



sylvac

OPC UA

English

Instruction manual

Sylcom

OPC-UA

OPC-UA configuration with Sylcom

10100
01101
11011
01001

1. INTRODUCTION

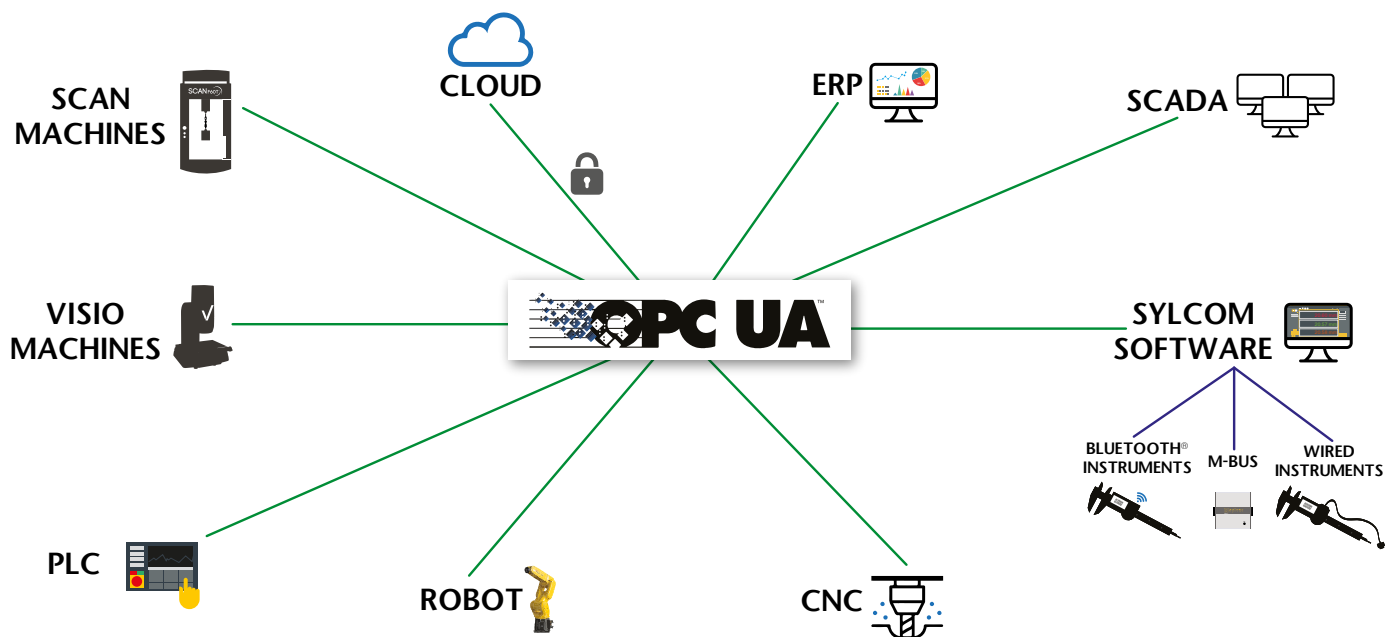
OPC UA is a communication protocol for the automation industry using an Ethernet port. It is the most flexible automation solution supported by Sylcom (e.g. access to the details of a measurement result).

Please note that the Sylcom only supports the binary protocol **opc.tcp**.

2. SYLCOM DEMO WITH OPC UA PROTOCOLE

2.1 The protocole

OPC Unified Architecture (OPC UA) is a communication protocol for industrial automation applications, perfectly adapted to Industry 4.0. It is based on the client-server principle and bridges the gap between the IP-based world of IT and the production floor.



2.2 Introduction

The measurements with Sylcom Expert can be automatized with a Robot. This document provides a notice how to setup the Sylcom software with an OPC UA client.

2.3 Requirements

The following material is required :

- Sylcom Expert

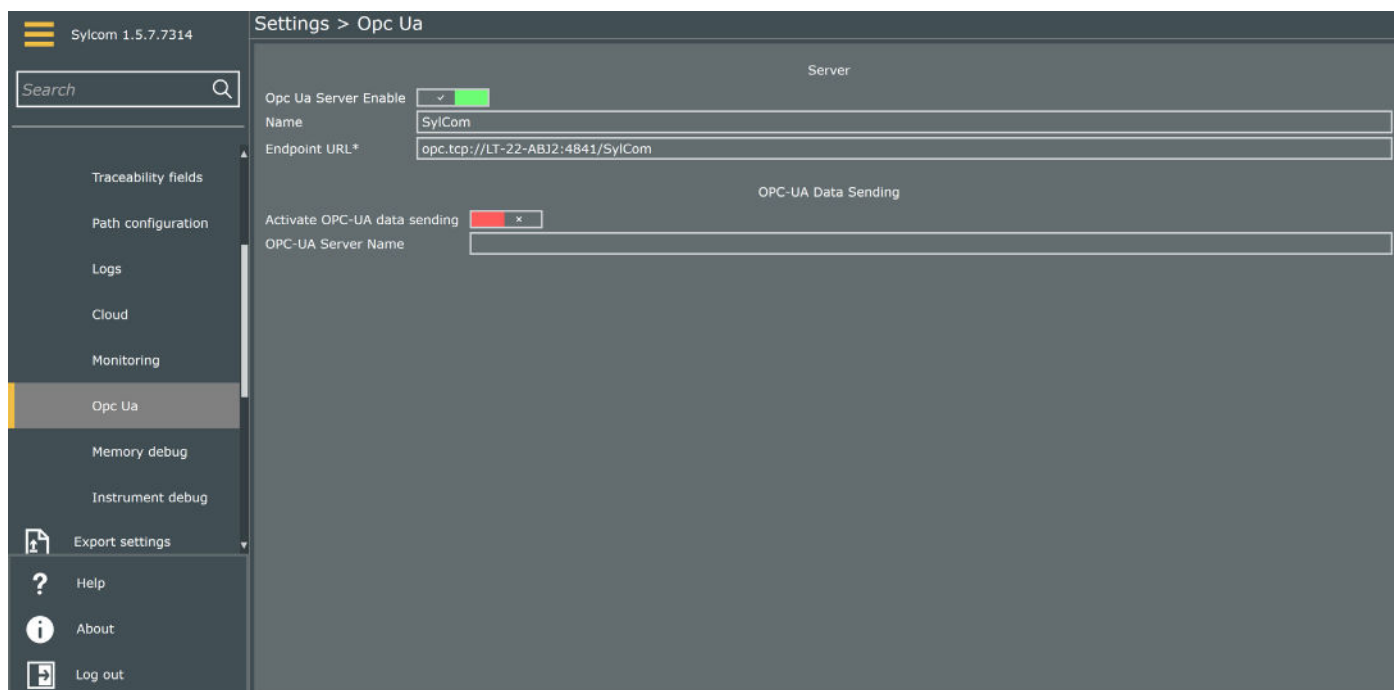
2.4 Installation

You must have the Sylcom Expert version to use the OPC-UA module.

2.5 Setup of the client

In Sylcom :

1. Go to Settings / OPC-UA
2. Enable OPC-UA server
3. Enter the URL
(for example '<opc.tcp://127.0.0.1:4841/SYLCOM>')



In order to use the OPC client, the communication must be setup on OPC Watch :

1. Add a new connection
2. Copy over the server endpoint URL of the Sylcom software
(e.g. '<opc.tcp://127.0.0.1:4841/SYLCOM>')
(e.g. '<opc.tcp://127.0.0.1:4841/SYLCOM>')
3. Press connection

The screenshot shows the OPC Watch application window. The title bar includes the 'TRAEGE.DE' logo. The interface is divided into several sections:

- Server Browser:** Located on the left, it shows a tree view of the connected server. The root node is '(1) opc.tcp://127.0.0.1:4841/SylCom'. Under it, there are nodes for 'Objects', 'Server', 'Sylcom', 'CommandOnTheFly', 'ProductionInfos', 'ToolCorrectorInfos', 'Types', and 'Views'. The 'Objects' node is expanded, showing a list of objects like 'ClearPresetOnTheFly', 'PresetOnTheFly', 'StoreOnTheFly', 'ConfigurationName', 'GetToolCorrections', etc.
- Configuration Monitor Analyse:** A tabbed interface at the top right. The 'Configuration' tab is active, showing a table of configuration data.
- Table:** The table has columns: Name, Value, Type, Base Type, Status, Server Timestamp, Source Timestamp, and Access. It contains three rows: 'Objects', 'Types', and 'Views'. All 'Type' and 'Access' values are 'Source' and 'None' respectively.
- Status Bar:** At the bottom left, it says 'Connected!'.

Name	Value	Type	Base Type	Status	Server Timestamp	Source Timestamp	Access
Objects		Source					None
Types		Source					None
Views		Source					None

3. METHODS AND NODES

3.1. Methods

Methods	Descriptions
Clear	Clear instrument Min/Max
DisconnectInstrument	Disconnect the instrument <InstrumentID>
GetConfigurationNames	Get the list of the configurations names
GetControlPlanNames	Get the list of control plan names
GetInstrumentsInfos	Get the list of instruments with all infos
GetInstrumentsInfosText	Get the list of instruments with all infos (Text)
LoadConfiguration	Load a configuration <ConfigurationName>
Preset	Preset the page
PresetAllInstruments	Preset all instruments
Set	Set the page
StartControlPlan	Start a control plan <ControlPlan name>
Store	Store the page
StoreMeasurement	Store the channel <json of the measurement>
ToolCorrectorInfos/GetToolCorrections	Get last corrections
ToolCorrectorInfos/GetToolCorrectionsText	Get last corrections (Text)
ToolCorrectorInfos/ResetCorrectionStatus	Reset the correction status
CommandOnTheFly/ClearPresetOnTheFly	Clear the preset on the fly
CommandOnTheFly/PresetOnTheFly	Preset on the fly
CommandOnTheFly/StoreOnTheFly	Store on the fly

3.2. Nodes

Nodes	Descriptions
ProductionInfos/ConfigurationName	Configuration Name
ToolCorrectorInfos/CorrectionDate	Correction date
ToolCorrectorInfos/CorrectionMachineName	Correction Machine Name
ToolCorrectorInfos/CorrectionStatus	Correction Status

3.3 Final note

The OPC foundation provided several powerful clients that can be used to test the interoperability of the system. They are a little more complex to be used, but are better to explore the properties (inputs and outputs parameters) of the callable methods.

These clients can be downloaded at the following address (required a free registration):

<https://opcfoundation.org/developer-tools/developer-kits-unified-architecture>





Changes without prior notice
Sous réserve de toute modification
Änderungen vorbehalten

Edition :

2022.11