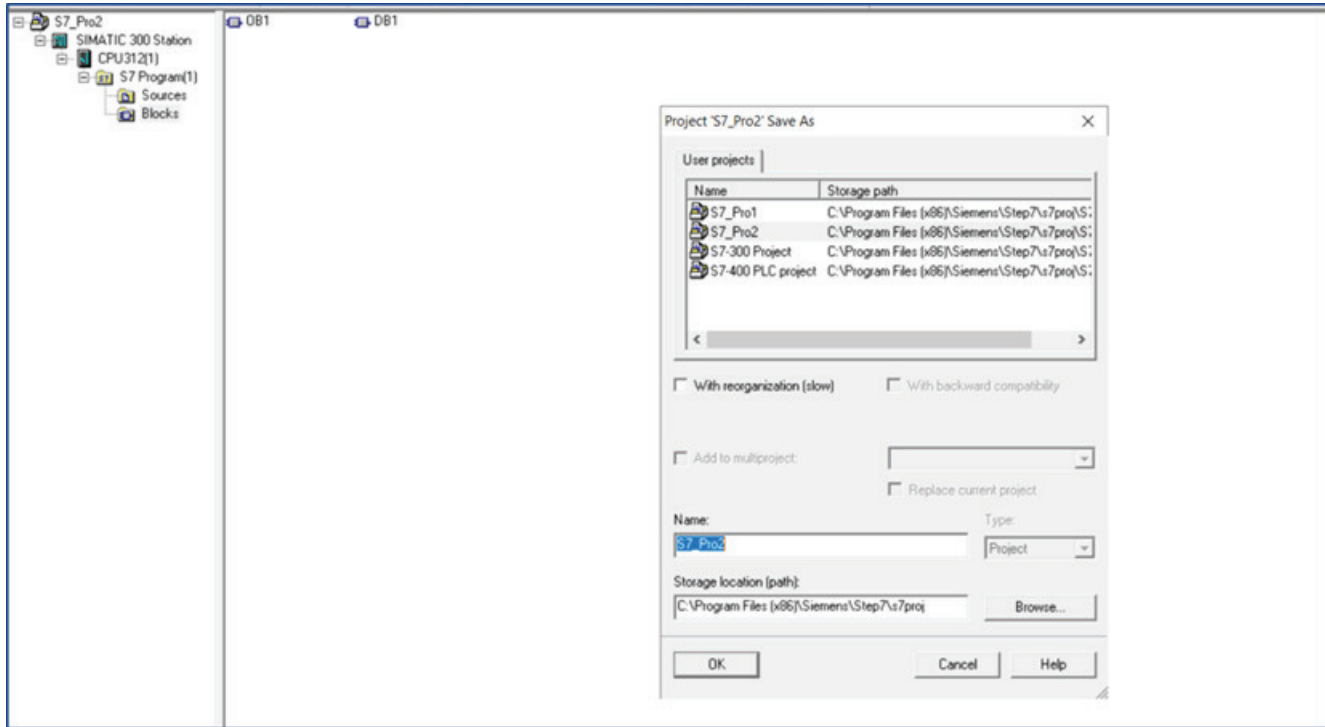
From TIA Portal, only XML file format is supported in the SIMATIC S7 Connector Configurator.

Tags in Data Blocks cannot be exported from the TIA Portal. This can be done in SIMATIC Manager. You can configure a 'Project' for S7 Classic PLC in STEP 7 SIMATIC Manager and can use this 'Project' to create an excel sheet or XML file which can be imported in S7-Protocol

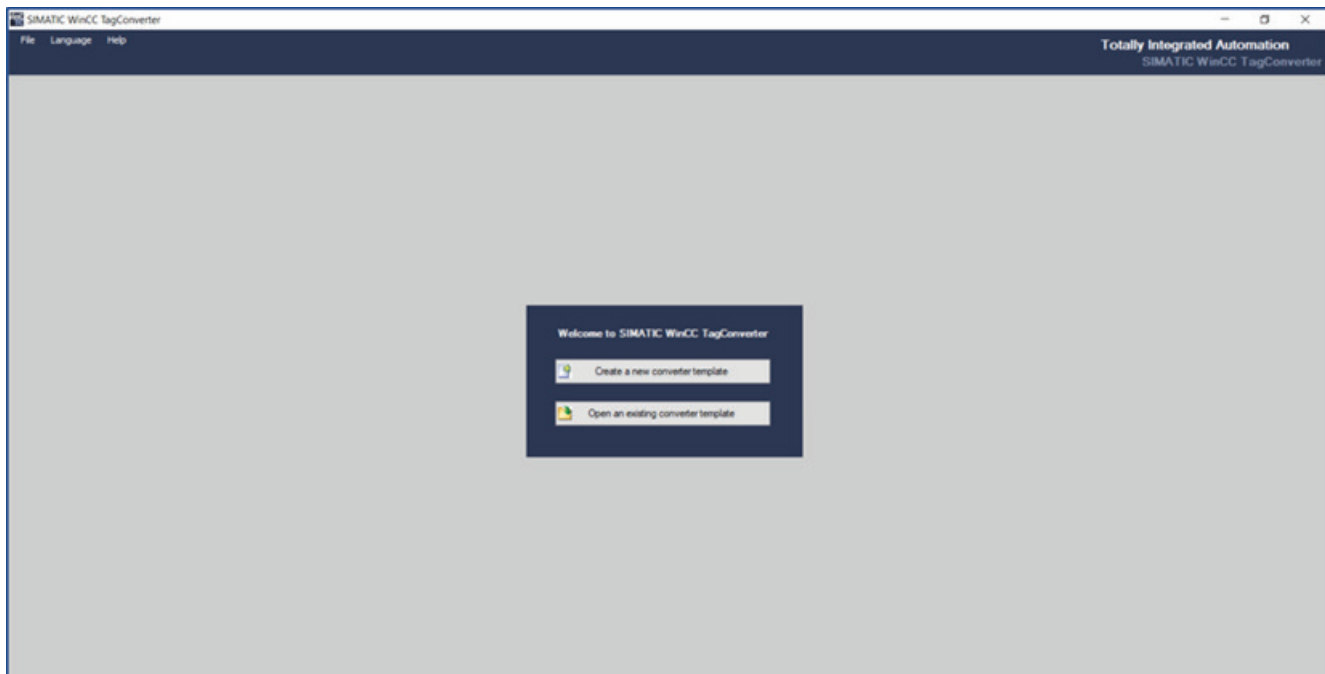
7.3 How to Export Tags from TIA Portal?

(S7-300/400/1200/1500) connection in SIMATIC S7 Connector Configurator. The procedure steps are as follows:

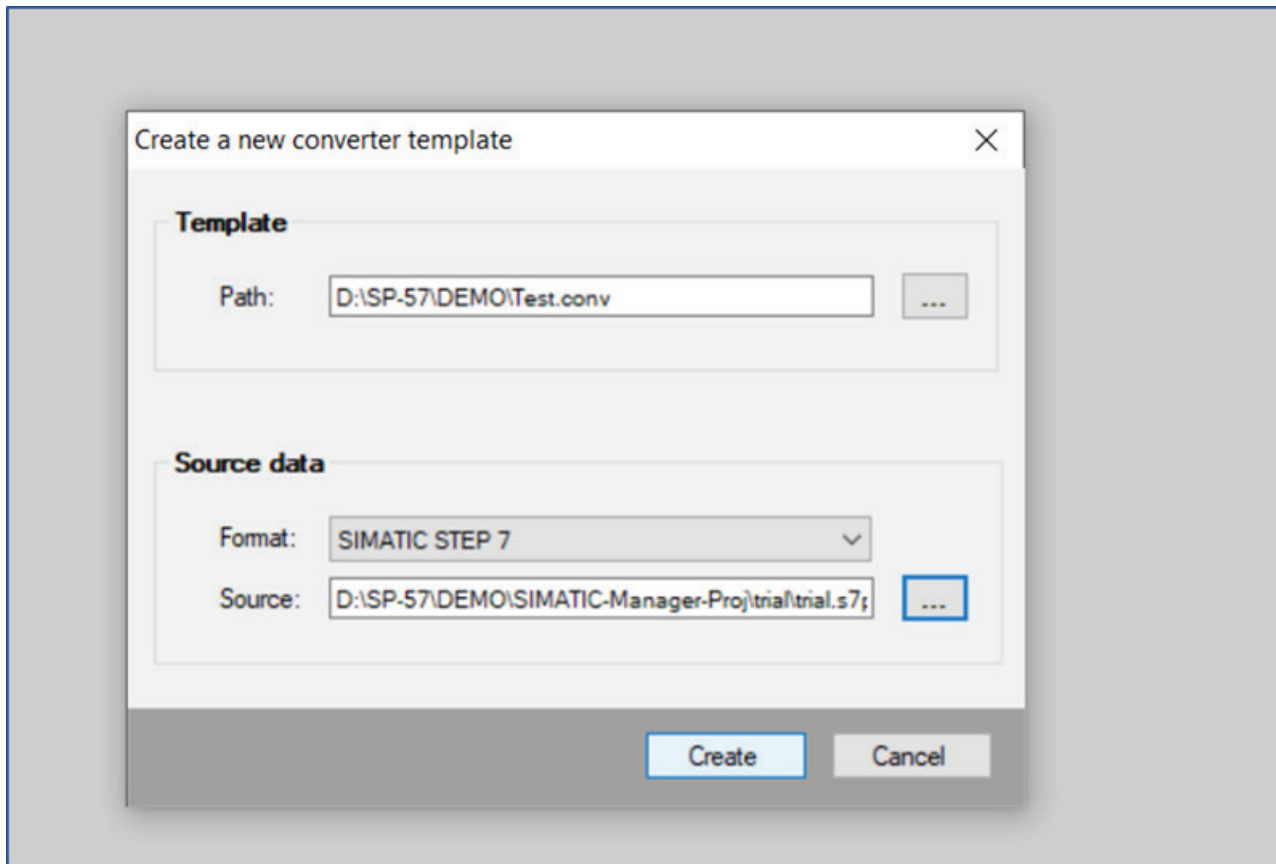
1. Create a project in SIMATIC Manager and save the project.



2. Open SIMATIC WinCC Tag Converter Tool.

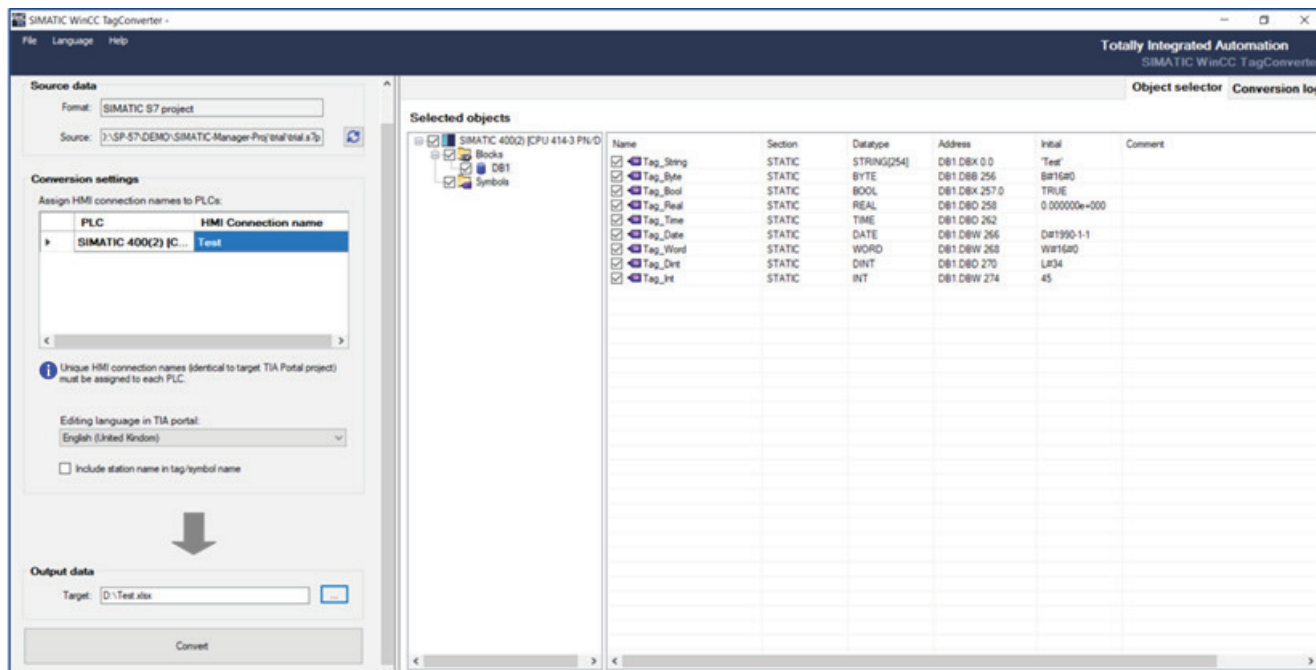


3. Click on "Create a new converter template" to create the template file. Select the Path to save template file and choose s7p (*.s7p) project created from SIMATIC Manager.



4. Click "Create".
5. Select the Tags from data Block and Symbols that you want to be added in the excel sheet or XML file.

6. Provide Connection Name and choose the output format as .xlsx or .xml as required.

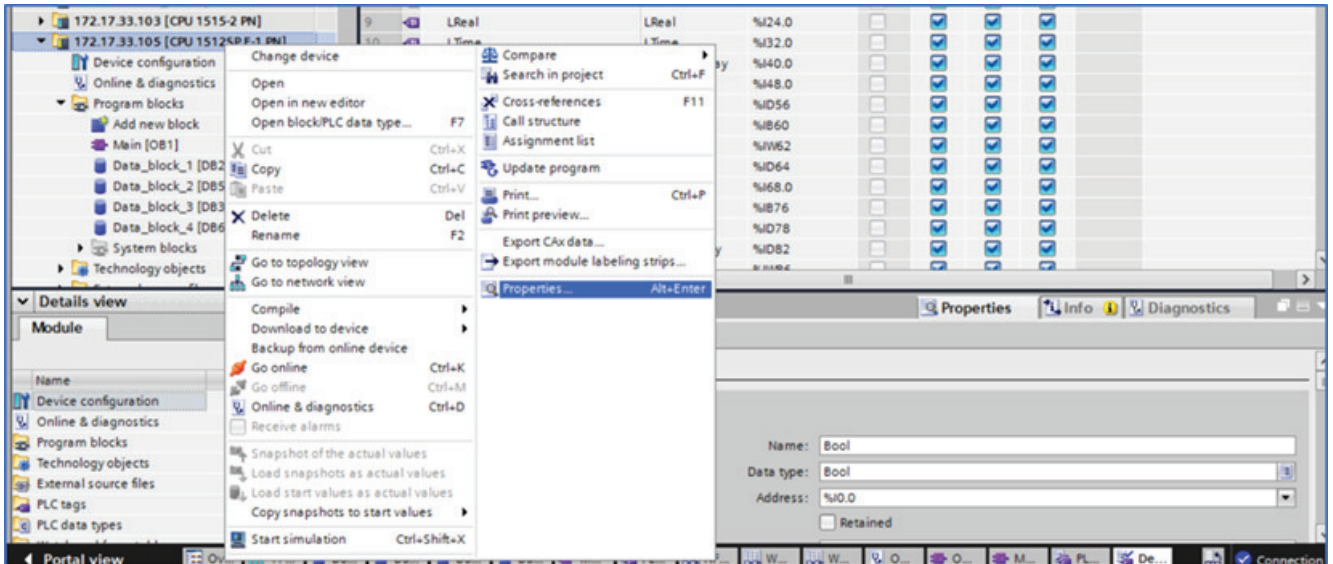


7. Click "Convert".
The file is exported.

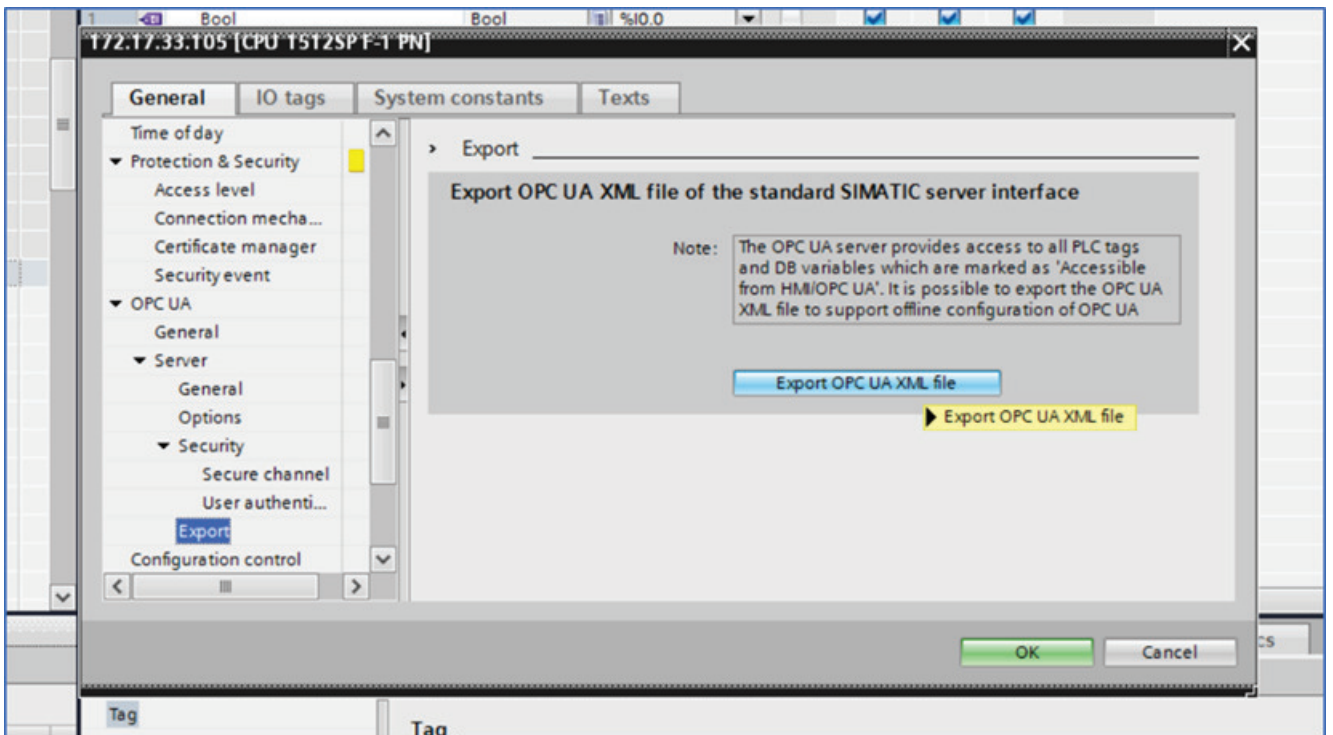
7.3.2 Export Tags for OPC-UA Connection

You can export both PLC Tags and Datablock tags from TIA Portal for OPC-UA. To export the XML file, follow these steps:

1. Go to TIA portal where PLC is configured.
2. Right-click on PLC and click "Properties" as follows:



3. Click "OPC UA" > "Server" > "Export" section in "Properties" window as follows:



7.4 How to Configure String Datatype in S7-Protocol (S7-300/400/1200/1500)?

4. Click "Export OPC UA XML file" button.
5. Provide the file name to store the tags in XML file and click "Save".
The file is exported.

The exported file may look as follows:

```
<?xml version="1.0" encoding="utf-8"?>
<UANodeSet LastModified="2021-04-09T14:25:56Z" xmlns="
http://opcfoundation.org/UA/2011/03/UANodeSet.xsd" xmlns:uax="
http://opcfoundation.org/UA/2008/02/Types.xsd" xmlns:si="
http://www.siemens.com/OPCUA/2017/SimaticNodeSetExtensions">
<UAVariable NodeId="ns=3;s="Tag_8"" BrowseName="3:Tag_8" ParentNodeId=
"ns=3;s=Outputs" DataType="BOOL" AccessLevel="3">
  <DisplayName>Tag_8</DisplayName>
  <References>
    <Reference ReferenceType="HasTypeDefinition">i=63</Reference>
    <Reference ReferenceType="Organizes" IsForward="false">ns=3;s=Outputs
  </Reference>
  </References>
</UAVariable>
<UAVariable NodeId="ns=3;s="Tag_9"" BrowseName="3:Tag_9" ParentNodeId=
"ns=3;s=Outputs" DataType="BOOL" AccessLevel="3">
  <DisplayName>Tag_9</DisplayName>
  <References>
    <Reference ReferenceType="HasTypeDefinition">i=63</Reference>
    <Reference ReferenceType="Organizes" IsForward="false">ns=3;s=Outputs
  </Reference>
  </References>
</UAVariable>
</UANodeSet>
```

7.4 How to Configure String Datatype in S7-Protocol (S7-300/400/1200/1500)?

To configure string datatype in S7-Protocol (S7-300/400/1200/1500), you must specify the address based on offset formed in TIA Portal. The format is as follows:

<DataBlock>. <DataBlock Offset>

For example, Tag 11 < string Type> in Data Block 1 (DB1)

Tag8	Int	16.0	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Tag9	Int	18.0	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Tag10	Int	20.0	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Tag11	String	22.0	'shrikant'	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
StrType	String	278.0	'SIEMENS'	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Off set of Tag 11 (String tag) is 22 as depicted in the above image. The address format for Tag11 (String tag) is 'DB1.DBB22'.

You can configure the string tag data type in S7-Protocol (S7-300/400/1200/1500) as follows:

Add Data Source		Delete		Deploy				Stop Project	
<input type="checkbox"/>	Name	Comments	Address	Data Type	Acquisition Cycle	Acquisition Mode	Access Mode	Actions	
<input type="checkbox"/>	128							✎ + - 🗑	
<input type="checkbox"/>	S7							✎ + - 🗑	
<input type="checkbox"/>	opc							✎ + - 🗑	
<input type="checkbox"/>	S7300							✎ + - 🗑	
<input type="checkbox"/>	StringTag		%DB1.DBB22	String	1 second	CyclicContinuous	Read	✎	

You can monitor string value in TIA Portal to cross check Data Block offsets.

	i	Name	Address	Display format	Monitor value	Modify value	🔧
1		"Data_block_1".Tag11[1]	%DB1.DBB24	Character	's'		<input type="checkbox"/>
2		"Data_block_1".Tag11[2]	%DB1.DBB25	Character	'h'		<input type="checkbox"/>
3		"Data_block_1".Tag11[3]	%DB1.DBB26	Character	'r'		<input type="checkbox"/>
4		"Data_block_1".Tag11[4]	%DB1.DBB27	Character	'i'		<input type="checkbox"/>
5		"Data_block_1".Tag11[5]	%DB1.DBB28	Character	'k'		<input type="checkbox"/>
6		"Data_block_1".Tag11[6]	%DB1.DBB29	Character	'a'		<input type="checkbox"/>
7		"Data_block_1".Tag11[7]	%DB1.DBB30	Character	'n'		<input type="checkbox"/>
8		"Data_block_1".Tag11[8]	%DB1.DBB31	Character	't'		<input type="checkbox"/>

7.5 How to Generate OPC-UA Client and Server Certificates?

SIMATIC S7 Connector Configurator provides the following two messaging modes for secure communication in OPC-UA:

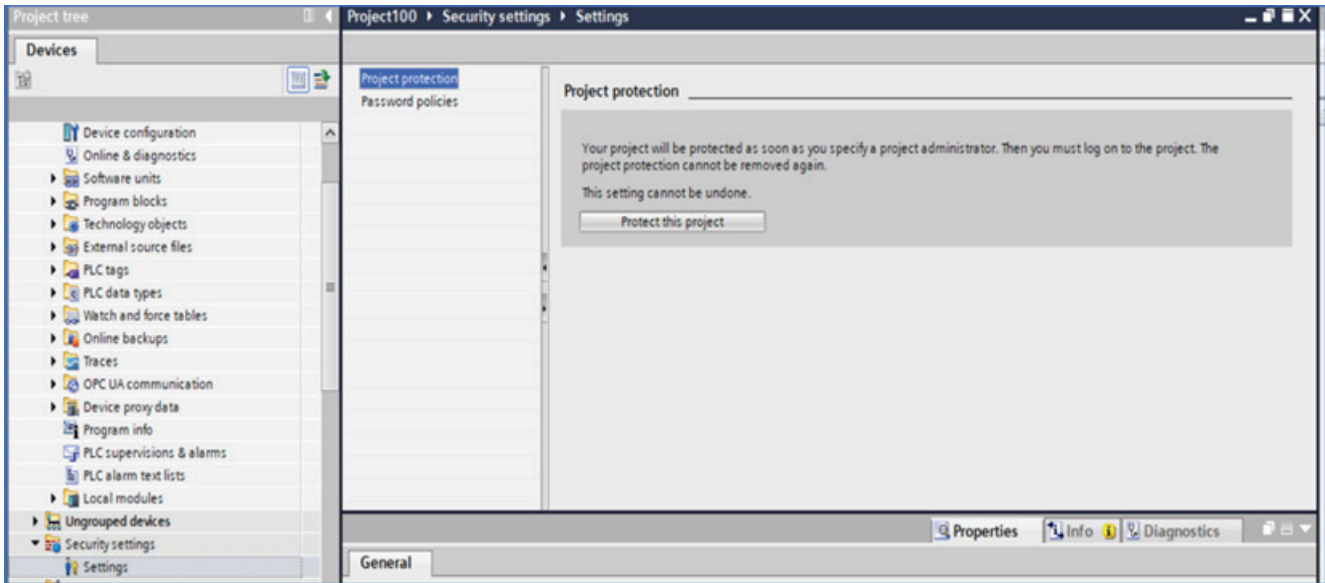
1. Sign
2. Sign & Encrypt

"Sign & Encrypt" messaging mode enables you to encrypt the data that is sent after establishing the communication.

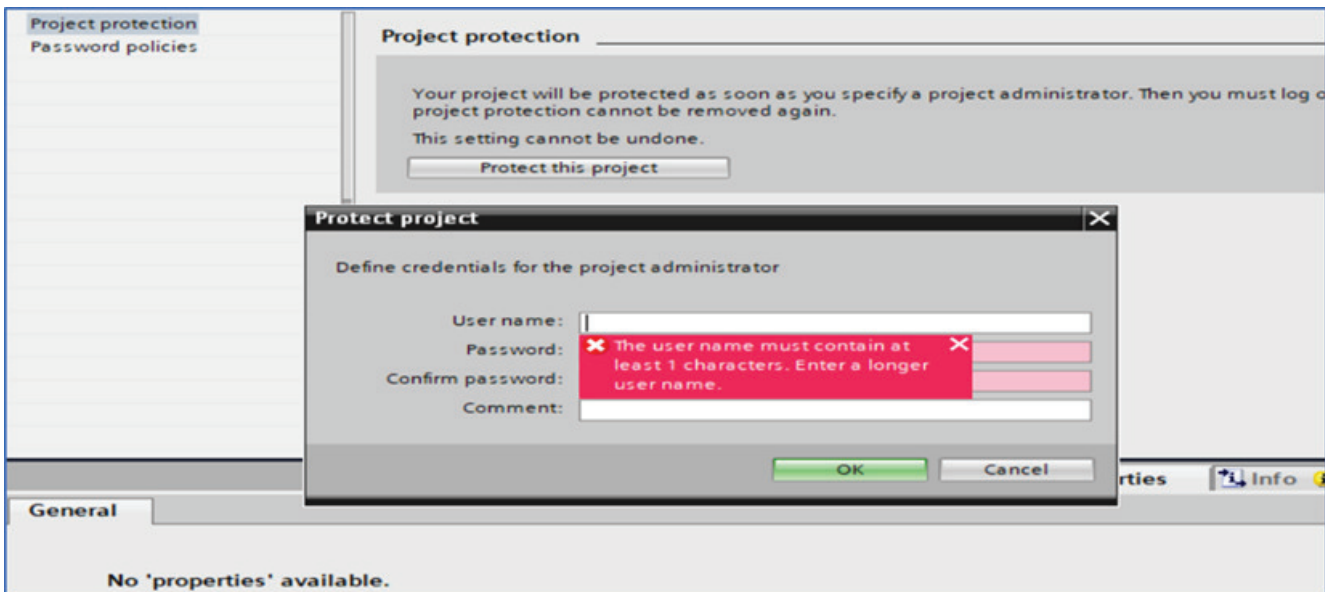
For secure communication, you must create OPC-UA client and server certificates using TIA Portal using the following steps:

Step 1: Protect the Project by selecting the "Project protection" option in TIA Portal.

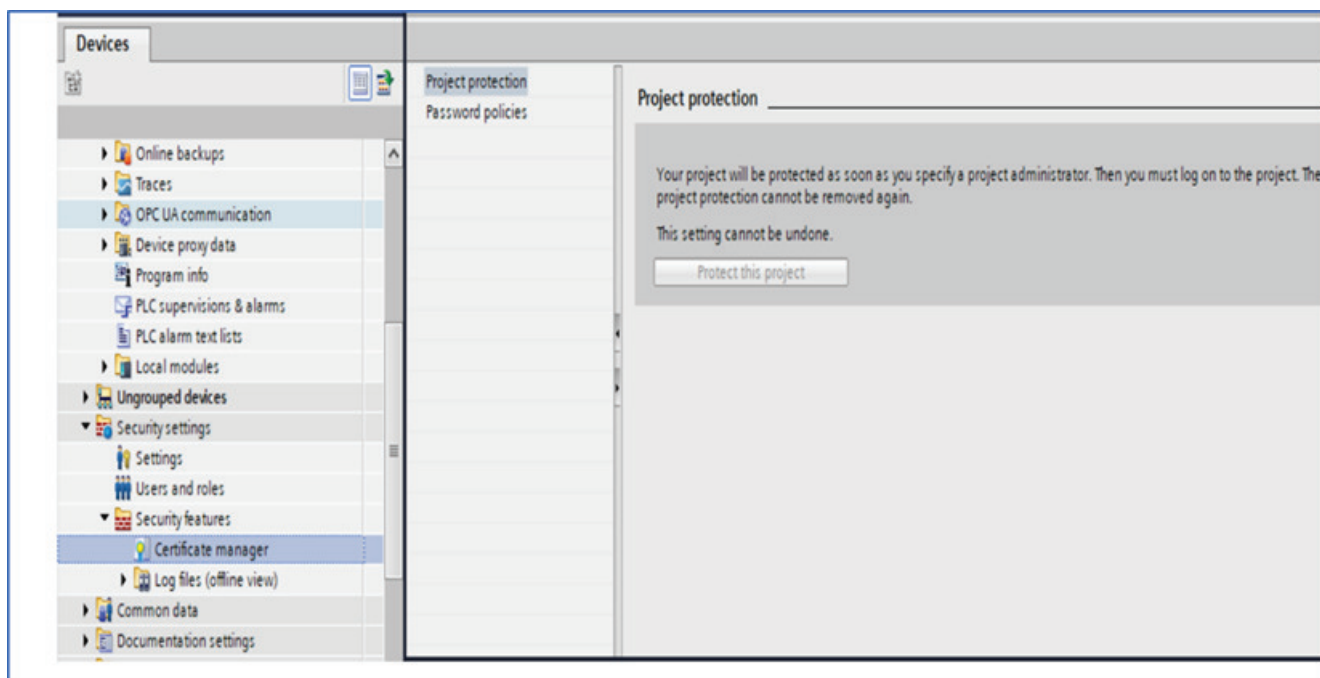
1. Click on "Security settings" > "Settings".
A new window is displayed for project as follow:



2. Click "Protect this project".
The "Protect project" dialog box is displayed as follows:

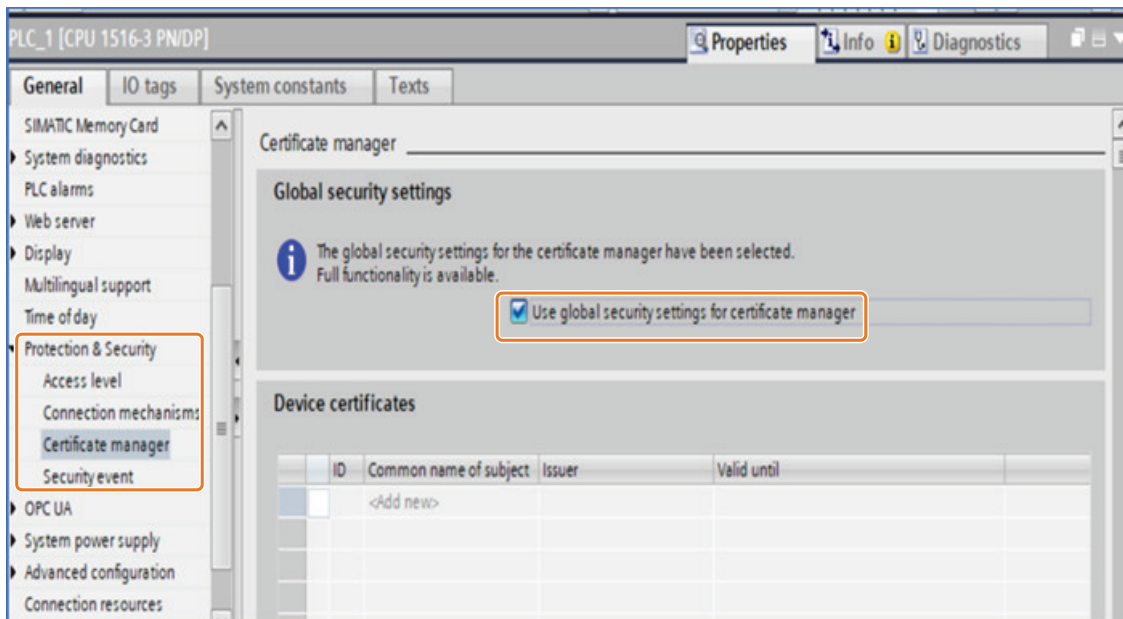


3. Enter the Project Credentials.
4. Ensure the "Certificate manager" option is enabled for the project.

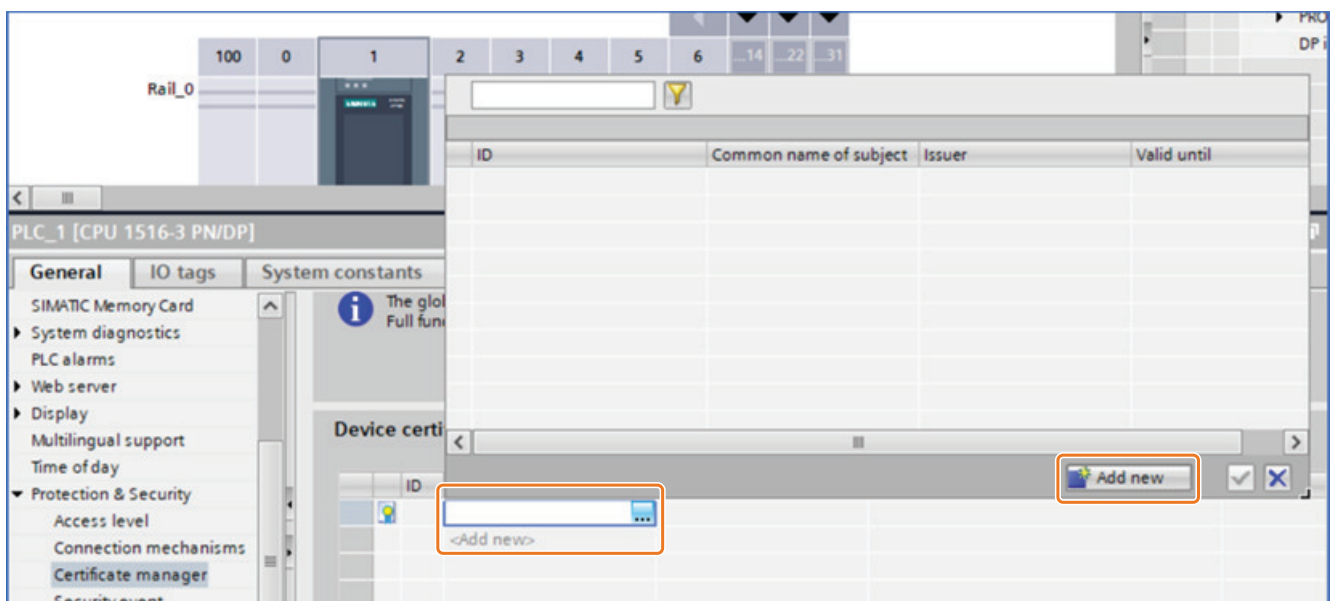


Step 2: Enable Global Security Setting and prepare server and client certificates.

1. Select the device properties, click "Protection & Security" > "Certificate manager", and mark true against "Use global security for certificate manager" checkbox.



2. In the "Device certificates", configure new security, and click "Add new" to add new certificate.



The "Create a new certificate" dialog box is displayed.

3. Complete the following fields to create 'Server Certificate'.
 - Select "Self signed" radio button.
 - Define the "Common Name of Subject" as 'ServerCert'.
 - Set the "Signature" as 'sha1RSA'.
 - Specify the "Usage" as "OPC UA server" for server certificate.
 - In "Subject Alternative Name (SAN)" field, define "IP" as IP of PLC. Delete all other fields of SAN.

Create a new certificate

CA

Choose how the new certificate is to be signed:

Self signed

Signed by certificate authority

CA name: 2: Siemens TIA Project(pJ3olPo0kku8587HG)

Certificate parameter

Enter the parameters for the new certificate:

Common name of subject: ServerCert

Signature: sha1RSA

Valid from: April 23, 2019 03:55:25 PM

Valid until: April 23, 2037 12:00:00 AM

Usage: OPC UA server

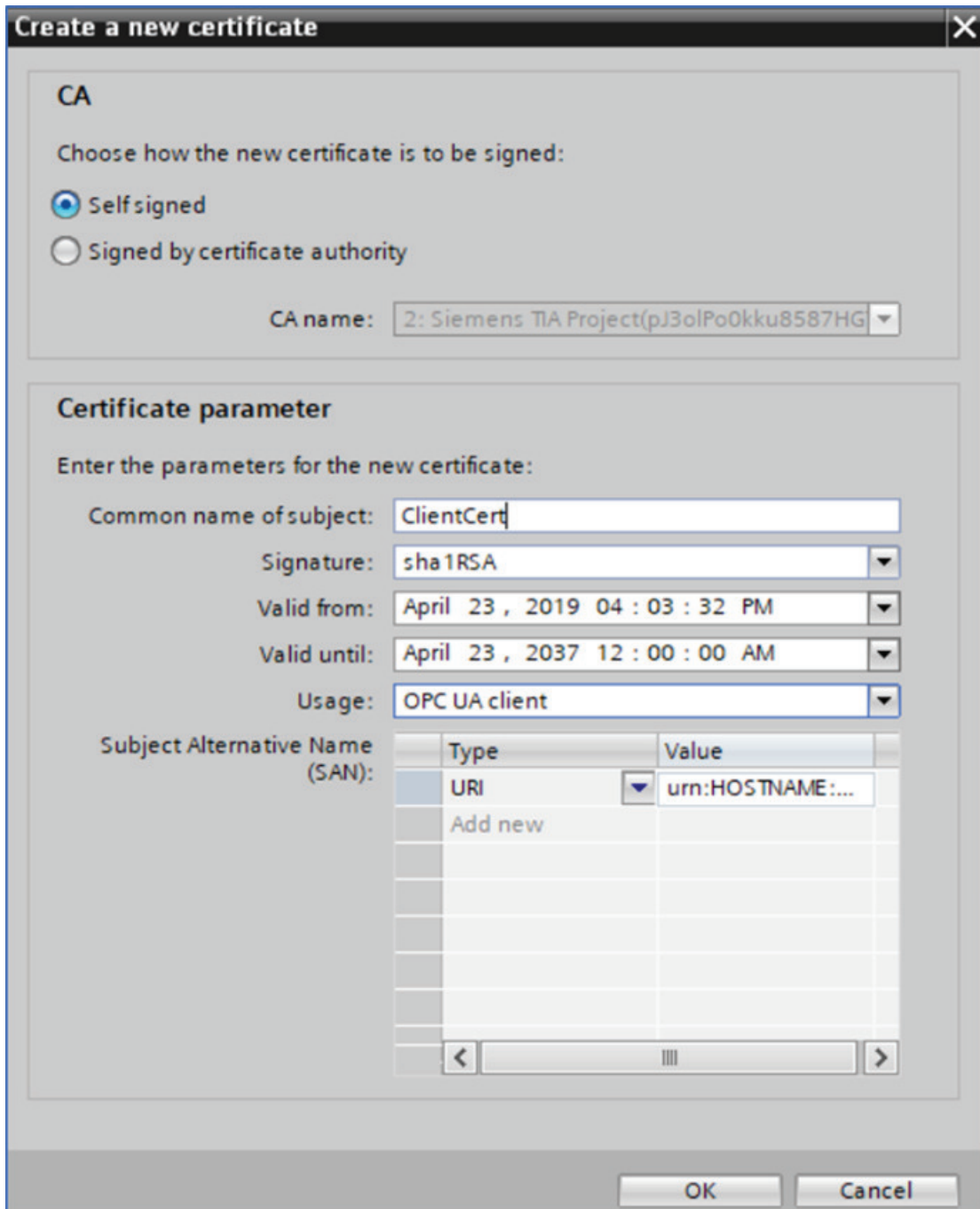
Subject Alternative Name (SAN):

Type	Value
IP	192.168.0.1
Add new	

4. Complete the following fields to create 'Client Certificate'.
 - Select "Self signed" radio button.
 - Define the "Common Name of Subject" as 'ClientCert'.
 - Set the "Signature" as 'sha1RSA'.
 - Specify the "Usage" as "OPC UA client" for client certificate.

7.5 How to Generate OPC-UA Client and Server Certificates?

- In "Subject Alternative Name (SAN)" field, define "URI" as 'urn:<hostname>:WCCILopcu'. Delete all other fields of SAN.

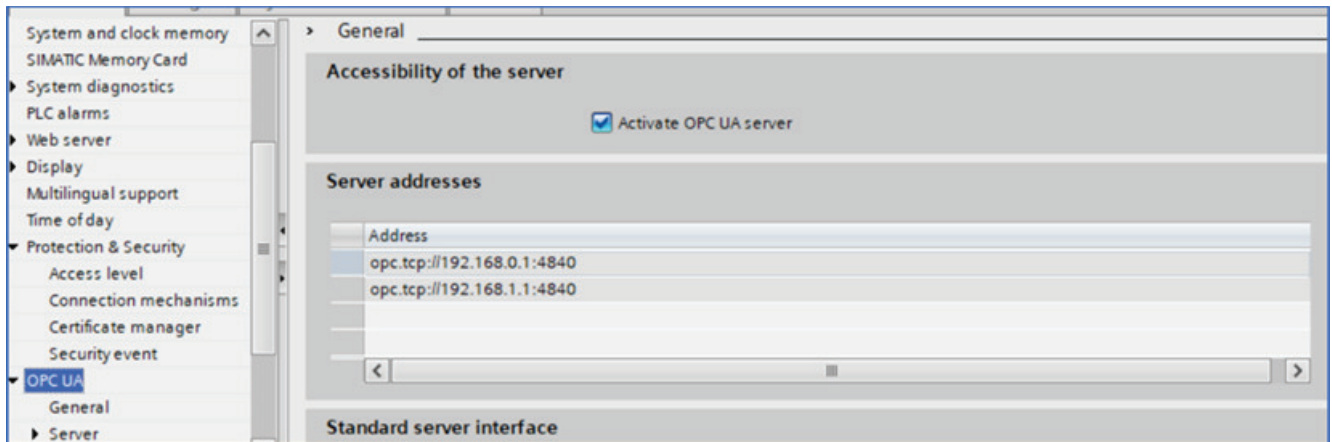


Note

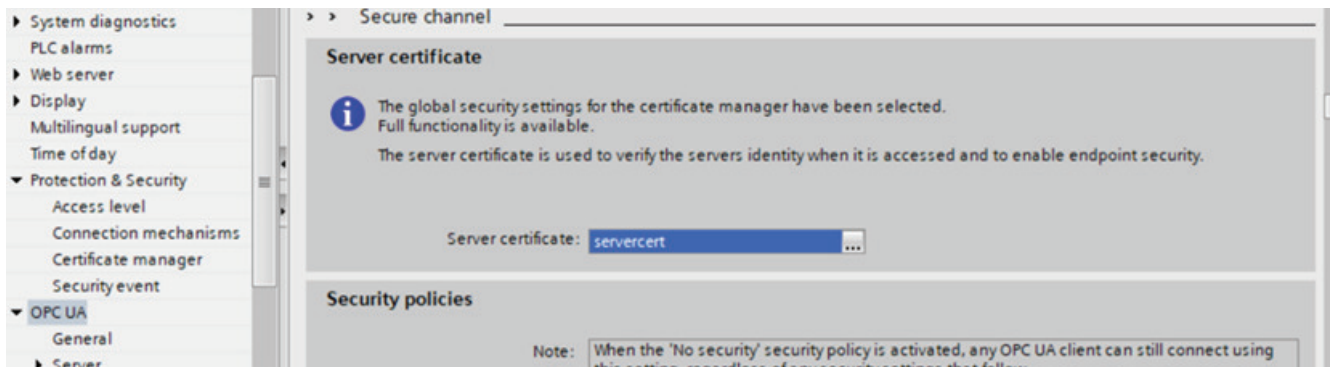
- For client certificate, the hostname of the container is required where OPC-UA application is running. Provide hostname as 's7connector' for client certificate.
- While creating the certificates, you must ensure that PLC/OPC-UA server time is not in future time. As the certificates with future time will not be activated in SIMATIC S7 Connector Configurator when configured.

Step 3: Add the Server Certificate and Client Certificate in OPC-UA Security options.

1. In OPC UA Device properties section, enable the "Activate OPC UA server" option by marking true against the checkbox.

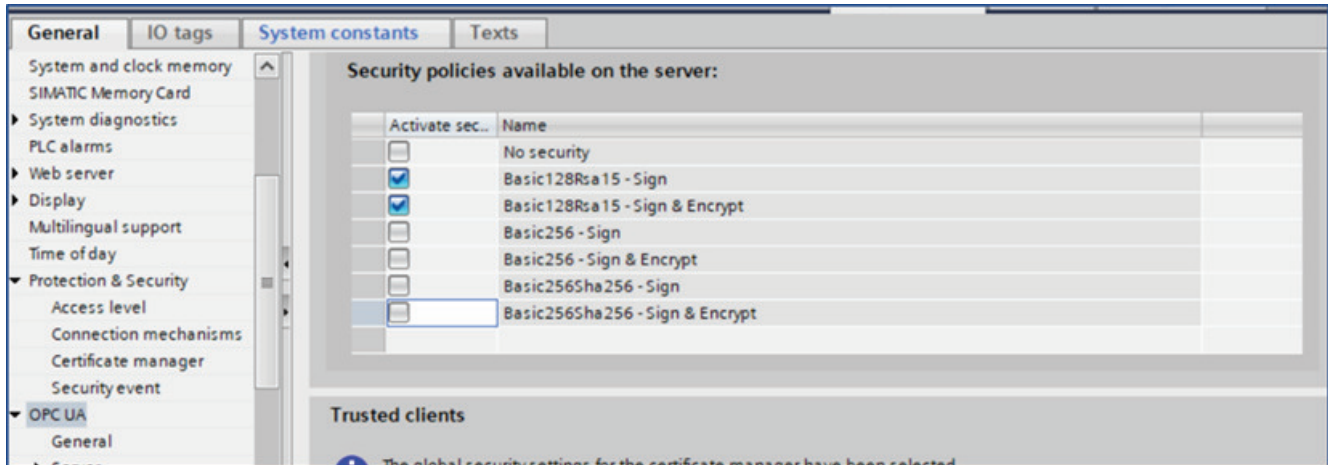


2. In "Server certificate" option, upload the created server certificate.



7.5 How to Generate OPC-UA Client and Server Certificates?

3. Select the following security policies for server functionality which corresponds 128-bit encryption algorithms.



4. Add the client certificate in Trusted Client Certificate option.

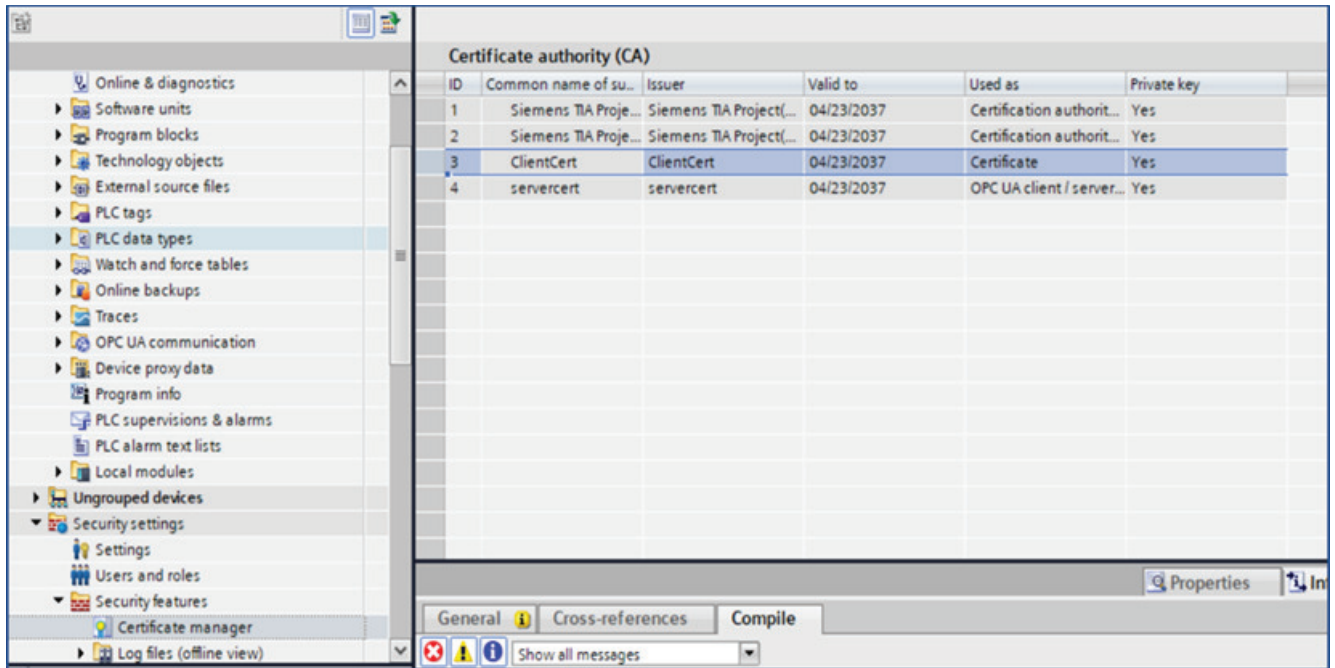


5. Save the TIA Project and download the project to PLC.

Step 4: Export the certificate to use in SIMATIC S7 Connector Configurator.

1. Click "Security settings" > "Certificate manager".

The list of certificates is displayed as follows:



2. Right-click on the "servercert" and export the certificate in .der format.
3. Right-click on the "Clientcert" and export the certificate in ".p12" format.

When you export client PKCS12 file, the password window is displayed. Define your own password. The same password must be entered in the "PKCS12 Import Password" field in SIMATIC S7 Connector Configurator as this is used to extract the client certificate data in the backend.

Note

You can follow the above steps to create certificates with 'sha256RSA' signature value as well.

7.6 How to Configure Access Password?

SIMATIC S7 Connector Configurator supports the following two access levels in which a PLC can be configured for Optimized S7-Protocol (S7-1200/1500) connection:

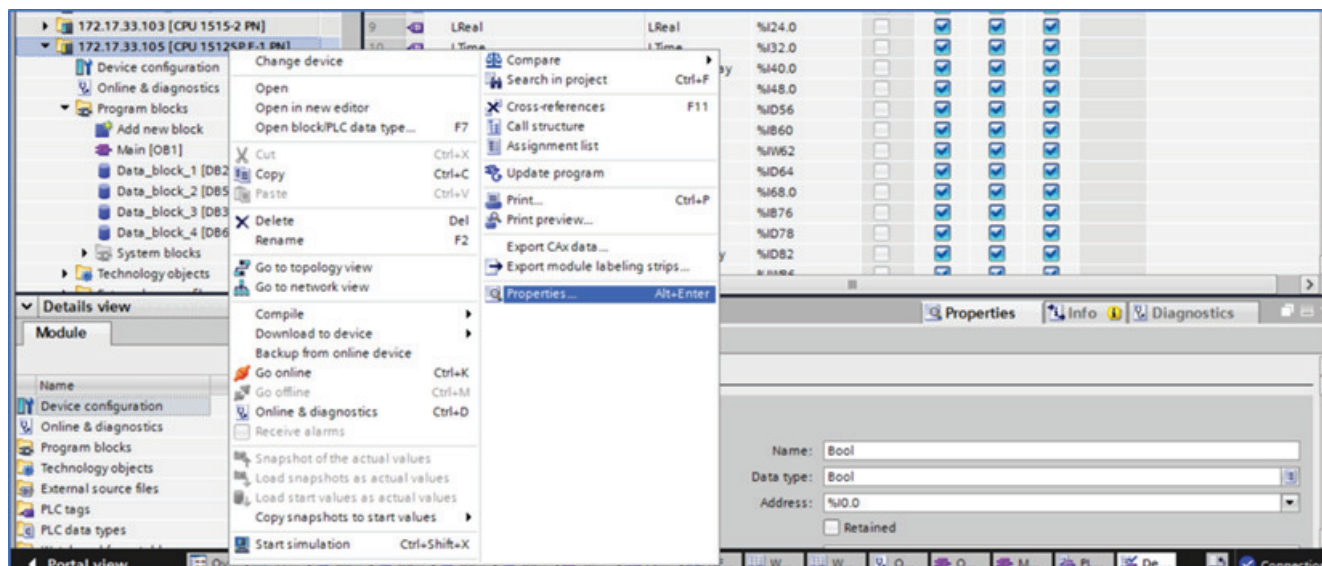
1. No Access (complete protection)
2. Full Access (no protection)

The 'Access Password' is required for 'No Access' access level.

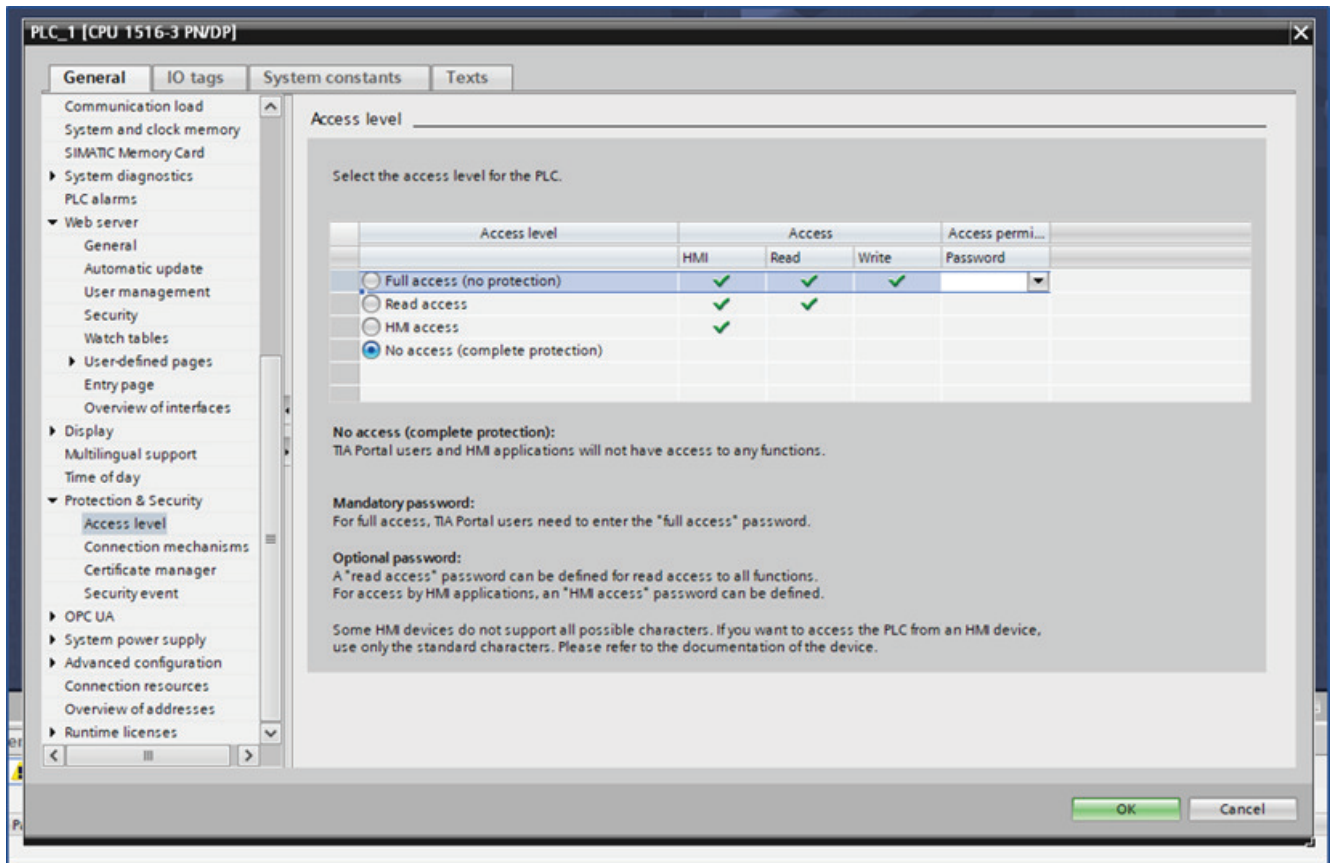
7.6 How to Configure Access Password?

To configure the access password, follow these steps:

1. Go to TIA portal where PLC is configured.
2. Right-click on PLC and click "Properties" as follows:

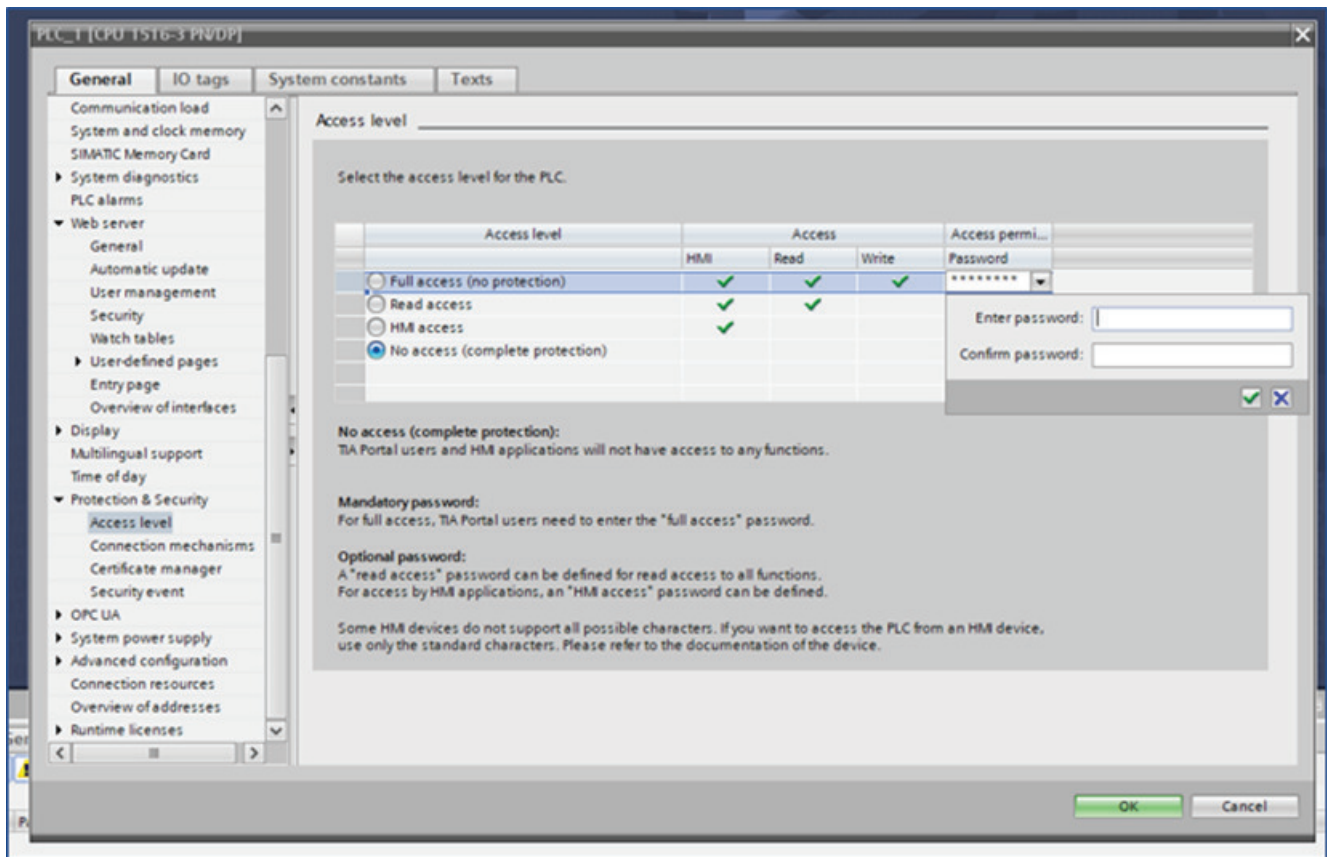


3. Click "Protection & Security" > "Access Level" section in "Properties" window as follows:



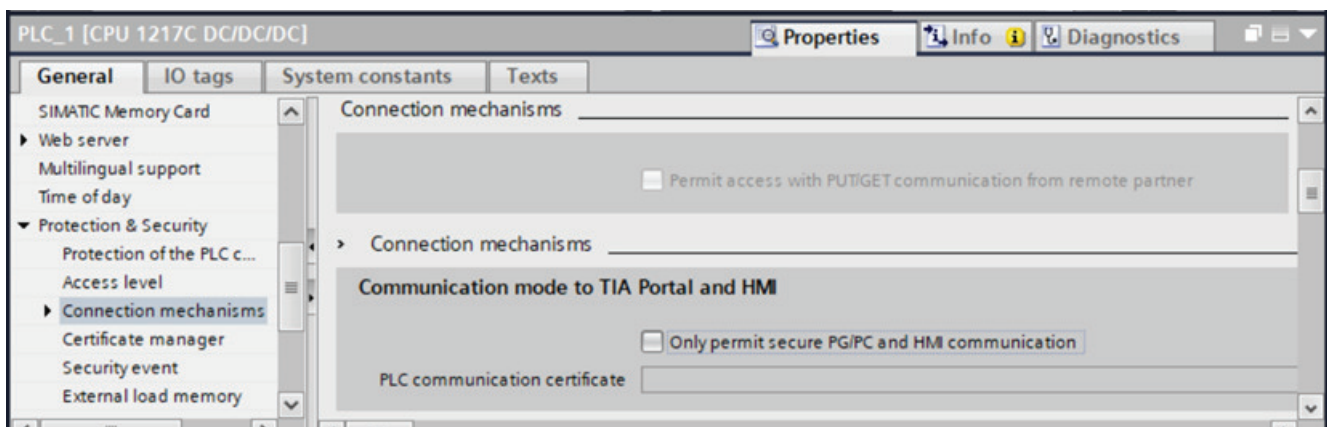
4. Select "No access" level and provide password for any of the above "Access level".

7.7 How to Configure S7-1200 PLC FW version 4.5?



7.7 How to Configure S7-1200 PLC FW version 4.5?

To support Tag read, write, and browsing function for Optimized S7-Protocol (S7-1200/1500) connection for S7-1200 PLC with firmware version 4.5, you must ensure that TLS security is disabled.



7.8 Configuration Version 1.1 vs 1.2

The SIMATIC S7 Connector Configurator provides you two configuration versions' support. You can select configuration version 1.1 or 1.2 as required.

The following table provides you the comparative view of the functionalities which differ in configuration version 1.1 and 1.2.

Functionality	Configuration Version 1.1	Configuration Version 1.2
Metadata payload for S7 Char array	Data type of individual element is published as 'Char Array' in tag metadata payload.	Data type of individual element is published as 'Char' in tag metadata payload.
Tag data payload for S7 Char array	With publish mode 'Bulk Publish', all elements are grouped together and published in a group. Any change in notification of any one tag results in publishing of all the elements.	The individual child elements data is published separately. Any change in notifications of one child element results in publishing of only that child tag value.
Quality code in tag data payload	Quality code with enum value {0,1,2, or 3} is published in tag data payload.	Quality code with enum value {0,1,2, or 3} is published in tag data payload. But along with <code>qc</code> , there is a new field <code>qx</code> which holds all the bits data: quality code, sub status, extended substatus, flag, and limit. <ul style="list-style-type: none"> • Limits bits 0,1 • Sub-status bits 2..5 • Quality bits 6,7 • Extended sub-status 8..11 • Flags 12..15
Value in tag data payload	Value is published as 'String'.	Value is published as per the data type and not in 'String'. For more information, refer Data Types (Page 77).
Value in tag data payload for tag write feature in mqtt publisher node	Value is published in 'String' while making a 'write' request to SIMATIC S7 Connector.	Value is published as per the data type while making a 'write' request to SIMATIC S7 Connector.

Other functionalities for configuration version 1.1 remain same as configuration version 1.2 as described in this manual.

7.9 Data Types

This section provides the information on the data types that are used in adding the tags. For more information about the address formatting of data types, you must refer "TIA Portal Documentation".

The following table describes how the tag data is published:

Siemens Data Type	JSON Data Type	
BOOL	INTEGER	
INT		
DINT		
USINT		
UINT		
UDINT		
BYTE		
WORD		
DWORD		
SINT		
REAL		REAL
LREAL	DOUBLE	
LINT	STRING	
ULINT		
STRING		
LTIME		
CHAR		
LWORD		
TIME		
TOD		
LTOD		
CHAR Array in S7		
DynArray in OPCUA		
DATETIME		STRING (ISO 8601 Zulu (UTC) Format)
DATE		

BOOL (bit)

An operand of data type BOOL represents a bit value and contains one of the following values:

- TRUE
- FALSE

The following table shows the properties of data type BOOL:

Length (bits)	Format	Value range	Examples of value input
1	Boolean	FALSE or TRUE BOOL#0 or BOOL#1 BOOL#FALSE or BOOL#TRUE	TRUE BOOL#1 BOOL#TRUE
	Unsigned integers (decimal system)	0 or 1	1
	Binary numbers	2#0 or 2#1	2#0
	Octal numbers	8#0 or 8#1	8#1
	Hexadecimal numbers	16#0 or 16#1	16#1

Note

Applies to CPUs of the S7-1500 series

For a block with the block property "Optimized block access", the bit has a length of 1 byte.

BYTE

An operand of data type BYTE is a bit string of 8 bits.

The following table shows the properties of data type BYTE:

Length (bits)	Format	Value range	Examples of value input	
			Constants	Absolute and symbolic addresses
8	Integers ¹⁾ (decimal system)	Signed integers: -128 to +127 Unsigned integers: 0 to 255	<ul style="list-style-type: none"> • 15 • BYTE#15 • BYTE#10#15 • B#15 	<ul style="list-style-type: none"> • IB2 • MB10 • DB1.DBB4 • Tag_Name
	Binary numbers	2#0 to 2#1111_1111	<ul style="list-style-type: none"> • 2#0000_1111 • BYTE#2#0000_1111 • B#2#0000_1111 	
	Octal numbers	8#0 to 8#377	<ul style="list-style-type: none"> • 8#17 • BYTE#8#17 • B#8#17 	
	Hexadecimal numbers	16#0 to 16#FF	<ul style="list-style-type: none"> • 16#0F • BYTE#16#0F • B#16#0F 	
¹⁾ The value range depends on the relevant interpretation or conversion.				

Note

The BYTE data type cannot be compared for more than or less than. It can only be supplied with the same decimal data that can be processed by the SINT and USINT data types.

CHAR

A tag of the CHAR (Character) data type has a length of 8 bits and occupies one BYTE of memory.

The CHAR data type stores a single character in ASCII coding. You can find information on the encoding of special characters under "STRING".

The following table shows the value range of the CHAR data type:

Length (bits)	Format	Value range	Example of value inputs
8	ASCII characters	ASCII character set	'A', CHAR#'A'

DATE

The DATE data type saves the date as an unsigned integer. The representation contains the year, the month, and the day.

The contents of an operand of DATE data type correspond in hexadecimal format to the number of days since 01-01-1990 (16#0000).

The following table shows the properties of data type DATE:

Length (bytes)	Format	Value range	Example of value inputs
2	IEC date (Year-Month-Day)	D#1990-01-01 to D#2169-06-06	D#2009-12-31, DATE#2009-12-31

DINT (32-bit integers)

An operand of data type DINT (Double INT) has a length of 32 bits and consists of two components: a sign and a numerical value in the two's complement. The signal states of bits 0 to 30 represent the number value. The signal state of bit 31 represents the sign. The sign may assume "0" for the positive, or "1" for the negative signal state.

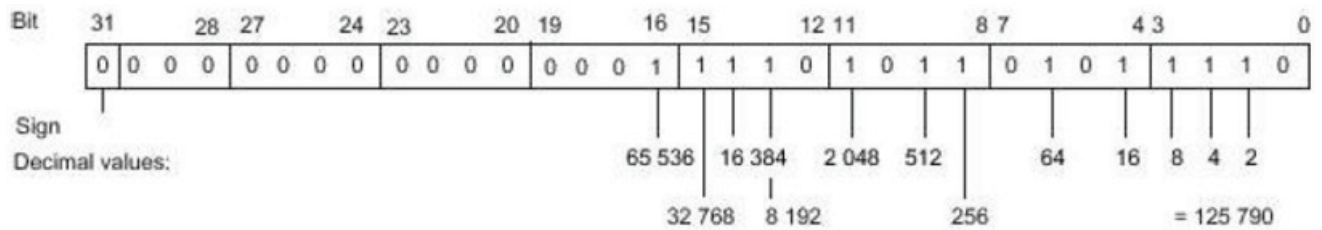
An operand of data type DINT occupies four BYTE in the memory.

The following table shows the properties of data type DINT:

Length (bits)	Format	Value range	Examples of value input
32	Signed integers (decimal system)	-2_147_483_648 to +2_147_483_647	<ul style="list-style-type: none"> +125_790 DINT#+125_790 DINT#10#+125_790 L#275
	Binary numbers (only positive)	2#0 to 2#0111_1111_1111_1111_111 1_1111_1111_1111	<ul style="list-style-type: none"> 2#0000_0000_0000_0001_1110_1011_0101_1110 DINT#2#0000_0000_0000_0001_1110_1011_0101_1110 DINT#2#10
	Octal numbers (only positive)	8#0 to 8#177_7777_7777	<ul style="list-style-type: none"> 8#36_5536 DINT#8#36_5536
	Hexadecimal numbers	16#0 to 16#7FFF_FFFF	<ul style="list-style-type: none"> 16#0001_EB5E DINT#16#0001_EB5E

Example

The following figure shows the integer +125790 as a binary number:



DWORD

An operand of data type DWORD is a bit string of 32 bits.

The following table shows the properties of data type DWORD:

Length (bits)	Format	Value range	Examples of value input	
			Constants	Absolute and symbolic addresses
32	Integers (decimal system)	Signed integers: -2_147_483_647 to +2_147_483_647 Unsigned integers: 0 to 4_294_967_295	<ul style="list-style-type: none"> +15_793_935 DWORD# +15_793_935 DWORD#10# +15_793_935 DW#+15_793_935 	<ul style="list-style-type: none"> MD10 DB1.DB8 Tag_Name
	Binary numbers	2#0 to 2#1111_1111_1111_1 111_1111_1111_1111_1111	<ul style="list-style-type: none"> 2#0000_0000_1111_0000_1111_1111_0000_1111 DWORD#2#0000_0000_1111_0000_1111_1111_0000_1111 DW#2#0000_0000_1111_0000_1111_1111_0000_1111 	
	Octal numbers	8#0 to 8#37_777_777_777	<ul style="list-style-type: none"> 8#74_177_417 DWORD#8#74_177_417 DW#8#74_177_417 	
	Hexadecimal numbers	16#0000_0000 to 16#FFFF_FFFF	<ul style="list-style-type: none"> 16#00F0_FF0F DWORD#16#00F0_FF0F DW#16#00F0_FF0F 	
	Decimal sequence	B#(0, 0, 0, 0) to B#(255, 255, 255, 255)	B#(127, 200, 127, 200)	

Note

The DWORD data type cannot be compared for more than or less than. It can only be supplied with the same decimal data that can be processed by the DINT and UDINT data types.

INT (16-bit integers)

An operand of data type INT has a length of 16 bits and consists of two components: a sign and a numerical value in the two's complement. The signal states of bits 0 to 14 represent the number value. The signal state of bit 15 represents the sign. The sign may assume "0" for the positive, or "1" for the negative signal state.

An operand of data type INT occupies two BYTE in the memory.

The following table shows the properties of data type INT:

Length (bits)	Format	Value range	Examples of value input
16	Signed integers (decimal system)	-32_768 to +32_767	<ul style="list-style-type: none"> • +3_785 • INT#+3_785 • INT#10#+3_785
	Binary numbers (only positive)	2#0 to 2#0111_1111_1111_1111	<ul style="list-style-type: none"> • 2#0000_1110_1100_1001 • INT#2#0000_1110_1100_1001 • INT#2#10
	Octal numbers (only positive)	8#0 to 8#7_7777	<ul style="list-style-type: none"> • 8#7311 • INT#8#7311
	Hexadecimal numbers (only positive)	16#0 to 16#7FFF	<ul style="list-style-type: none"> • 16#0EC9 • INT#16#0EC9

LINT (64-bit integers)

An operand of data type LINT (Long INT) has a length of 64 bits and consists of two components: a sign and a numerical value in the two's complement. The signal states of bits 0 to 62 represent the number value. The signal state of bit 63 represents the sign. The sign may assume "0" for the positive, or "1" for the negative signal state.

An operand of data type LINT occupies eight BYTE in the memory.

The following table shows the properties of data type LINT:

Length (bits)	Format	Value range	Examples of value input
64	Signed integers (decimal system)	-9_223_372_036_854_775_808 to +9_223_372_036_854_775_807	<ul style="list-style-type: none"> +154_325_790_816_159 LINT# +154_325_790_816_159 LINT#10# +154_325_790_816_159
	Binary numbers (only positive)	2#0 to 2#0111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111	<ul style="list-style-type: none"> 2#0000_0000_0000_0000_1000_1100_0101_1011_1100_0101_1111_0000_1111_0111_1001_1111 LINT#2#0000_0000_0000_0000_1000_1100_0101_1011_1100_0101_1111_0000_1111_0111_1001_1111 LINT#2#10
	Octal numbers (only positive)	8#0 to 8#7_7777_7777_7777_7777_7777	<ul style="list-style-type: none"> 8#4305_5705_7417_3637 LINT#8#4305_5705_7417_3637
	Hexadecimal numbers (only positive)	16#0 to 16#7FFF_FFFF_FFFF_FFFF	<ul style="list-style-type: none"> 16#0000_8C5B_C5F0_F79F LINT#16#0000_8C5B_C5F0_F79F

LREAL

Operands of the data type LREAL have a length of 64 bits and are used to represent floating-point numbers. An operand of the LREAL data type consists of the following three components:

- Sign: The sign is determined by the signal state of bit 63. The bit 63 assumes the value "0" (positive) or "1" (negative).
- 11-bit exponents to base 2: The exponent is increased by a constant (base, +1023), so that it has a value range of 0 to 2047.
- 52-bit mantissa: Only the fraction part of the mantissa is shown. The integer part of the mantissa is always 1 with normalized floating-point numbers and is not stored.

The LREAL data type is processed with a precision of 15 digits.

The following table shows the properties of data type LREAL:

Length (bits)	Format	Value range	Examples of value input
64	Floating-point numbers according to IEEE754	-1.7976931348623157e+308 to -2.2250738585072014e-308	1.0e-5; LREAL#1.0e-5
	Floating-point numbers	±0.0 +2.2250738585072014e-308 to +1.7976931348623157e+308	1.0; LREAL#1.0

Note

With floating-point numbers, only the precision defined by the IEEE754 standard is stored. Additionally specified decimals are rounded off according to IEEE754.

The number of decimal places may decrease for frequently nested arithmetic calculations.

If more decimal places are specified than can be stored by the data type, the number is rounded to the corresponding value of the precision allowed by this value range.

LTIME (IEC time)

The contents of an operand of data type LTIME is interpreted as nanoseconds. The representation contains information for days (d), hours (h), minutes (m), seconds (s) and milliseconds (ms), microseconds (us), and nanoseconds (ns).

The following table shows the properties of data type LTIME:

Length (bits)	Format	Value range	Examples of value input
64	Signed duration	LT#-106751d_23h_47m_16s_854ms_775us_808ns to LT# +106751d_23h_47m_16s_854ms_775us_807ns	LT#11350d_20h_25m_14s_830ms_652us_315ns, LTIME#11350d_20h_25m_14s_830ms_652us_315ns

It is not necessary to specify all time units. LT#5h10s is therefore a valid entry, for example. If only one unit is specified, the absolute value of days, hours, and minutes must not exceed the high or low limits. When more than one time unit is specified, the value must not exceed 106751 days, 23 hours, 59 minutes, 59 seconds, 999 milliseconds, 999 microseconds, or 999 nanoseconds.

Note

In SIMATIC S7 Connector Configurator, LTIME value is published in nanoseconds which represents a duration in 100 nanosecond intervals.

LTOD (LTIME_OF_DAY)

Data type LTOD (LTIME_OF_DAY) occupies two double words and stores the number of nanoseconds since the beginning of the day (0:00 h) as unsigned integer.

The following table shows the properties of data type LTOD:

Length (bytes)	Format	Value range	Examples of value input
8	Time-of-day (hours:minutes: seconds.nanoseconds)	LTOD#00:00:00.0000000 00 to LTOD#23:59:59.9999999 99	LTOD#10:20:30.400_365_21 5, LTIME_OF_DAY#10:20:30.400_365_215

You always need to specify the hours, minutes, and seconds. The specification of nanoseconds is optional.

Note

In SIMATIC S7 Connector Configurator, LTOD value is published in nanoseconds which represents a duration in 100 nanosecond intervals.

LWORD

An operand of data type LWORD is a bit string of 64 bits.

The following table shows the properties of data type LWORD:

Length (bits)	Format	Value range	Examples of value input
64	Integers (decimal system)	Signed integers: -9_223_372_036_854_775_808 to +9_223_372_036_854_775_807 Unsigned integers: 0 to 18_446_744_073_709_551_615	<ul style="list-style-type: none"> +26_123_590_360_715 LWORD# +26_123_590_360_715 LWORD#10# +26_123_590_360_715 LW#+26_123_590_360_715
	Binary numbers	2#0 to 2#1111_1111_1111_1111_1111_11 1 1_1111_1111_1111_1111_1111_11 11_1111_1111_1111_1111	<ul style="list-style-type: none"> 2#0000_0000_0000_0000_0000_0000_1011_1110_0001_0010_1111_0101_0010_1101_1110_1000_1011 LWORD#2#0000_0000_0000_0000_0000_0000_1011_1110_0001_0010_1111_0101_0010_1101_1110_1000_1011 LW#2#0000_0000_0000_0000_0000_0000_1011_1110_0001_0010_1111_0101_0010_1101_1110_1000_1011
	Octal numbers	8#0 to 8#1_777_777_777_777_777_777_777	<ul style="list-style-type: none"> 8#13_724_557_213 LWORD#8#13_724_557_213 LW#8#13_724_557_213
	Hexadecimal numbers	16#0000_0000 to 16#FFFF_FFFF_FFFF_FFFF	<ul style="list-style-type: none"> 16#0000_0000_5F52_DE8B LWORD#16#0000_0000_5F52_DE8B LW#16#0000_0000_5F52_DE8B
	Decimal sequence	B#(0, 0, 0, 0, 0, 0, 0, 0) to B#(255, 255, 255, 255, 255, 255, 255, 255)	B#(127, 200, 127, 200, 127, 200, 127, 200)

Note

The LWORD data type cannot be compared for more than or less than. It can only be supplied with the same decimal data that can be processed by the LINT and ULINT data types.

REAL

Operands of the data type REAL have a length of 32 bits and are used to represent floating-point numbers. An operand of the REAL data type consists of the following three components:

- Sign: The sign is determined by the signal state of bit 31. The bit 31 assume the value "0" (positive) or "1" (negative).
- 8-bit exponents to basis 2: The exponent is increased by a constant (base, +127), so that it has a value range of 0 to 255.
- 23-bit mantissa: Only the fraction part of the mantissa is shown. The integer part of the mantissa is always 1 with normalized floating-point numbers and is not stored.

The REAL data type is processed with a precision of 6 digits.

Note

With floating-point numbers, only the precision defined by the IEEE754 standard is stored. Additionally specified decimals are rounded off according to IEEE754.

The number of decimal places may decrease for frequently nested arithmetic calculations.

If more decimal places are specified than can be stored by the data type, the number is rounded to the value corresponding to the precision allowed by this value range.

The following table shows the properties of data type REAL:

Length (bits)	Format	Value range	Examples of value input
32	Floating-point numbers according to IEEE754	-3.402823e+38 to -1.175495e-38 ±0.0	1.0e-5; REAL#1.0e-5
	Floating-point numbers	+1.175495e-38 to +3.402823e+38	1.0; REAL#1.0

SINT (8-bit integers)

An operand of data type SINT (Short INT) has a length of 8 bits and consists of two components: a sign and a numerical value in the two's complement. The signal states of bits 0 to 6 represent the number value. The signal state of bit 7 represents the sign. The sign may assume "0" for the positive, or "1" for the negative signal state.

An operand of data type SINT occupies one BYTE in the memory.

The following table shows the properties of data type SINT:

Length (bits)	Format	Value range	Examples of value input
8	Signed integers (decimal system)	-128 to +127	<ul style="list-style-type: none"> +44 SINT#+44 SINT#10#+44 <p>The value range extends to a maximum of SINT#255 when using the type SINT#. This value is interpreted as an integer with -1.</p>
	Binary numbers (only positive)	2#0 to 2#0111_1111	<ul style="list-style-type: none"> 2#0010_1100 SINT#2#0010_1100 SINT#2#10
	Octal numbers (only positive)	8#0 to 8#177	<ul style="list-style-type: none"> 8#54 SINT#8#54
	Hexadecimal numbers (only positive)	16#0 to 16#7F	<ul style="list-style-type: none"> 16#2C SINT#16#2C <p>The value range extends to a maximum of SINT#16#FF when using the type SINT#. This value is interpreted as an integer with -1.</p>

STRING

An operand of the STRING data type saves several characters in a character string that can consist of up to 254 characters. In a character string, all characters of the codepage created on the system are permitted. The characters are specified in single quotation marks.

A character string can also contain special characters. The escape character \$ is used to identify control characters, dollar signs, and single quotation marks.

Note

Different code pages

Please note that the special characters are coded using the code page currently set in Windows. This means that a string that contains special characters can be displayed differently on a different operating system with a different code page.

The dependency of the codepage on the created system makes an international use of the user program more difficult. Only the characters from the 7-bit ASCII coding are internationally valid.

7.9 Data Types

The following table shows the properties of a STRING tag:

Length (bytes)	Format	Value range	Examples of value input
$n + 2$ ¹⁾	ASCII character string incl. special characters	0 to 254 characters	<ul style="list-style-type: none"> 'Name' STRING#'NAME' STRING#'Na... (The actual length of the string is longer than the space available on the screen.) STRING#" (The string is empty.)

¹⁾ An operand of the STRING data type occupies two bytes more than the specified maximum length in the memory.

TIME (IEC time)

The contents of an operand of the data type TIME is interpreted as milliseconds. The representation contains information for days (d), hours (h), minutes (m), seconds (s), and milliseconds (ms).

The following table shows the properties of data type TIME:

Length (bits)	Format	Value range	Examples of value input
32	Signed duration	T#-24d_20h_31m_23s_648ms to T#+24d_20h_31m_23s_647ms	T#10d_20h_30m_20s_630ms, TIME#10d_20h_30m_20s_630ms

It is not necessary to specify all time units. T#5h10s is a valid entry, for example. If only one unit is specified, the absolute value of days, hours, and minutes must not exceed the high or low limits. When more than one time unit is specified, the value must not exceed 24 days, 23 hours, 59 minutes, 59 seconds, or 999 milliseconds.

Note

In SIMATIC S7 Connector Configurator, TIME value is published in nanoseconds which represents a duration in 100 nanosecond intervals.

TIME_OF_DAY (TOD)

Data type TOD (TIME_OF_DAY) occupies a double word and stores the number of milliseconds since the beginning of the day (0:00 h) as unsigned integer.

The following table shows the properties of data type TOD:

Length (bytes)	Format	Value range	Examples of value input
4	Time-of-day (hours:minutes: seconds.milliseconds)	TOD#00:00:00.000 to TOD#23:59:59.999	TOD#10:20:30.400, TIME_OF_DAY#10:20:30.40 0

You always need to specify the hours, minutes, and seconds. The specification of milliseconds is optional.

Note

In SIMATIC S7 Connector Configurator, TOD value is published in nanoseconds which represents a duration in 100 nanosecond intervals.

UDINT (32-bit integers)

An operand of data type UDINT (Unsigned Double INT) has a length of 32 bits and contains unsigned numerical values.

An operand of data type UDINT occupies four BYTE in the memory.

The following table shows the properties of data type UDINT:

Length (bits)	Format	Value range	Examples of value input
32	Unsigned integers (decimal system)	0 to 4_294_967_295	<ul style="list-style-type: none"> • 4_042_322_160 • UDINT#4_042_322_160 • UDINT#10#4_042_322_160
	Binary numbers	2#0 to 2#1111_1111_1111_1111_1111_1111_1111_1111	<ul style="list-style-type: none"> • 2#1111_0000_1111_0000_1111_1_0000_1111_0000 • UDINT#2#1111_0000_1111_0_000_1111_0000_1111_000_0 • UDINT#2#10
	Octal numbers	8#0 to 8#377_7777_7777	<ul style="list-style-type: none"> • 8#360_7417_0360 • UDINT#8#360_7417_0360
	Hexadecimal numbers	16#0 to 16#FFFF_FFFF	<ul style="list-style-type: none"> • 16#F0F0_F0F0 • UDINT#16#F0F0_F0F0

UINT (16-bit integers)

An operand of data type UINT (Unsigned INT) has a length of 16 bits and contains unsigned numerical values.

An operand of data type UINT occupies two BYTE in the memory.

The following table shows the properties of data type UINT:

Length (bits)	Format	Value range	Examples of value input
16	Unsigned integers (decimal system)	0 to 65_535	<ul style="list-style-type: none"> • 65_295 • UINT#65_295 • UINT#10#65_295
	Binary numbers	2#0 to 2#1111_1111_1111_1111	<ul style="list-style-type: none"> • 2#1111_1111_0000_1111 • UINT#2#1111_1111_0000_1 111 • UINT#2#10
	Octal numbers	8#0 to 8#17_7777	<ul style="list-style-type: none"> • 8#17_7417 • UINT#8#17_7417
	Hexadecimal numbers	16#0 to 16#FFFF	<ul style="list-style-type: none"> • 16#FF0F • UINT#16#FF0F

ULINT (64-bit integers)

An operand of data type ULINT (Unsigned Long INT) has a length of 64 bits and contains unsigned numerical values.

An operand of data type ULINT occupies eight BYTE in the memory.

The following table shows the properties of data type ULINT:

Length (bits)	Format	Value range	Examples of value input
64	Unsigned integers (decimal system)	0 to 18_446_744_073_709_551_6 15	<ul style="list-style-type: none"> • 154_325_790_816_159 • ULINT#154_325_790_816_159 • ULINT#10#154_325_790_816_159
	Binary numbers	2#0 to 2#1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111_1111	<ul style="list-style-type: none"> • 2#0000_0000_0000_0000_1000_1100_0101_1011_1100_0101_1111_0000_1111_0111_1001_1111 • ULINT#2#0000_0000_0000_0000_1000_1100_0101_1011_1100_0101_1111_0000_0111_1001_1111 • ULINT#2#10
	Octal numbers	8#0 to 8#17_7777_7777_7777_7777_7777	<ul style="list-style-type: none"> • 8#4305_5705_7417_3637 • ULINT#8#4305_5705_7417_3637
	Hexadecimal numbers	16#0 to 16#FFFF_FFFF_FFFF_FFFF	<ul style="list-style-type: none"> • 16#0000_8C5B_C5F0_F79F • ULINT#16#0000_8C5B_C5F0_F79F

USINT (8-bit integers)

An operand of data type USINT (Unsigned Short INT) has a length of 8 bits and contains unsigned numerical values:

An operand of data type USINT occupies one BYTE in the memory.

7.9 Data Types

The following table shows the properties of data type USINT:

Length (bits)	Format	Value range	Examples of value input
8	Unsigned integers (decimal system)	0 to 255	<ul style="list-style-type: none"> • 78 • USINT#78 • USINT#10#78
	Binary numbers	2#0 to 2#1111_1111	<ul style="list-style-type: none"> • 2#0100_1110 • USINT#2#0100_1110 • USINT#2#10
	Octal numbers	8#0 to 8#377	<ul style="list-style-type: none"> • 8#116 • USINT#8#116
	Hexadecimal numbers	16#0 to 16#FF	<ul style="list-style-type: none"> • 16#4E • USINT#16#4E

WORD

An operand of data type WORD is a bit string of 16 bits.

The following table shows the properties of data type WORD:

Length (bits)	Format	Value range	Examples of value input	
			Constants	Absolute and symbolic addresses
32	Integers (decimal system)	Signed integers: -32_768 to +32_767 Unsigned integers: 0 to 65_535	<ul style="list-style-type: none"> • 61_680 • WORD#61_680 • WORD#10#61_6 80 • W#61_680 	<ul style="list-style-type: none"> • MW10 • DB1.DBW2 • Tag_Name
	Binary numbers	2#0 to 2#1111_1111_1111_1111	<ul style="list-style-type: none"> • 2#1111_0000_1111_0000 • WORD#2#1111_0000_1111_0000 • W#2#1111_0000_1111_0000 	
	Octal numbers	8#0 to 8#177_777	<ul style="list-style-type: none"> • 8#170_360 • WORD#8#170_360 • W#8#170_360 	
	Hexadecimal numbers	16#0 to 16#FFFF	<ul style="list-style-type: none"> • 16#F0F0 • WORD#16#F0F0 • W#16#F0F0 	
	BCD	C#0 to C#999	C#55	
	Decimal sequence	B#(0, 0) to B#(255, 255)	B#(127, 200)	

Note

The WORD data type cannot be compared for more than or less than. It can only be supplied with the same decimal data that can be processed by the INT and UINT data types.

7.10 Notes on use

The following restrictions apply with the delivery of the SIMATIC S7 Connector Configurator.

Configuration update

Whenever you change the configuration in the SIMATIC S7 Connector Configurator, the system app on the respective Industrial Edge Device (IED) gets updated only when you click on the "Deploy" button.

You can check the status of the configuration update using the tasks icon in the IED UI.

Multiple user access to system configurators

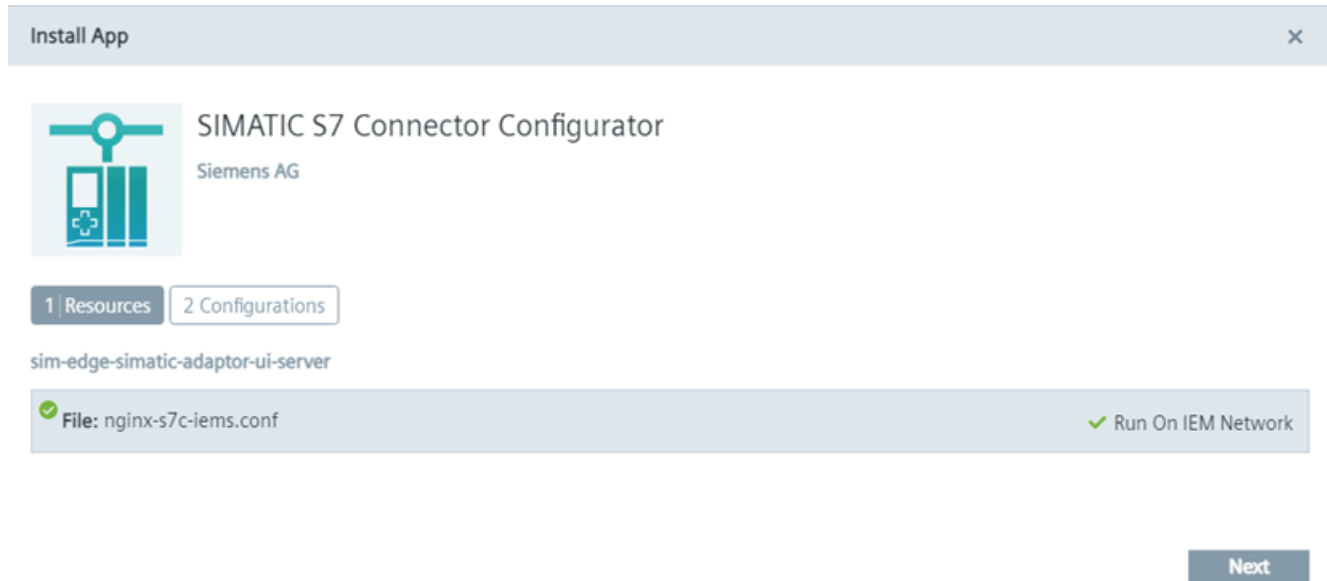
For the system configurators, it is not possible to use the same system configurator of an IED with two or more users' instances at the same time. When another user launches a system configurator at the same time while a current user is working with the same system configurator, the second user will be blocked, and the primary user continues working with the system configurator. Just when the primary user closes the system configurator, the second user can launch the same system configurator.

Supported language

When you provide the needed information in input fields, the Industrial Edge Management (IEM), the system configurators, and other components only support characters from the English language. For example, when you add a data source and tag name with special characters like "ö", "ä" or "ü", these characters are not supported and result in error messages.

Updating system configurators

- When you update the system configurators in the IEM, there is no specific defined sequence for updating the configurators. For the update sequence of other components, refer <https://support.industry.siemens.com/cs/document/10979314>
- If you are using IEM version v1.2, then all the system configurators must be updated to version v1.2.
- When you update the system configurators, you must ensure to select the 'nginx conf' file during the process. The selected file is displayed with green-check circle as follows:



Before you update the system configurators, take a snapshot of the cluster respectively the VM.

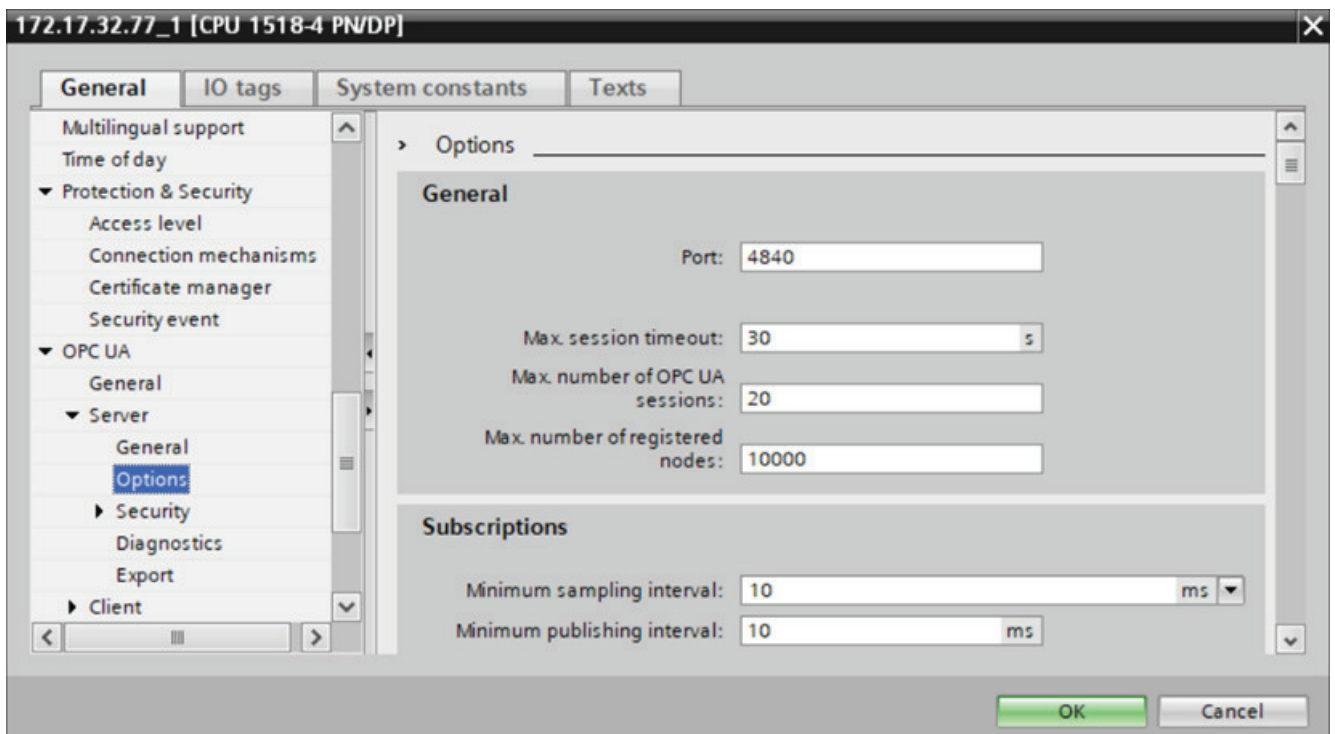
Job management for system configurators

The job management for all system configurators is not handled by the system configurators themselves but by the IEM. In case of any job failure, you will be notified by the configurators. In that case, check the jobs in the IED UI respectively in the IEM and delete the failed or skipped jobs from the "Job Status" screen.

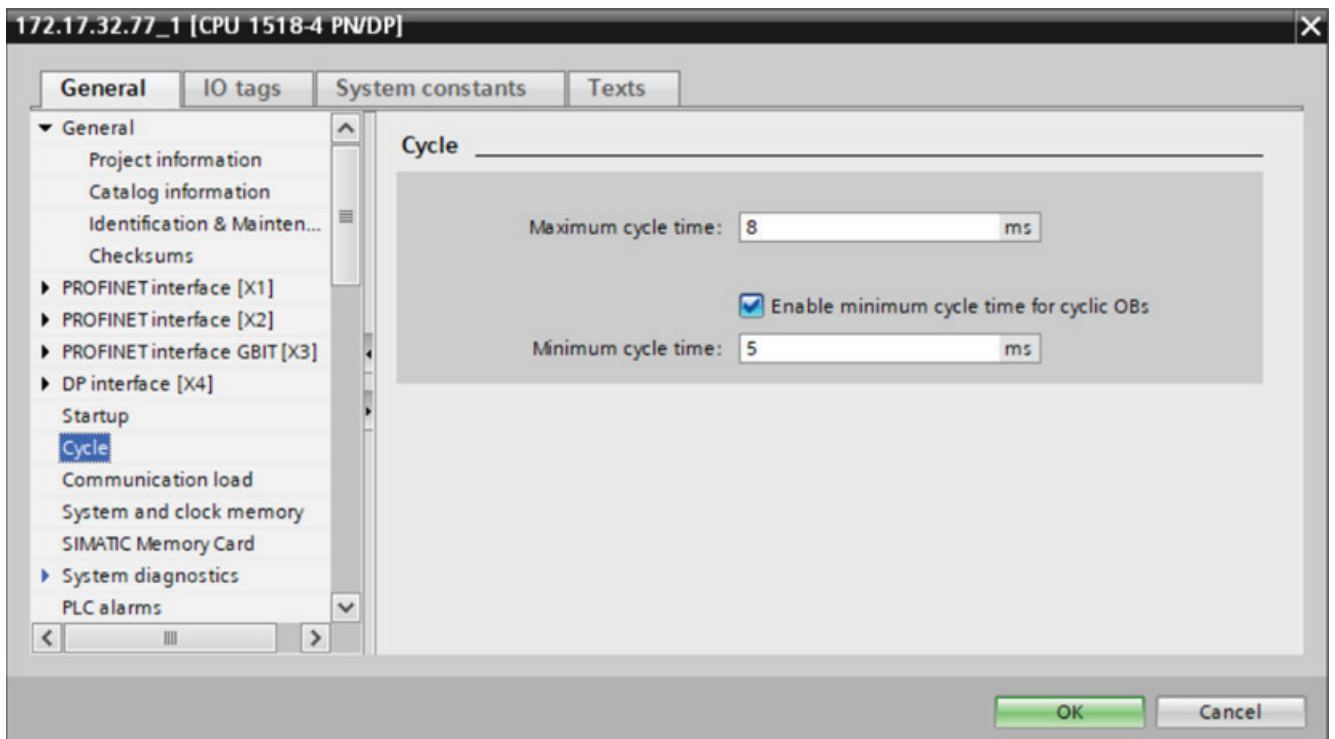
High speed acquisition cycle support in OPC-UA

You can configure OPC-UA tags with acquisition cycle of 10 milliseconds. You must follow below criteria to get tags at 10 milliseconds cycle:

PLC Type: PLC with OPC-UA server which supports "Minimum sampling interval" and "Minimum publishing interval" of 10 milliseconds. For example, CPU 1518-4 PN/DP.



Cycle time: For testing 10 milliseconds acquisition cycle, cycle time must be greater than 5 milliseconds and less than 9 milliseconds.



CPU Spike Issue

Under resource constraints, the tags may be repeated with different timestamps within the same packet. This behavior can happen temporarily under sudden spike in resource usage of other processes limiting the resources availability for SIMATIC S7 Connector. To reproduce this scenario, follow these steps:

1. Configure an IED VM with 1CPU core count and 2GB RAM.
2. Install SIMATIC S7 Connector, IE Databus, IE Flow Creator, and IE Cloud Connector applications.
3. Configure datapoints in SIMATIC S7 Connector with low acquisition cycle supported by the connection.

If tags are repeated and you are interested only in the newest data available in the packet, then follow these steps:

- Parse the tag data packet received from the broker.
- Create a map with "id" as the key to hold the unique tag values.
- In case "id" is repeated, replace the tag data which has maximum "ts" value in the packet.

If tags are repeated and you are interested in all the packets with no data loss, then follow these steps:

- Parse the tag data packet received from the broker.
- Create a multi-map with "id" as the key to hold the multiple tag values for the same tag.
- Sort the tags with increasing order of timestamp before processing/storage.

Data Publishing Interval

TIME, LTIM, TOD, and LTOD values are published as nanosecond in 100 nanosecond intervals.

7.11 Known issues

The following are the known issues:

Issue	Troubleshooting
The delta download of S7+ connections for legitimization from "NoAccess" to "FullAccess" does not work.	-
You will get no response from the UI after you click on "Deploy", "Start", and "Stop". In that case, the operation remains in progress state and buttons are disabled.	Close and reopen the SIMATIC S7 Connector Configurator. Perform again the required operation.
The following characters are not supported for tag and connection names: # : > < [] „ " + / \ . \$	-
S7-1200 PLCs are not supported for OPC-UA connection in the S7 Connector Configurator.	-

Issue	Troubleshooting
<p>The following data types are not supported for the "TagWrite" operation:</p> <ul style="list-style-type: none"> • S7 data sources: DATE, TIME, and CHAR ARRAY • S7+ data sources: DATE • OPC-UA data sources: ARRAY, SInt, UInt, and USInt 	-
The "DateTime" data type is not supported for S7-1200 PLCs.	-
Editing of "CHAR ARRAY" data types for S7 connections is not supported.	-
<p>The browse feature is provided as preview for ease of configuration; hence the following issues are known with regard to this feature:</p> <ul style="list-style-type: none"> • When the browsing data source (OPC-UA server or PLC) has a large number of tags, it takes approximately 2 minutes to display the tag list. • When the browsing data source (OPC-UA server or PLC) has not responded within 2 minutes, a time out error occurs. • When you browse S7+ data sources, editing these data sources and browsed tags are not supported. 	<p>Configure manually the tags or import the exported files from the TIA Portal.</p> <p>The browse feature is provided as beta functionality in V1.2.0, the complete feature will be available in V1.3.0.</p>
If the browsing feature fails due to any reason after several attempts.	Configure manually the tags or import the exported files from the TIA Portal.
Browsing S7+ data sources is not supported for ET200 Open Control PLCs.	-
Manual adding of symbolic addresses for S7+ tags is not supported.	Symbolic addresses are only supported in browsing S7+ data sources.
Delta download of full text alarms is not supported.	Full text alarms are only supported in S7-300 and S7-400 PLCs in the full download configuration. To get full text alarms, stop the project and download the project with S7-300 or S7-400 PLCs.
When you browse an OPC-UA server from a project in the TIA Portal with 1 function block (FB) and several data block (DB) instances, only data of 1 DB instance is returned and thus results in false read data.	Configure manually the OPC-UA tags or use the exported "*.xml" file of the DB instances from the TIA Portal.
Deleting a configuration of the SIMATIC S7 Connector by clicking the "Delete Configuration" button under the "My Installed Apps" menu entry starts a task on the according Edge Device but fails immediately.	-
A maximum array size of 999 is supported for the "CHAR ARRAY" data type in S7 connections.	-
If you import a configuration file, which was exported from the IEM through a SIMATIC S7 Connector Configurator version less than 1.0.6, into the SIMATIC S7 Connector Configurator, full downloads fail if there are S7+ tags with symbolic address.	Use imported files which are exported from a version \geq 1.0.6. Browse all S7+ connections and then click "Deploy".
Connection names should be unique and names within a connection tag should also be unique.	-

7.11 Known issues

Issue	Troubleshooting
If you import a configuration file, which was exported from the IEM through a SIMATIC S7 Connector Configurator version less than 1.0.9, into the SIMATIC S7 Connector Configurator, full and delta downloads fail if you use duplicate tag names in the same connection or if you use tag names with unsupported characters.	Use imported files which are exported from a version $\geq 1.0.9$.
<p>Tags that are displayed in the invalid table after you click on the details link in the import or browse dialog window, follow the following priority:</p> <ul style="list-style-type: none"> • All tags that have duplicate names in the import file or that were browsed are listed first • Remaining tags that are present in the import file or that were browsed and tags that are already present in the configurator are listed next • From the remaining list, tags with unsupported data types are listed • Finally, from the remaining list, tags that contain unsupported characters in their names are listed 	-
<p>If column widths were resized previously, the column widths will be reset to default size due to the following scenarios:</p> <ul style="list-style-type: none"> • Browser window is resized • Browser window is refreshed • Configurator is relaunched 	Resize the column widths again.
If the number of tags is more than 4000 or if you import respectively browse tags, the response of resizing columns becomes slow, and you may observe flickering of the UI.	-
If before completion of the mass deployment operation, you launch the UI in one of the selected Industrial Edge Devices, then new configuration is not reflected in the Industrial Edge Device.	You must wait for the completion of mass deployment operation before launching UI in other Industrial Edge Device which was selected as part of the mass deployment.
Mass deployment feature is available only when Industrial Edge Runtime for the selected Industrial Edge Device is not running.	You must manually start, configure project when Industrial Edge Runtime is started, and stop the project by launching the UI of that Industrial Edge Device and perform these operations.
If you select Industrial Edge Device where Industrial Edge Runtime is running, then after the mass deployment the project will be stopped in those Industrial Edge Devices.	You must explicitly start the Industrial Edge Runtime in those Industrial Edge Device once again.
SIMATIC S7 Connector Configurator publishes tag data to broker using unreliable Qos policy -Qos0. This may result in some packet loss.	-
In SIMATIC S7 Connector Configurator, you can browse maximum of 2000 tags in OPC-UA connection. If PLC contains more than 2000 Tags, then "Browse Timeout" error is displayed.	This is the NFR limit of OPC-UA browser. If you want to browse, then you must use a PLC Project with less than 2000 tags.
In SIMATIC S7 Connector Configurator, you can browse maximum of 6000 tags in S7+ connection. If PLC contains more than 6000 tags, then "Browse Timeout" error is displayed.	This is the NFR limit of S7+ browser. If you want to browse, then you must use a PLC Project with less than 6000 tags.
Tag write does not work if it is performed repeatedly (each 100 milliseconds write operation is scheduled) for a group of Tags configured with 100 milliseconds Acquisition Cycle.	You can reduce the frequency of Tag write operation scheduled or increase the Acquisition Cycle.

Issue	Troubleshooting
<p>S7+ browsing is not supported for following PLC type and FW version:</p> <ul style="list-style-type: none"> • CPU 1513F-1 PN V2.8 • CPU 1518-4 PN/DP V2.8 • CPU 1518F-4 PN/DP V2.6.1 • CPU 1512SP-1 PN V2.8 	-
<p>S7+ driver supports maximum of eight connections. In the S7 Connector project, you can configure maximum eight Optimized S7-Protocol (S7-1200/1500).</p>	-
<p>S7 Connector publishes multiple instances of same tag with different timestamp to broker when IED CPU is overloaded. This is done to ensure that no data loss happens even if there is data delay because of low CPU resources.</p>	You must ensure S7 Connector gets enough CPU to process all the packets and send to broker.
<p>In Optimized S7-Protocol (S7-1200/1500), filter is supported only for Datablocks. Tag filter based on M, Q, and I area is not yet supported.</p>	-
<p>Browsing in Optimized S7-Protocol (S7-1200/1500) with legitimization enabled fetches only PLC Tags. Datablock tags are missed.</p>	You must disable the legitimization from the PLC and the SIMATIC S7 Connector Configurator and then browse to fetch both PLC tags and datablock tags.
<p>In OPC-UA data source,</p> <ul style="list-style-type: none"> • Browsing functionality does not support "Browse Filter" with tag name and datablock. • Browsing is possible only of tags with namespace=3. 	You can edit the exported OPC-UA XML file and Import it in SIMATIC S7 Connector Configurator. You can add/modify the tag name, address namespace, and datatype. For more information, refer Export Tags for OPC-UA Connection (Page 63).
<p>The download fails for a project with 6000 tags if the tag name and connection name are long.</p>	It is recommended to use smaller tag names and connection names, if you want to download large number of tags.

7.11 Known issues