Learn-/Training Document

Siemens Automation Cooperates with Education (SCE) | As of Version V9 SP1

PA Module P03-02

SIMATIC PCS 7 – Vertical integration with OPC

siemens.com/sce

SIEMENS

Global Industry Partner of WorldSkills International

worldskills

Matching SCE Trainer Packages for this Learn-/Training Document

- **SIMATIC PCS 7 Software Package V9.0 (set of 3)**
  Order No.: 6ES7650-0XX58-0YS5
- **SIMATIC PCS 7 Software Package V9.0 (set of 6)**
  Order No.: 6ES7650-0XX58-2YS5
- **SIMATIC PCS 7 Software Upgrade Packages (set of 3)**
  Order No.: 6ES7650-0XX58-0YE5 (V8.x→ V9.0)
- **SIMIT Simulation Platform with Dongle V10**
  (contains SIMIT S & CTE, FLOWNET, CONTEC libraries) – 2500 simulation tags
  Order No.: 6DL8913-0AK00-0AS5
- **Upgrade SIMIT Simulation Platform V10**
  (contains SIMIT S & CTE, FLOWNET, CONTEC libraries) from V8.x/V9.x
  Order No.: 6DL8913-0AK00-0AS6
- **Demo Version SIMIT Simulation Platform V10**
  Download
- **SIMATIC PCS 7 AS RTX Box (PROFIBUS) only in combination with ET 200M for RTX –**
  Order No.: 6ES7654-0UE23-0XS1
- **ET 200M for RTX Box (PROFIBUS) only in combination with PCS 7 AS RTX Box –**
  Order No.: 6ES7153-2BA10-4AB1

Note that these trainer packages are replaced with successor packages when necessary.
An overview of the currently available SCE packages is available at: siemens.com/sce/tp

Continued training

For regional Siemens SCE continued training, get in touch with your regional SCE contact
siemens.com/sce/contact

Additional information regarding SCE

siemens.com/sce

Information regarding use

The SCE Learn-/Training Document for the integrated automation solution Totally Integrated Automation (TIA) was prepared for the program "Siemens Automation Cooperates with Education (SCE)" specifically for training purposes for public educational facilities and R&D institutions. Siemens does not guarantee the contents.

This document is to be used only for initial training on Siemens products/systems, which means it can be copied in whole or part and given to those being trained for use within the scope of their training. Circulation or copying this Learn-/Training Document and sharing its content is permitted within public training and advanced training facilities for training purposes.

Exceptions require written consent from the Siemens. Send all related requests to scesupportfinder.i-ia@siemens.com.

Offenders will be held liable. All rights including translation are reserved, particularly if a patent is granted or a utility model or design is registered.

Use for industrial customer courses is explicitly not permitted. We do not consent to commercial use of the Learn-/Training Document.

We wish to thank the TU Dresden, particularly Prof. Dr.-Ing. Leon Urbas and the Michael Dziallas Engineering Corporation and all other involved persons for their support during the preparation of this Learn-/Training Document.

Table of contents

1 Goal....5
2 Prerequisite....5
3 Required hardware and software....6
4 Theory....7
4.1 Theory in brief....7
4.2 Introduction....8
4.3 Client-Server principle....9
4.4 OPC Specification....10
4.5 OPC server of SIMATIC NET....14
4.6 OPEN PCS 7....15
4.7 References....15
5 Task....16
6 Planning....16
7 Learning objective....16
8 Structured step-by-step instructions....17
8.1 Starting simulation and WinCC Runtime....17
8.2 Tag access with OPC Scout....19
8.3 Configuration of Excel for tag access with VBS/ActiveX....22
8.4 Tag access in Excel....29
8.5 Checklist – step-by-step instruction....31
9 Exercises....32
9.1 Tasks....32
9.2 Checklist – exercise....32
10 Additional information....33

Vertical integration with OPC

1 Goal

In this chapter, the students learn to integrate automation systems of a wide range of manufacturers in higher level programs of the plant control level. The necessary basic information on the design and method of functioning of OPC and the possibilities of integration using PCS 7 are explained.

2 Prerequisite

This chapter builds on chapter 'Functional safety'. To implement this chapter, you can use an existing project from the previous chapter or the archived project 'p03-01-exercise-r1905-en.zip' provided by SCE.

Also needed is the file 'p03-02-opc-template-r1905-en.xls' (contained in zip file 'p03-02-files-r1905-en.zip'), which is also provided by SCE.

The (optional) simulation for the SIMIT program can be retrieved from the file 'p01-04-plantsim-v10-r1905-en.simarc'. It can be run in demo mode.

The download of the projects/files is stored as a zip file on the SCE Internet for the respective module.

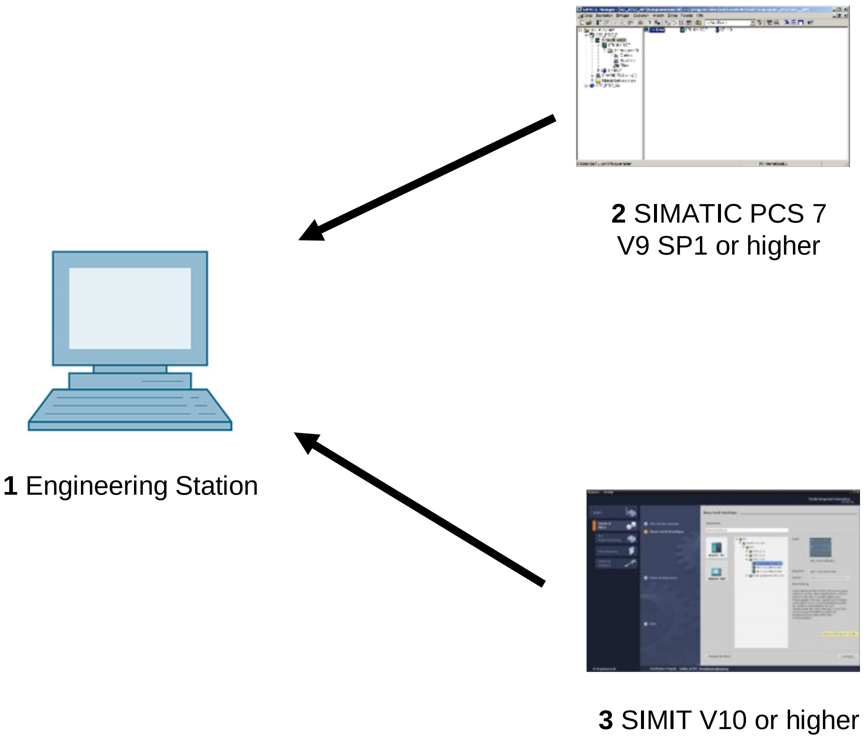
3 Required hardware and software

1 Engineering station: Requirements include hardware and operating system (for further information, see Readme on the PCS 7 installation DVD)

2 SIMATIC PCS 7 software V9 SP1 or higher

- Installed program packages (contained in SIMATIC PCS 7 Software Trainer Package):
    - *Engineering → PCS 7 Engineering*
    - *Engineering → BATCH Engineering*
    - *Runtime → Single Station → OS Single Station*
    - *Runtime → Single Station → BATCH Single Station*
    - *Options → SIMATIC Logon*
    - *Options → S7-PLCSIM V5.4 SP8*

3 Demo Version SIMIT Simulation Platform V10



4 Theory

4.1 Theory in brief

OLE for Process Control (OPC) provides a standard mechanism for communication with a variety of data sources. It does not matter whether these sources are machines in your factory or a database in your control room. OPC is based on the OLE/COM technology from Microsoft.

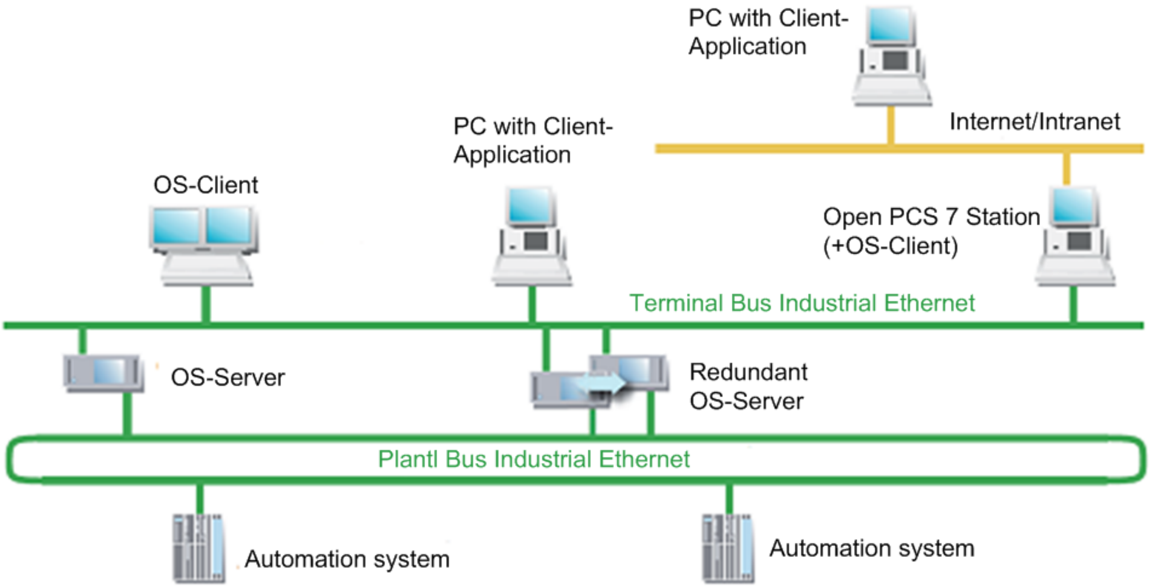


Figure 1: Interfacing of automation systems with the IT world [3]

By means of an OPC server, data can be exchanged with external systems without knowledge of the topology and a PCS 7 OS installation being required.

From the view of the user programs generated in higher level programming languages such as C++ and Visual Basic, OPC is a bridge to the process data and the device data of the automation systems. For device manufacturers, the development of an OPC server is necessary instead of special drivers. For software developers, the advantage is that device-independent applications can be written. Users in turn have more freedom regarding the selection of devices and software products [1].

4.2 Introduction

OPC provides a standardized, open and manufacturer-independent software interface for integrated data communication between automation systems and MS Windows applications with OPC capability. OPC has developed into a de-facto standard for interfacing automation systems by different manufacturers with higher level programs of the plant control level for the following:

– Process visualization (monitoring of individual product lines with direct data exchange)

– Integrated plant management (ordering, quality control, maintenance, material management, production planning)

The software interface OPC is based on the Windows technologies COM (Component Object Model) and DCOM (Distributed Component Object Model). OPC XML on the other hand is based on the Internet standards XML, SOAP and http. COM is the Microsoft protocol standard for communication between objects that are located on one computer but in different programs. With DCOM, COM was expanded with the capability to access objects beyond computer boundaries. This basis allows a standardized data exchange between applications from industry, office and manufacturing. Communication via DCOM is limited to local networks. Data exchange via XML uses the SOAP protocol (Simple Object Access Protocol). SOAP is an XML based protocol that is independent of the platform. With SOAP, applications on the Internet or in heterogeneous computer networks can communicate with each other via HTTP (HyperText Transfer Protocol).

The OPC Foundation defined the standard for the OPC interface. Leading corporations in industry automation comprise the OPC Foundation. The OPC servers of the OS system support the following specifications:

– OPC Data Access 1.0, 2.05a and 3.0

– OPC XML Data Access 1.01

– OPC Historical Data Access 1.20

– OPC Alarm & Events 1.10

4.3 Client-Server principle

OPC communication is based on the client-server principle (refer to Figure 2). The client (customer) takes the initiative and makes a request to the server (service provider). The server replies, executes or supplies. The connection establishment needed for this always emanates from the OPC client. The advantage of such a communication scheme is that only the clients have to "know" the server.

The OPC server can access the process data of the automation system.

OPC server

An OPC software component that provides data when an OPC client requests it is called an ***OPC server***. A server must be installed on the PC since OPC-specific entries are required in the Windows registry. "Upwards", the OPC server supports the interface specification Data Access; "downwards" it is connected through a lower level communication network to the connected automation system as the actual data source.

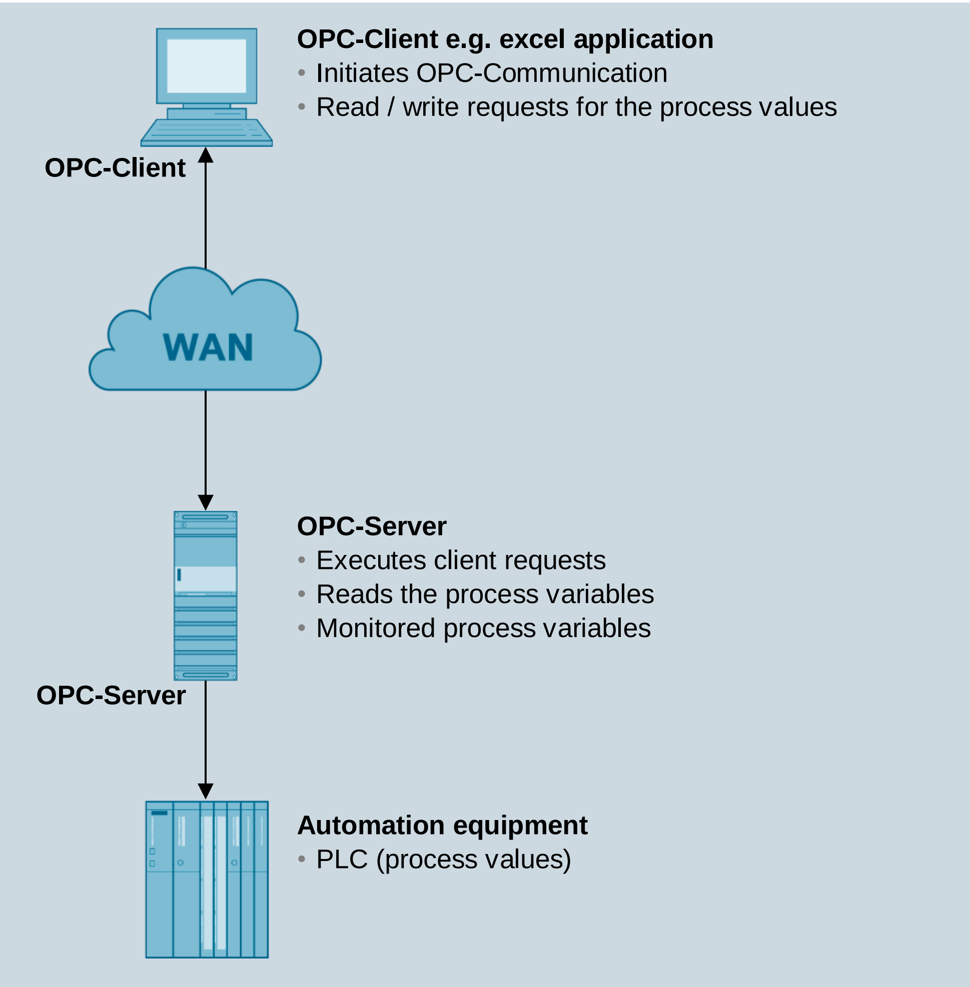


Figure 2: Client-Server principle

OPC client

OPC components that use an OPC server as a data source are called ***OPC clients***. An OPC client is, as a rule, part of the user program that has to be configured. Two OPC interfaces are available:

– The ***custom interface*** (customer specific interface) for programming languages that address interfaces with the function pointer principle, such as C/C++.

– The ***automation interface*** for programming languages that address interfaces with object names, such as Visual Basic.

4.4 OPC Specification

OPC Data Access (OPC DA)

Data Access is an OPC specification for access to process data by means of variables. An OPC server for Data Access manages the process tags and the various access possibilities to these variables. As a result, the OPC server can:

– Read the value of one or more process tags

– Change the value of one or more process tags by writing a new value

– Monitor the value of one or more process tags

– Signal value changes

Process tags are placeholders for values that must be determined at the moment. The OPC specification defines the interface between client and server programs for managing the process data. Data Access servers allow transparent access to a wide range of data sources (for example, temperature sensor) and data sinks (for example, control systems) for one or more Data Access clients.

These data sources and sinks can be located on I/O cards inserted directly in the PC. But they can also be on any device such as controllers, input/output modules, etc., that are connected via serial connections or fieldbuses. A Data Access client can also access several Data Access servers at the same time.

Data Access clients can be very simple Excel sheets or extensive programs (for example, Visual Basic). Data Access clients in turn can be part of larger programs.

OPC Data Access class model

The hierarchical class model of Data Access assists in adapting the time expenditure for and the resulting content of the data access by the client to the current requirements of the application. Data Access differentiates three classes:

– OPC server

– OPC group

– OPC item

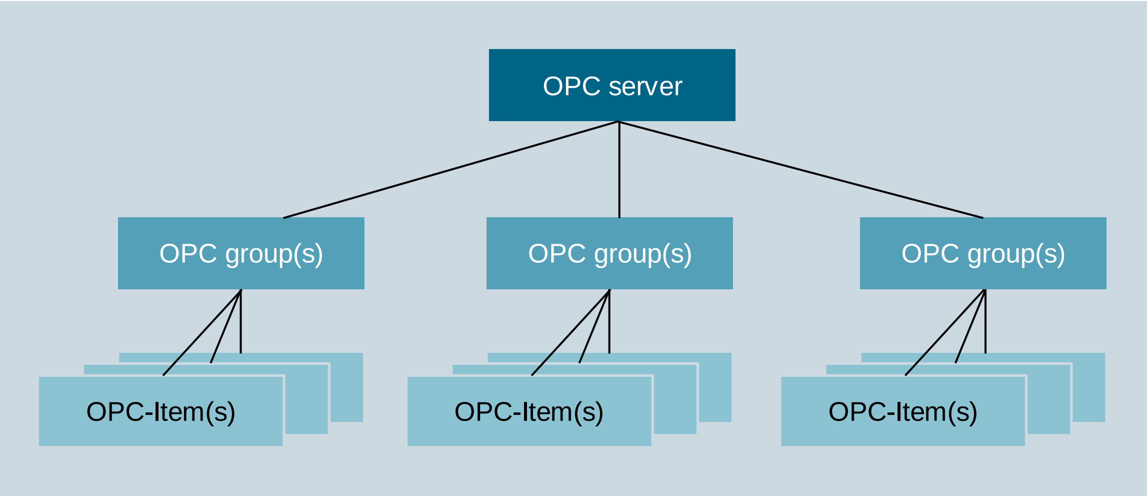


Figure 3: Class model of the Data Access interface [2]

The client application uses COM calls of the operating system only to generate an object of the OPC server class. The other objects are generated through corresponding OPC methods of the OPC server class or lower-level classes.

The ***OPC server*** class is at the top. Every OPC server belongs to this class. This class represents the access for all other services of the Data Access server. With the aid of class-specific attributes, information about the status, version and (optionally) name space of the available process tags can be obtained. Objects of the OPC server class manage the instances of the lower-level OPC group class.

The ***OPC group*** class is directly subordinate to the OPC server class and structures the process tags utilized by the OPC server. An OPC client can use several objects of this class simultaneously. With the aid of the objects of the OPC group class, a client can generate useful units of process tags and perform operations with them. For example, all process tags of a screen page of an operator control and monitoring system can be combined into one group.

The OPC group class defines methods that can be used to read and write the values of the process tags.

The ***OPC item*** class represents the actual process tag and allows targeted polling of individual data. Each variable is an element (item) in the namespace of the OPC server and is identified with an item ID. The item ID is specified by the manufacturer of the server and must be unique within the server's namespace. Each value has the following properties:

– ***Value***

  The last recorded value of the variable

– ***Quality***

  Significance of the value. If the quality is good, the value was able to be determined with certainty.

– ***Time stamp***

  Point in time when the current value of the variable was determined. With each value change indicated to the client, the time stamp is updated. If the value of the variable does not change, the time stamp remains the same also.

Variables must be specified when the OPC interface is called in order to obtain process values. By specifying variables, the client is able to request the needed values from the server. The client must register each desired variable with the server to specify which variables are to be read. Variables can be read and written synchronously as well as asynchronously.

The client can transfer the monitoring of variables to the server. If the value of a variable changes, the server sends a corresponding message to the client. The variables provided by the OPC server can be subdivided into:

– ***Process tags***

  Represent measured values and control values of input/output devices

  or

– ***Control variables***

  The use of these variables triggers certain additional services, e.g. the transfer of passwords

  or

– ***Information variables***

  These variables are made available by the communication system and by the OPC server and provide information about the status of connections, devices, etc.

Some examples of the variables of an OPC Data Access server:

– Control values of a PLC

– Data of a measurement acquisition system

– Status variables of the communication system

OPC eXtensible Markup Language DA (OPC XML DA)

OPC XML is a standard that allows communication with a platform-independent protocol over the Internet. The client is no longer set to a Windows environment (DCOM). With the HTTP protocol and the SOAP interface, other operating system, such as LINUX, can monitor and exchange OPC data via the Internet.

SOAP provides a simple and transparent mechanism for exchanging structured and type-defined information between computers in a decentralized, distributed environment. SOAP is a basis for XML-based information exchange.

Data access by means of OPC XML has a functional scope based on OPC Data Access. Change-driven feedback regarding data changes as in the case of DCOM interfaces is not provided for OPC XML because of the non-fixed Internet connection.

Accessing methods directly from the Internet represents a considerable security risk. For that reason, SOAP exclusively uses the Internet http channel (HTTP = Hyper Text Transfer Protocol) for the data transfer, which can be easily administered through a firewall.

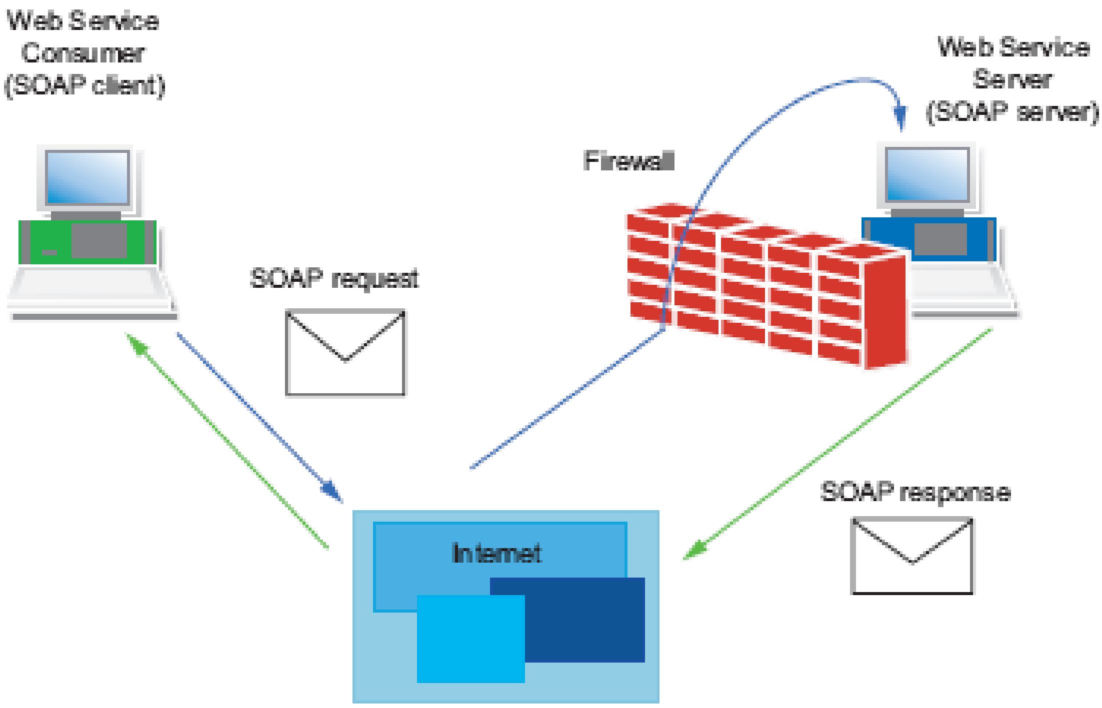


Figure 4: Data transfer with the HTTP protocol [2]

OPC Alarms & Events (OPC A&E)

OPC Alarm & Events is an additional specification for transferring process alarms and events. Events are special states in the process that have to be signaled to a recipient. The OPC specification defines three types of events for this:

– Condition-related events

They signal the state transitions defined in the OPC state model and are tied to defined conditions.

– Tracking events

  They signal process changes; for example, when a user changes the setpoint of a controller

– Simple events

  They signal all other stateless events; for example, the failure of a system component.

Alarms&Events servers are used, for example, for

– Detecting events – for example, reactor level reached.
– Determining the state of an event - product tank full
– Confirming an event – reaching of reactor level detected
– Monitoring the confirmation – the confirmation is monitored by the reactor alarm signaler, the alarm was detected, the warning signal can be switched off

There is the option to signal new events even without a confirmation. The events signaled to the OPC client are set by the OPC client using filter criteria. All events corresponding to the set filter criteria must be forwarded from the generator of the event up to the user. This distinguishes Alarms & Events from Data Access. When variables are monitored, only the value changes within the specified time grid are communicated.

The OPC specification defines the syntax of the interface for message receipt. The manufacturer of the OPC server specifies the event types that a server provides.

4.5 OPC server of SIMATIC NET

The open OPC interface is the central interface of the products on the PG/PC of SIMATIC NET. The OPC server of SIMATIC NET supports all communication protocols and services that the communication modules make available.

The OPC server of SIMATIC NET supports the interface specification OPC Data Access for all protocols. For protocols that have mechanisms for communication of events (S7 communication), OPC Alarms & Events is supported also.

The OPC server of SIMATIC NET allows access to the industrial communication networks PROFIBUS and Industrial Ethernet of SIMATIC NET. It provides OPC clients the values of process tags or signals events of the partner device. For this, it uses the protocol software and the communications processor of SIMATIC NET to access the partner devices via the communication network (see Figure 5).

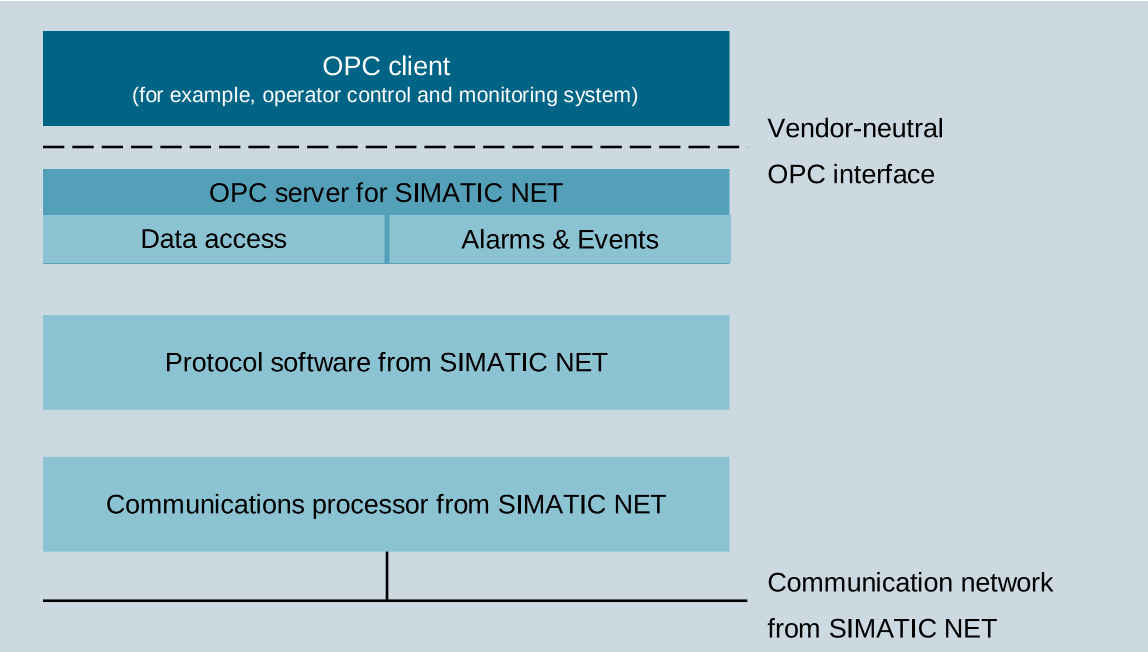


Figure 5: OPC server for SIMATIC NET with OPC client [2]

4.6 OPEN PCS 7

OpenPCS 7 was developed for use on a multi-user system. With a separate PC station with OpenPCS 7 (OpenPCS 7 station), data can be exchanged with external systems without knowledge of the topology and without a PCS 7 OS installation being necessary. This server collects the data from existing OS server systems via OPC and makes it available in a concentrated form. For that reason, OpenPCS 7 can be used exclusively on multi-user systems.

OpenPCS 7 can be used for data exchange with the following levels:

– Automation level

– Plant control and production control levels

– MES level (Manufacturing Execution Systems)

– ERP level (Enterprise Resource Planning)

4.7 References

[1] Wellenreuther, G.; Zastrow, D. (2009) Automatisieren mit SPS (4. Auflage). Vieweg + Teubner.

[2] SIEMENS (2010): SIMATIC NET: Industrial Communication with PG/PC Volume 1 - Basics. C79000-G8900-C172-09. (support.automation.siemens.com/WW/view/en/42783968)

[3] SIEMENS (2017-12): SIMATIC Process Control System PCS 7 Engineering System (V9.0 SP1). A5E39221271-AC. (support.automation.siemens.com/WW/view/en/109754984)

5 Task

The example below shows how to access tags in WinCC from Microsoft Excel using OPC. In these step-by-step instructions, you will select the level of Reactor R001 to access them.

The settings necessary for this are made in an Excel file. The required macros already exist in the Excel file and are started automatically with the correct settings.

Detailed knowledge of macro-programming is not required in this module nor is it imparted. Refer to the manuals and technical references on Microsoft Excel.

To diagnose the availability of the OPC server of WinCC, the software OPC Scout V10 of SIMATIC NET is used.

6 Planning

All requirements for access via OPC are already met. All tags can be viewed and manipulated via OPC Scout.

For access via Excel, only the ActiveX macro has to be activated in Excel and, if necessary, the driver 'Siemens OPC DAAutomation 2.0' must be added in the references.

- The following tags are to be read-accessed:

  A1_multipurpose_plant/T2_reaction/reactor$R001/A1T2L001/monitor_A1T2L001.PV#Value

- The following tags are to be write-accessed:

  A1_multipurpose_plant/T2_reaction/reactor$R001/A1T2L001/monitor_A1T2L001.PV_AH_Lim

7 Learning objective

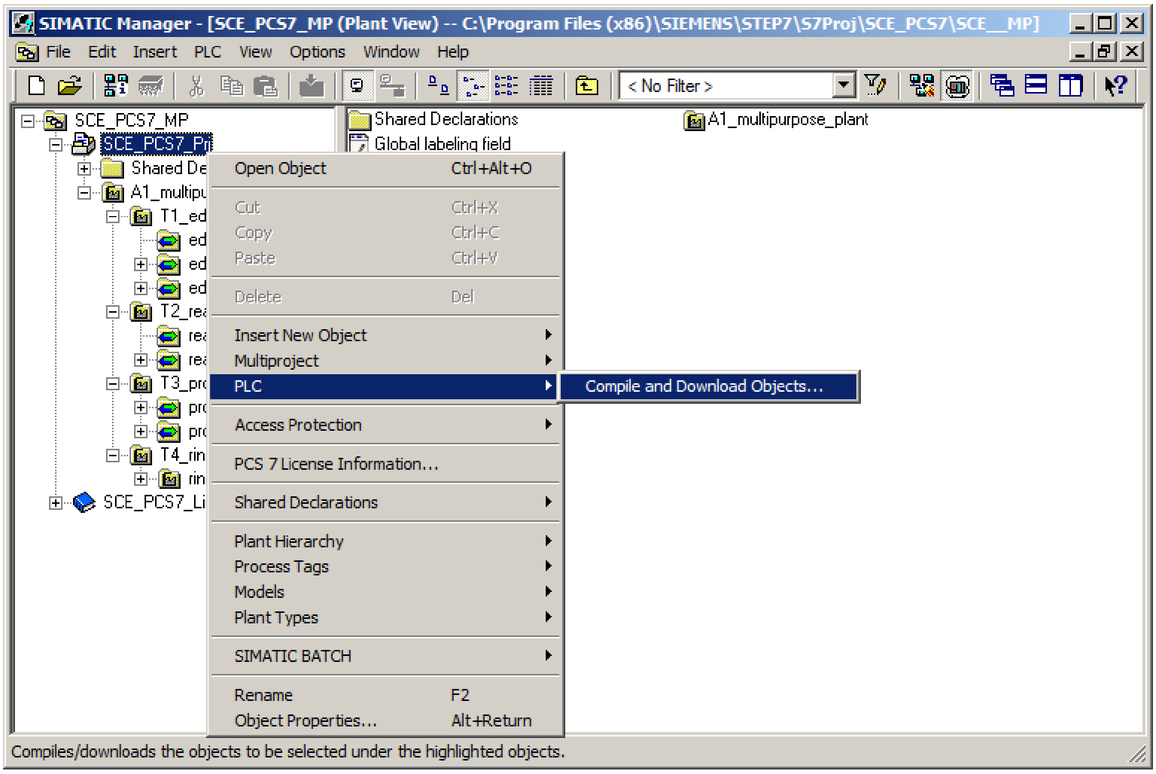
In this chapter, students learn the following:

- Diagnosis of the local OPC servers with OPC Scout V10
- Display of tags available via OPC in OPC Scout V10
- Settings in Excel for OPC access to Tag Management of WinCC
- Testing of a Microsoft Excel application with access to tags of WinCC via OPC

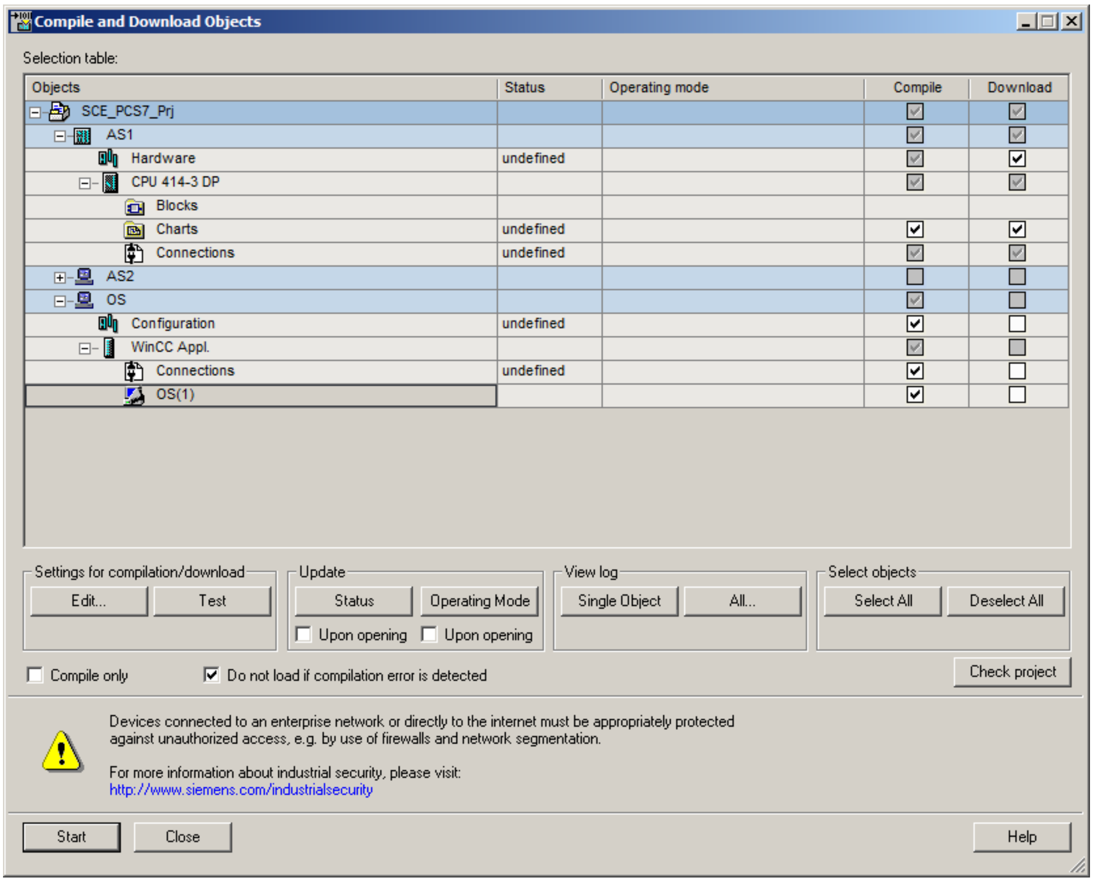
8 Structured step-by-step instructions

8.1 Starting simulation and WinCC Runtime

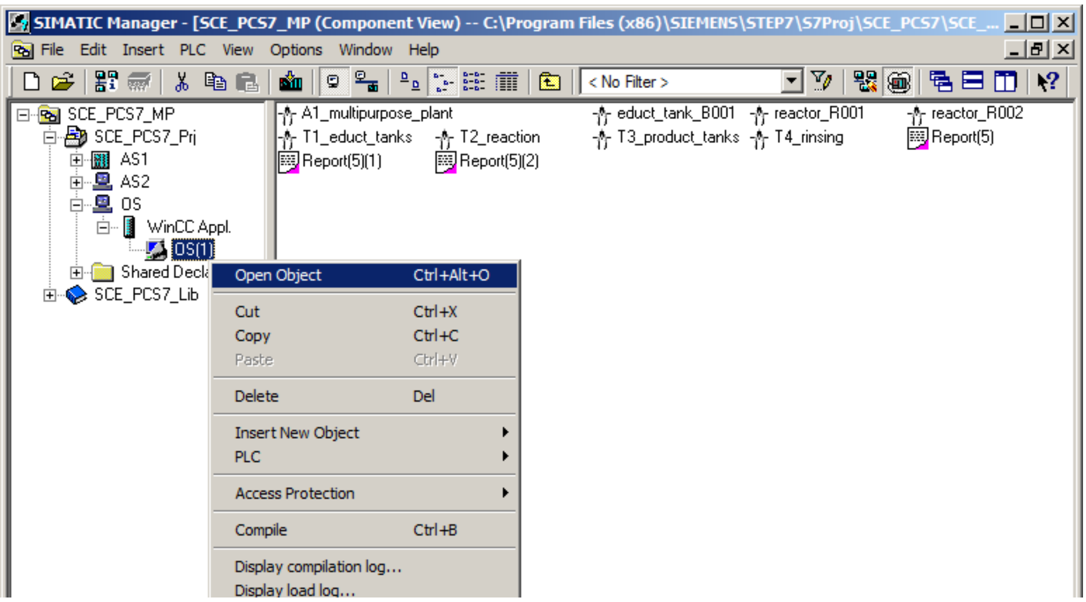
1. After opening the exercise project from chapter "Advanced Layouts for UIs P03-01", start PLCSIM, and initiate the compilation and download of objects of the project in the plant view.
(→ SCE_PCS7_Prj → PLC → Compile and Download Objects)



2. Before starting the compilation with 'Start', make sure that S7-PLCSIM is in STOP mode. For the charts, compile and download everything. For the OS, compile without 'Memory reset of OS'. (→ Start)



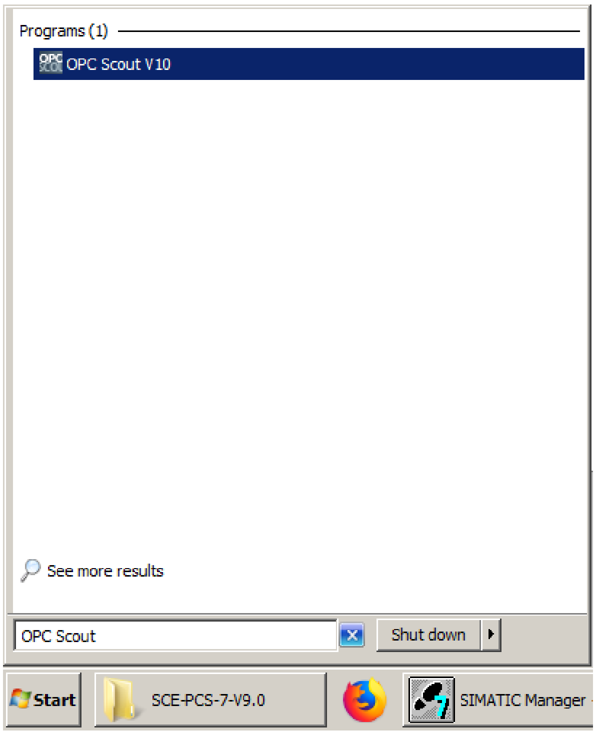
3. Next, set S7-PLCSIM to 'RUN' mode and open WinCC. (→ SIMATIC PC-Station(1) → WinCC Appl. → OS(1) → Open Object)



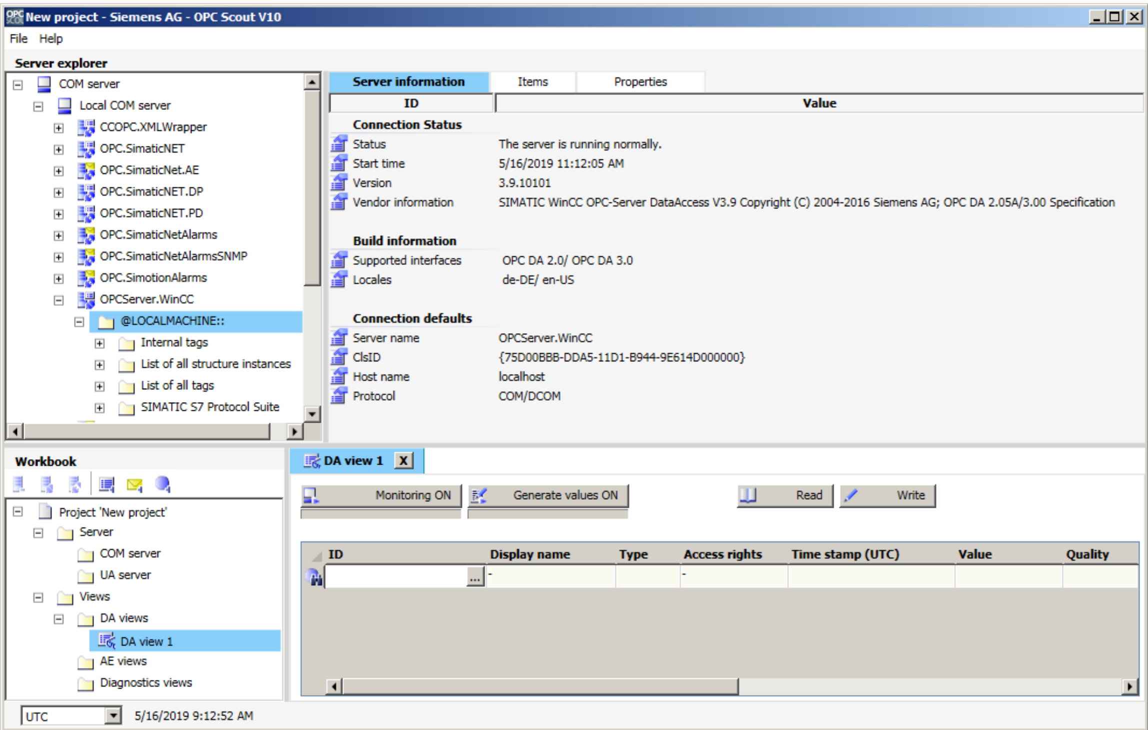
4. Then, activate Runtime in WinCC. (→  Activate)

8.2 Tag access with OPC Scout

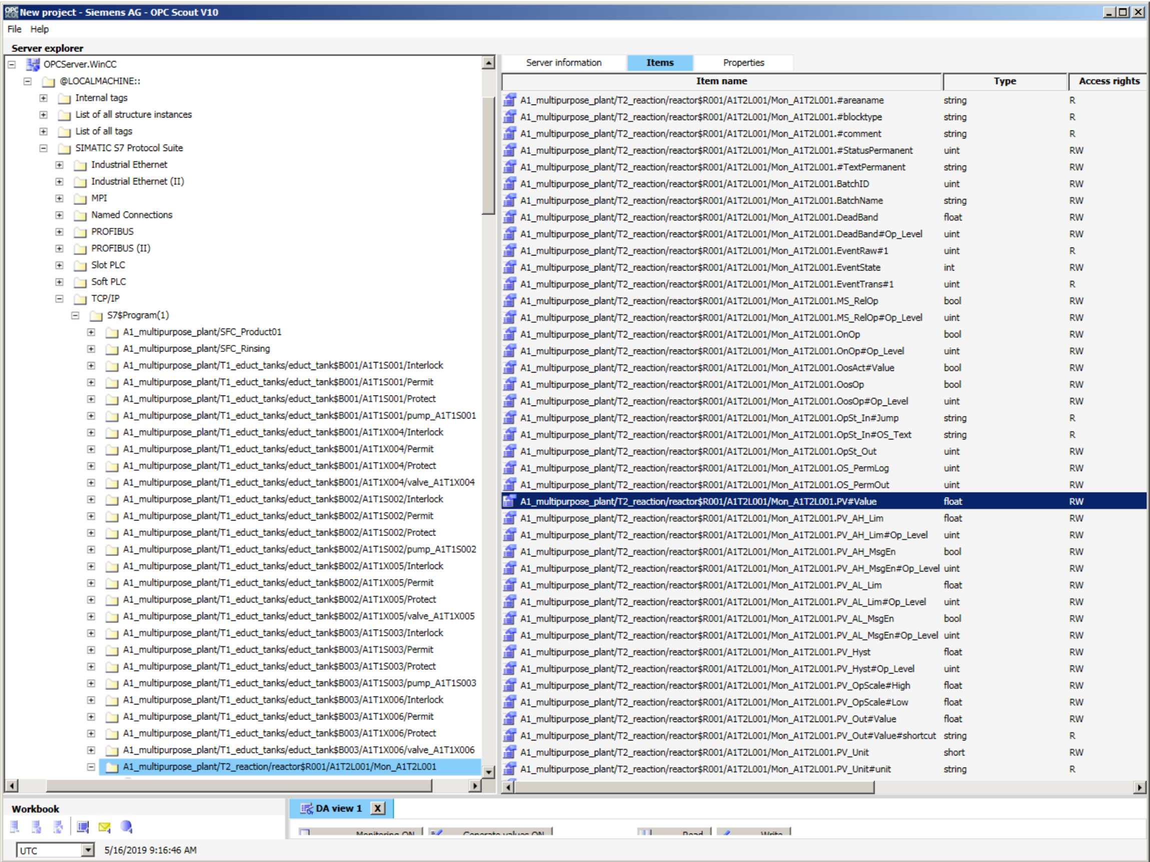
1. As soon as Runtime has started up completely, start 'OPC Scout V10'. (→ Start → Siemens Automation → SIMATIC → SIMATIC NET → OPC Scout V10)



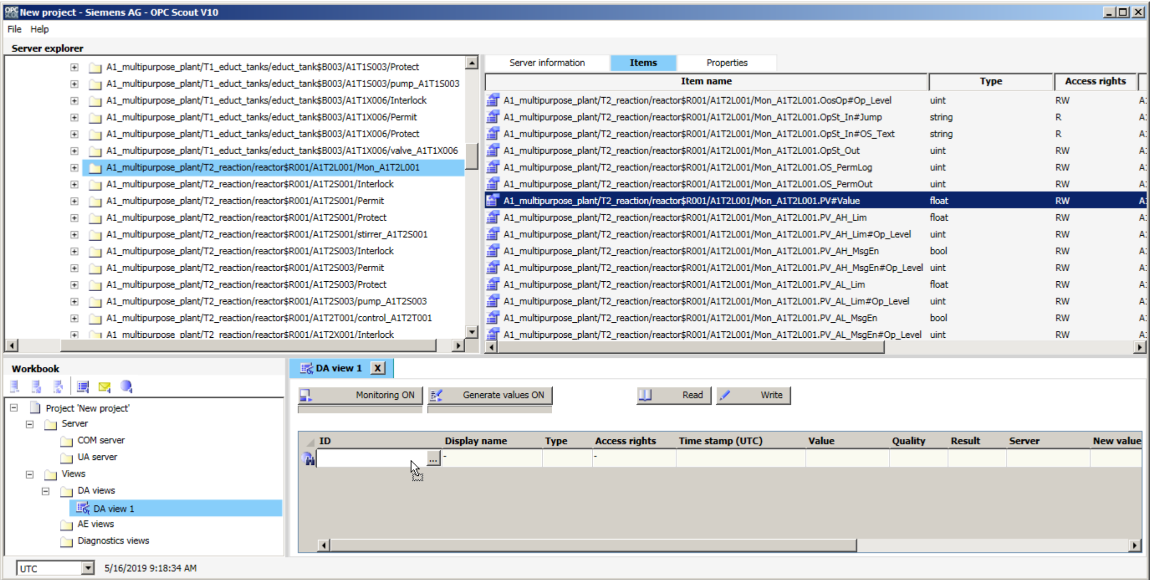
2. Diagnostic data for the various OPC servers of the local PC can be seen in OPC Scout V10. If it is not possible to establish a connection to a certain server, this will be displayed as an error.



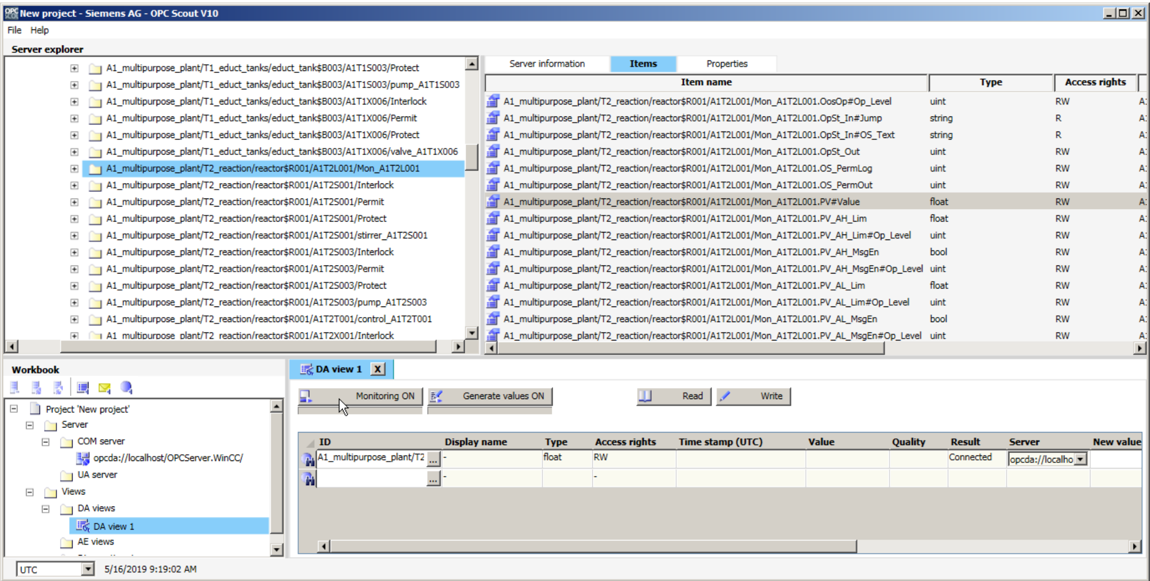
3. The WinCC server is called 'OPCServer.WinCC'. Below this server, a folder structure can be expanded all the way down to the CFC blocks that can be monitored. In this example, you have selected the 'Monitor_A1T2L001' block for displaying the level of Reactor R001. (→ OPCServer.WinCC → @LOCALMACHINE → SIMATIC S7 Protocol Suite → TCP/IP → S7 Program → A1_multipurpose_plant/T2_Reaction/ reactor$R001/A1T2L001/Mon_A1T2L001)



4. From the data of this monitoring block, you want to monitor the 'PV#Value' input. Therefore, drag it into the lower area of the 'DA view 1'. (→ A1_multipurpose_plant/T2_reaction /reactor$R001/A1T2L001/Mon_A1T2L001.PV#Value)



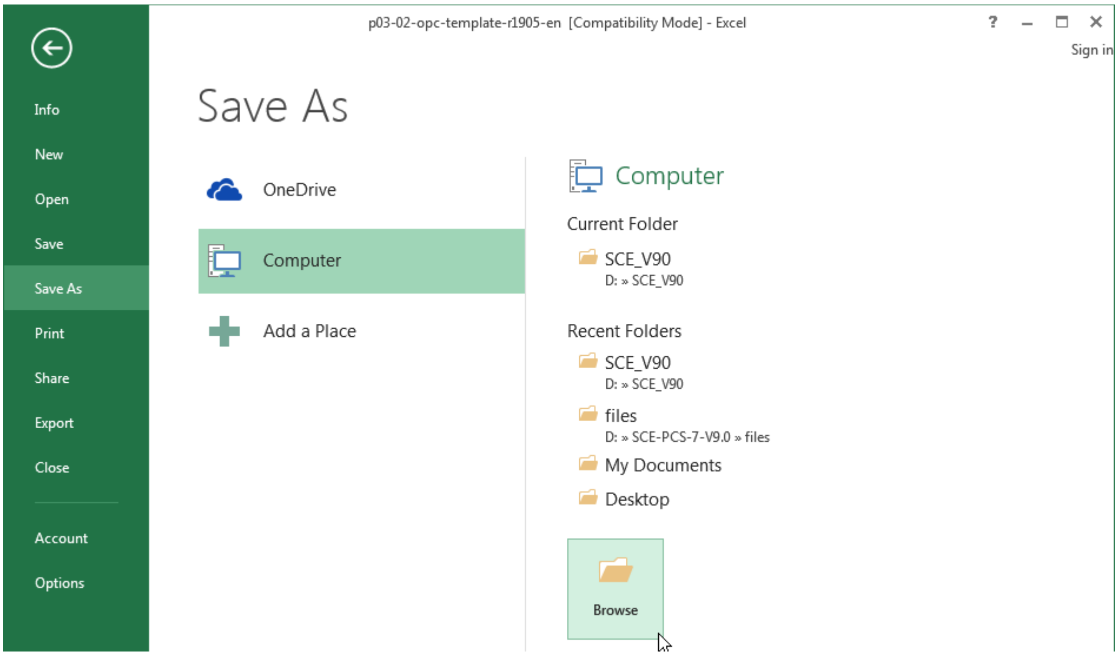
5. Next, click on 'Monitoring ON'. Now, in addition to other information, you can monitor the value, time stamp and quality of this tag. This shows you whether 'OPCServer.WinCC' is working properly. (→ Monitoring ON)

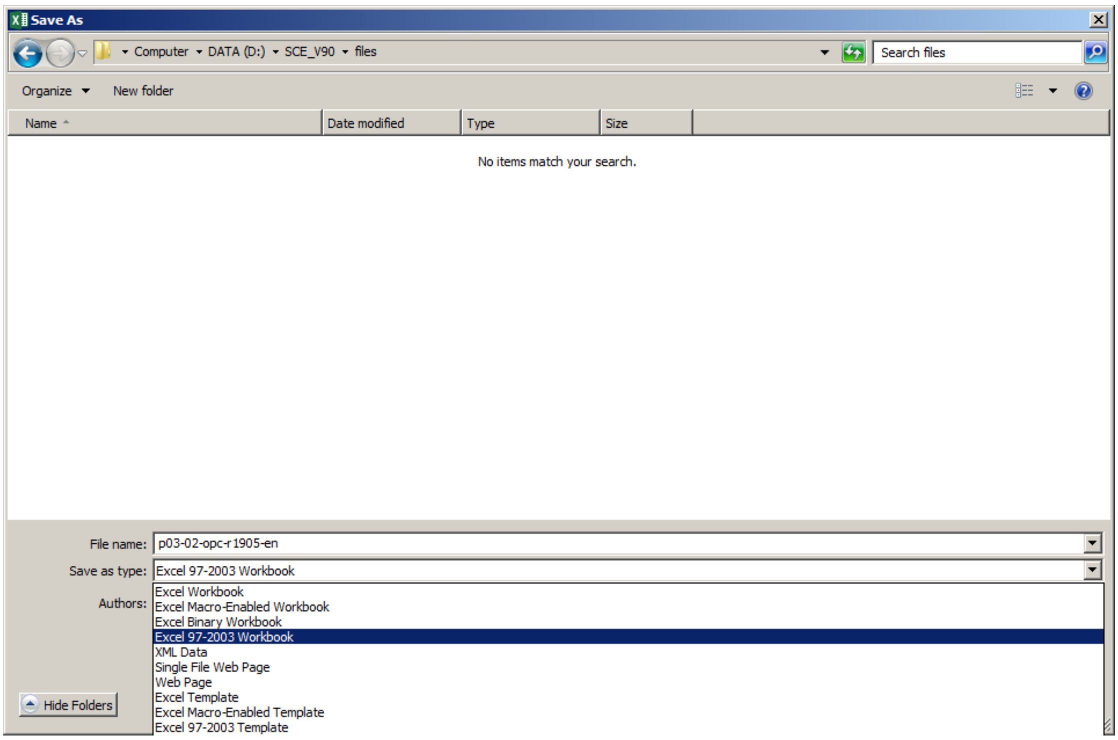


8.3 Configuration of Excel for tag access with VBS/ActiveX

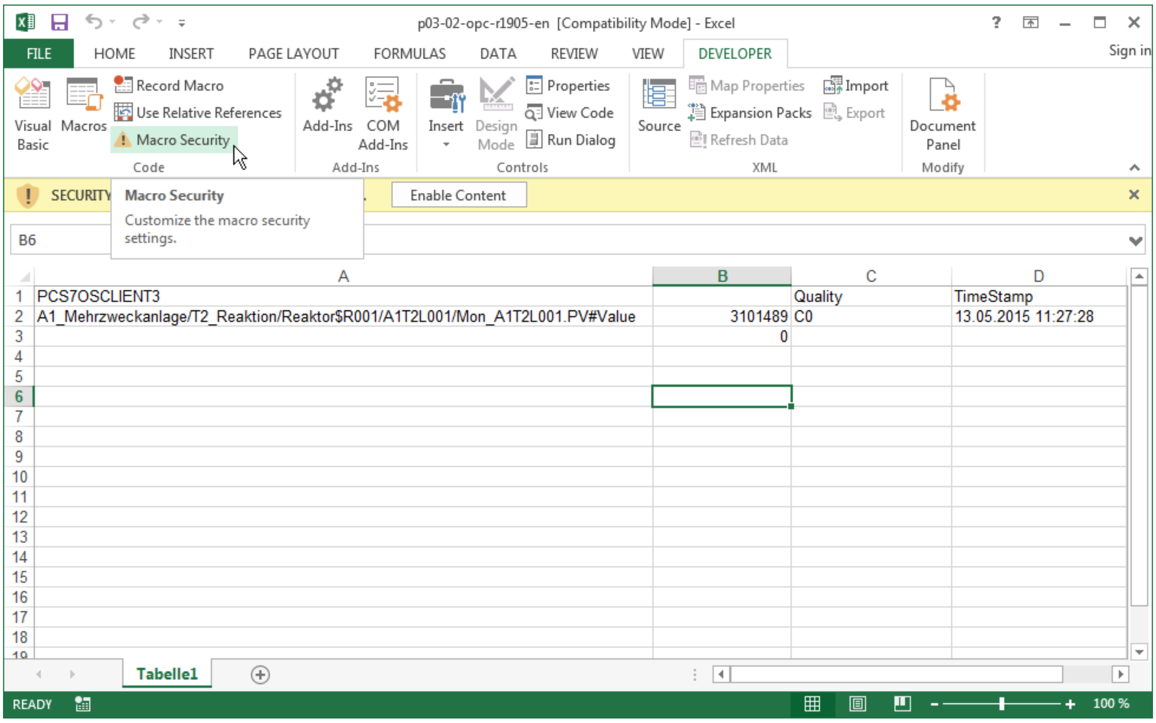
1. If this is the case, you can open the Microsoft Excel file supplied with the module with a double-click. (→ p03-02-opc-template-r1905-en.xls)

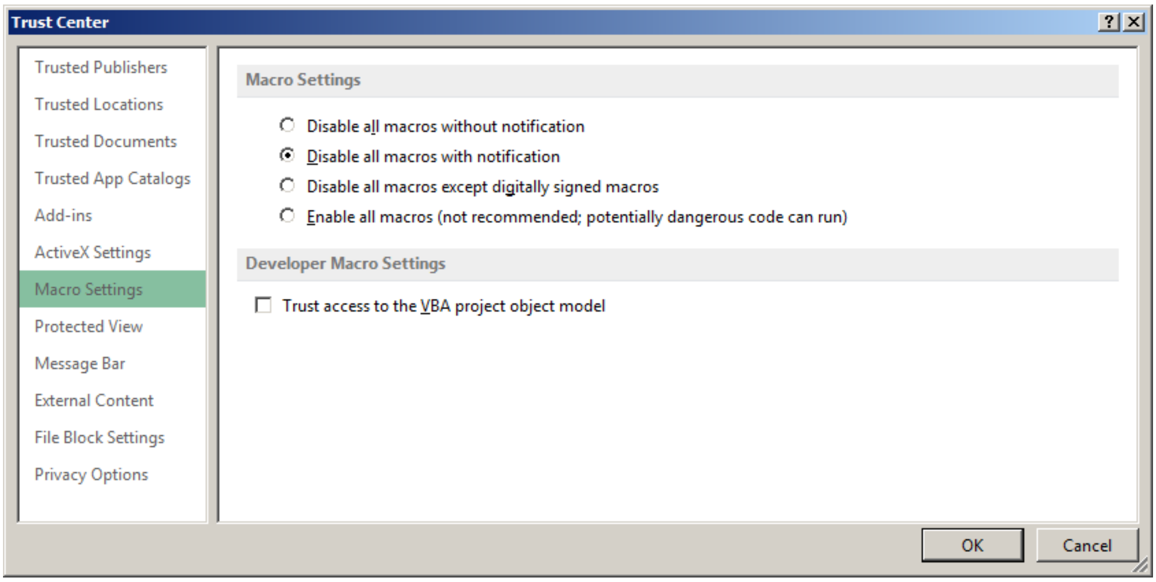
2. Then, save it in Microsoft Excel under a new name. (→ File → Save As → p03-02-opc-r1905-en.xls)



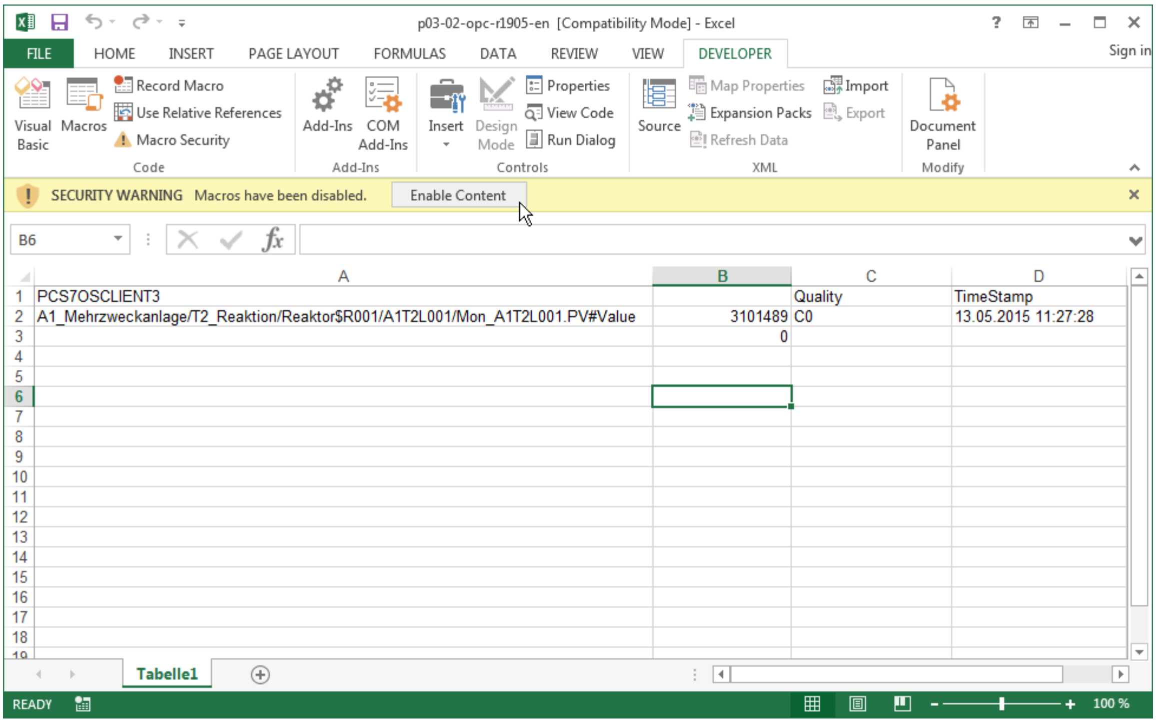


3. In Microsoft Excel, the correct security settings must be made so that the macros can be started (→ Developer tools → Macro Security → Macro Settings → Disable all macros with notification → OK)

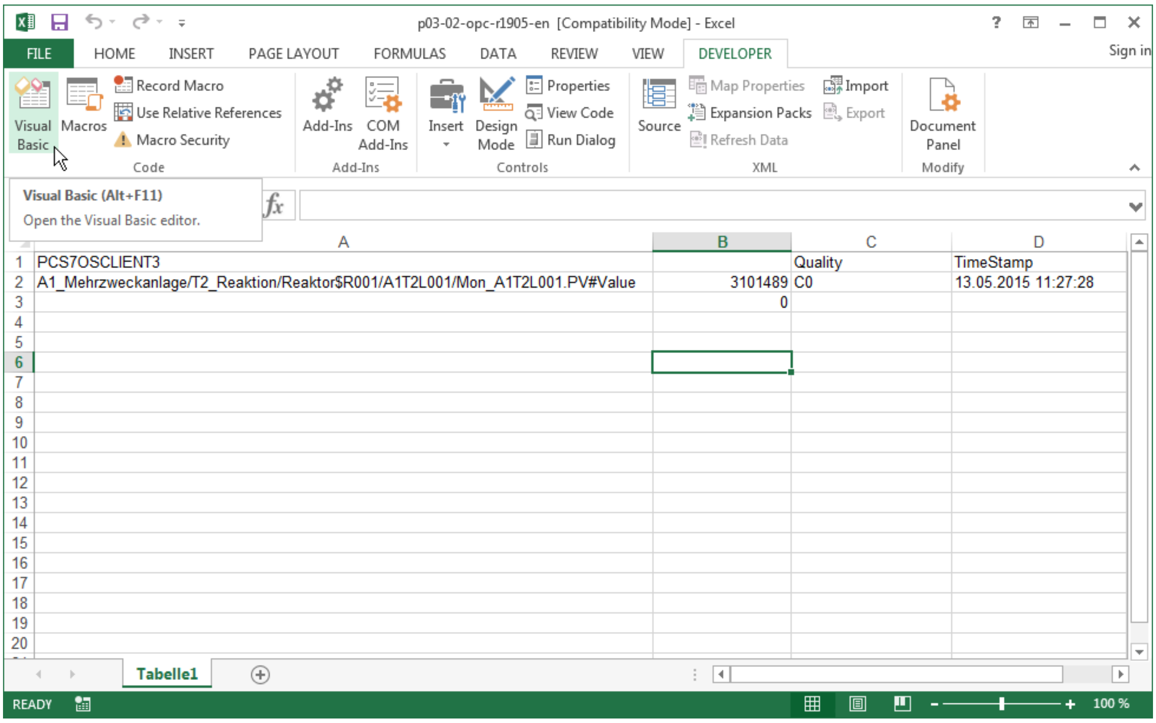




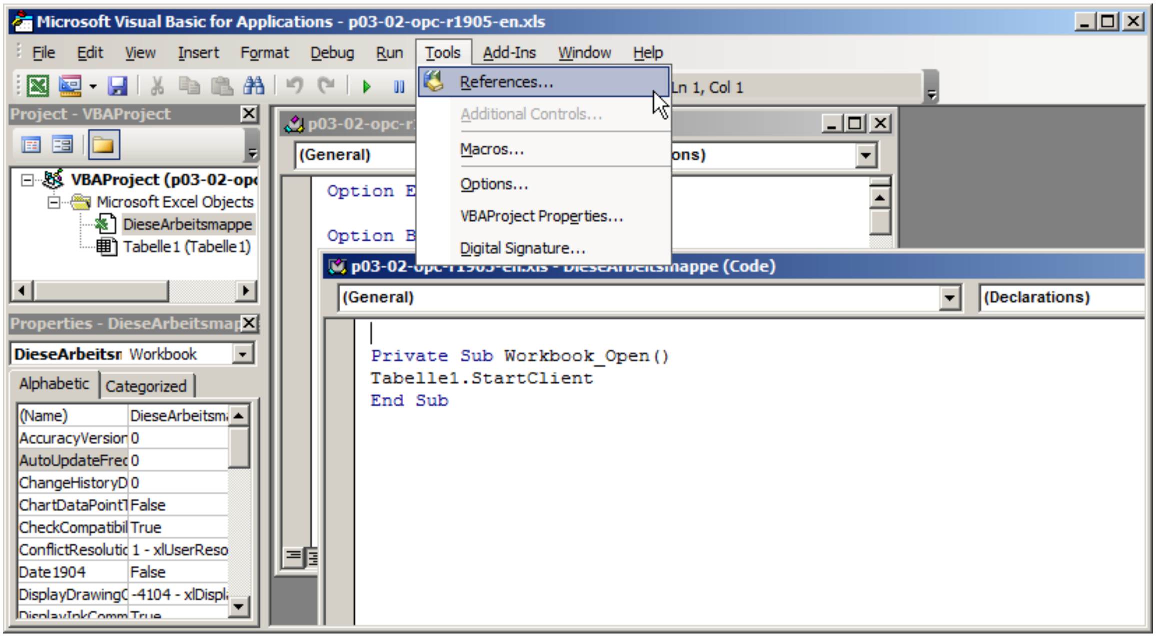
4. Then, enable the macros for this session in the security warning. (→ Enable Content)



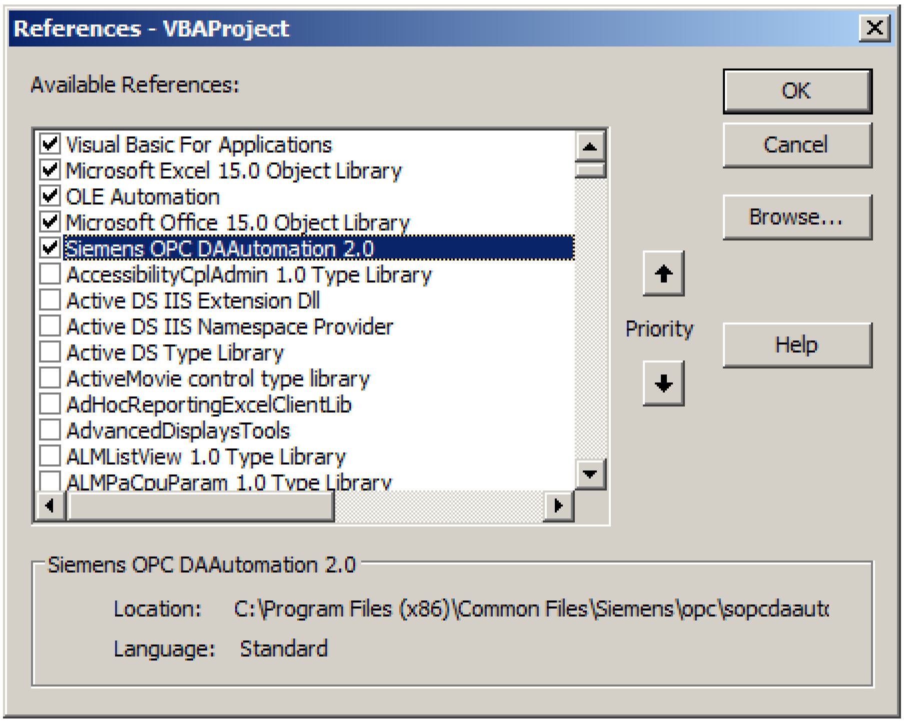
5. For the remaining settings, open 'Visual Basic' in Microsoft Excel. (→ Developer tools → Visual Basic)



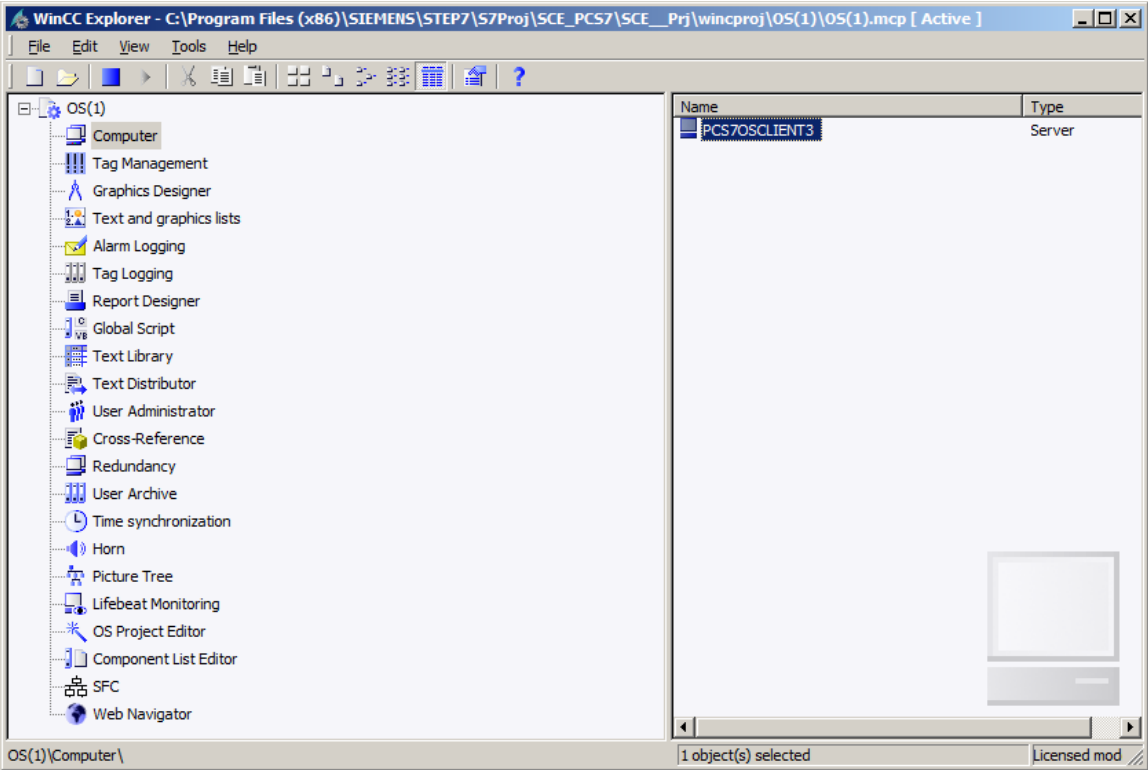
6. There, under 'Tools' open 'References'. (→ Tools → References)

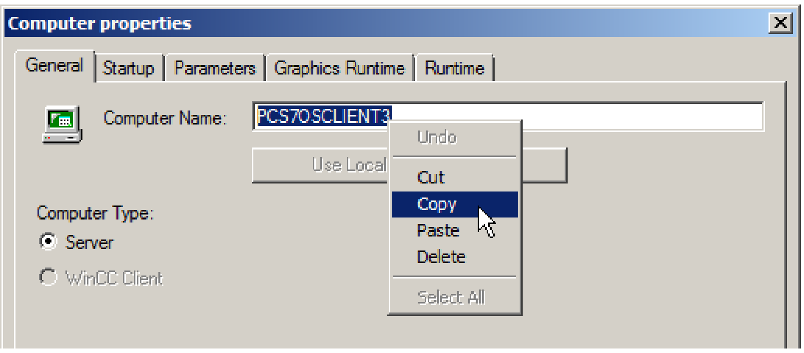


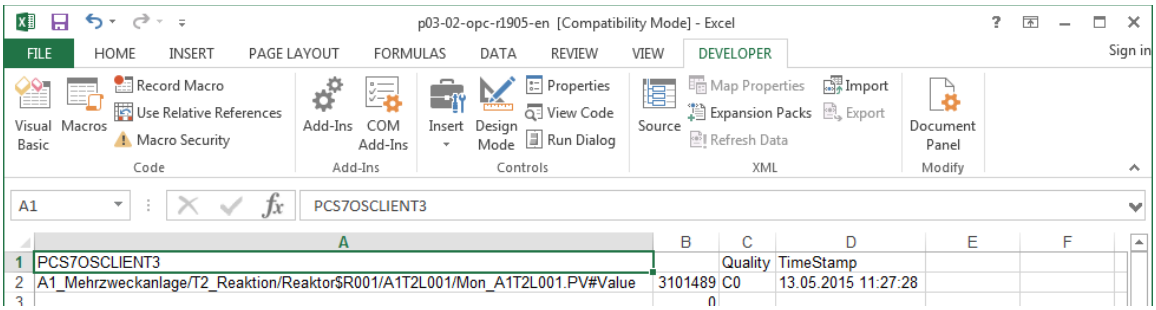
7. For the access to tags in 'OPCServer WinCC' to be able to function, the reference to the dll 'Siemens OPC DA Automation 2.0' must be selected here. If the reference is not available, this must be entered here using the 'Browse' function. The path is 'C:\Program Files (x86)\Common Files\Siemens\opc\sopcdaauto.dll' (→ Siemens OPC DAAutomation 2.0 → Browse → C:\Program Files\Common Files (x86)\Siemens\opc\ sopcdaauto.dll → OK)



8. Now, the computer name must be entered in cell A1. You can copy it in WinCC under Computer properties. (→ A1 → Computer Name)



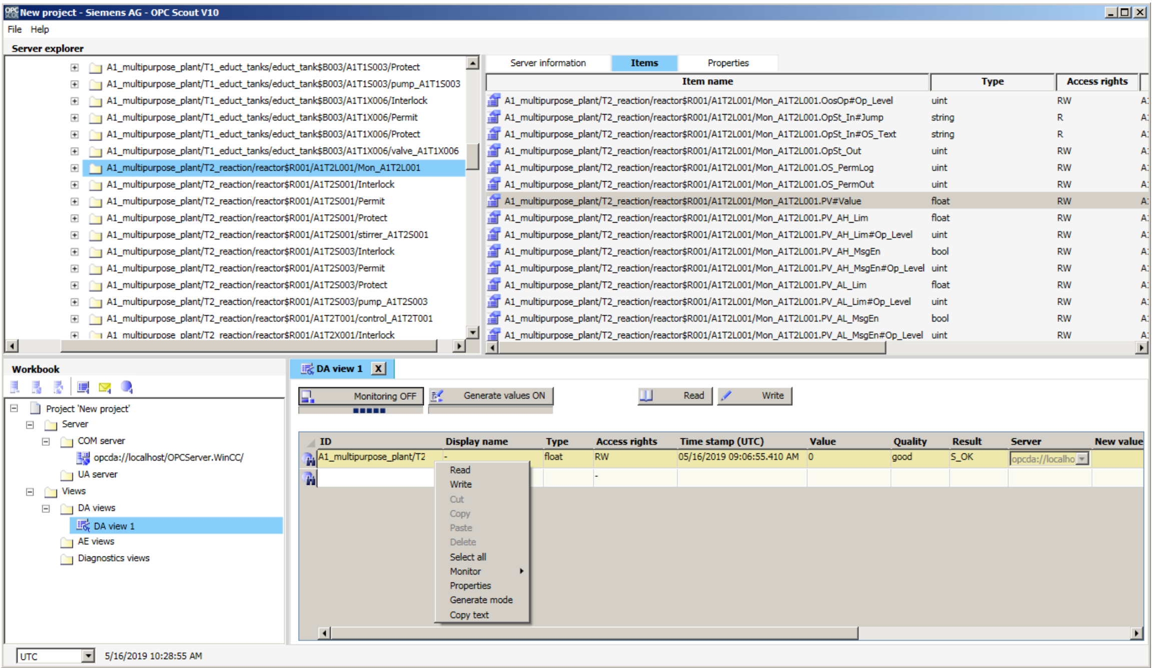


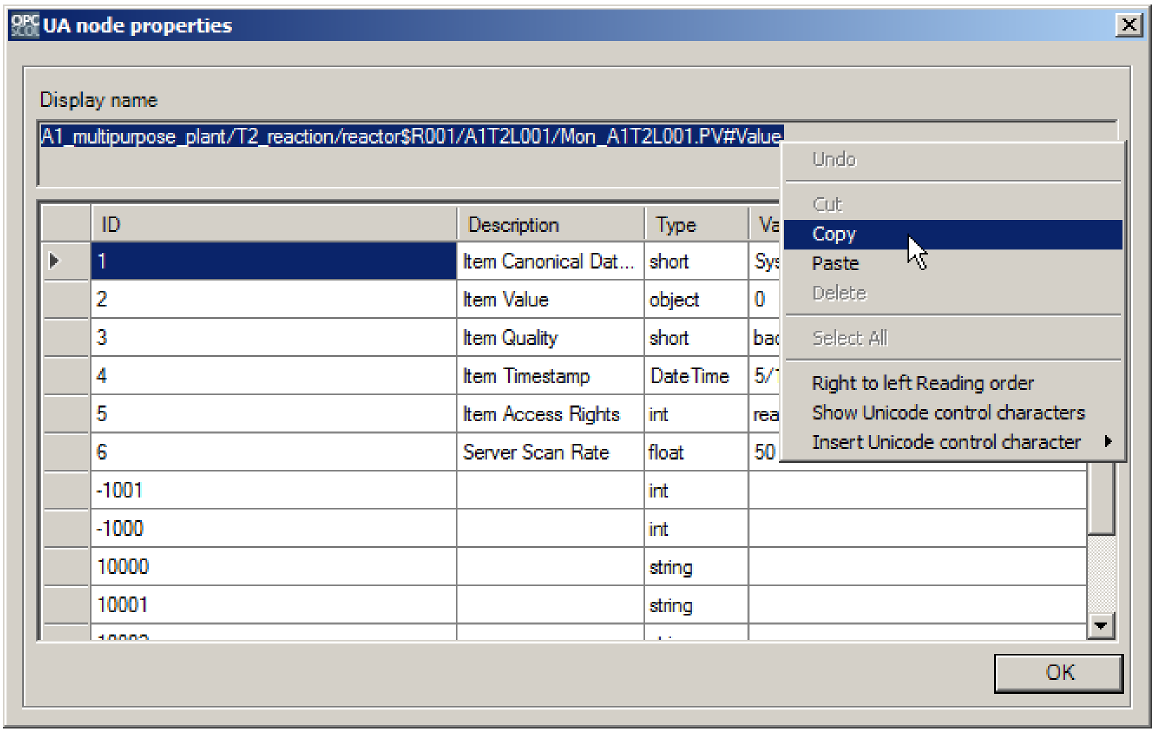


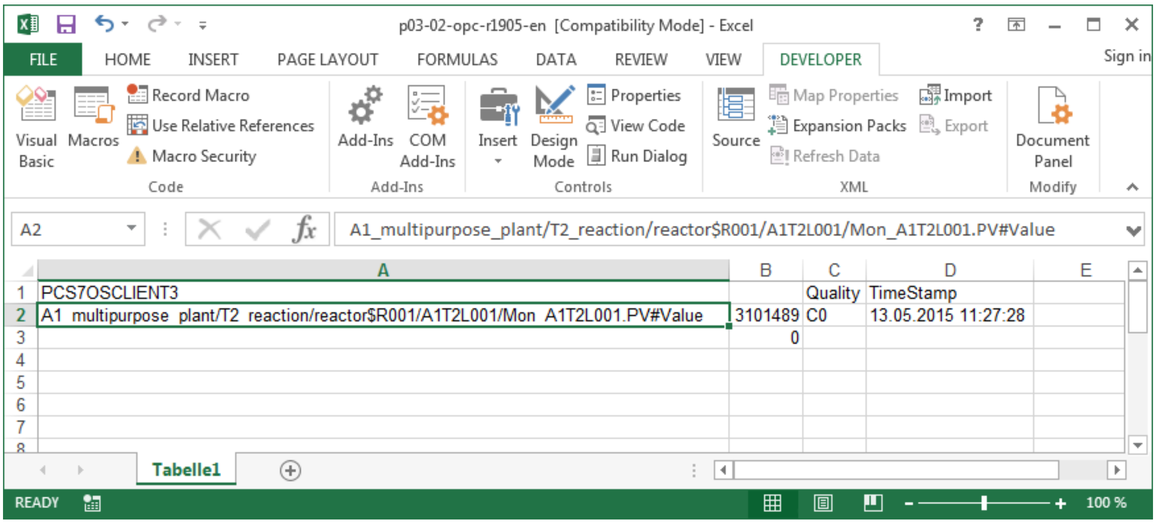
Note

– *The needed computer name shown here is only an example. You must enter your own local computer name, which you can determine as shown above. If the configured and local computer names are different, click the 'Use Local Computer Name' button to obtain the correct name.*

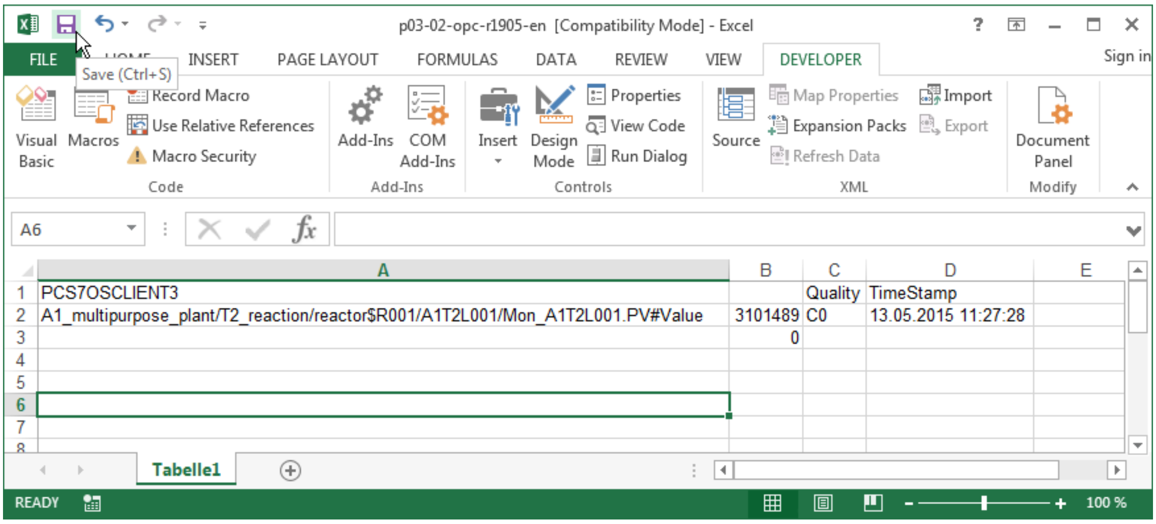
9. The complete tag name must be entered in cell A2. It can be copied from OPC Scout V10. (→ A2 → A1_multipurpose_plant/T2_reaction/reactor$R001/A1T2L001/monitor_A1T2L001.PV#Value)





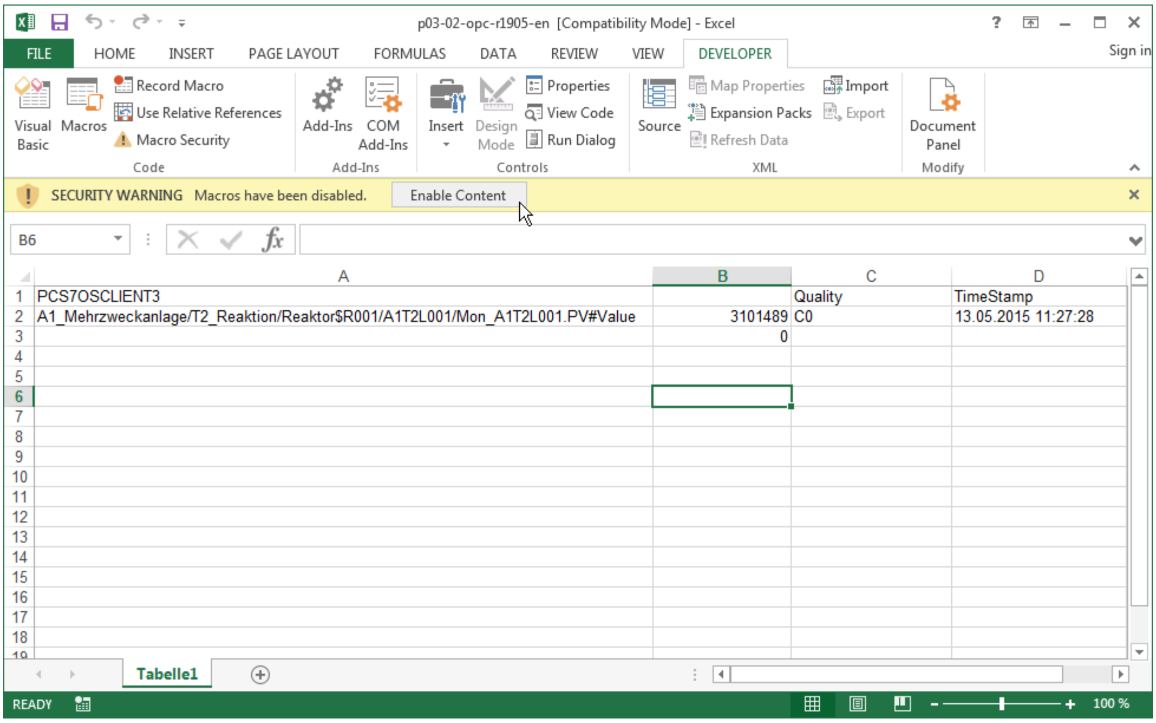


10. Next, save and close the Microsoft Excel file prior to reopening it with a double-click. (→ Save → ✕ → PCS7_SCE_0302_OPC.xls)

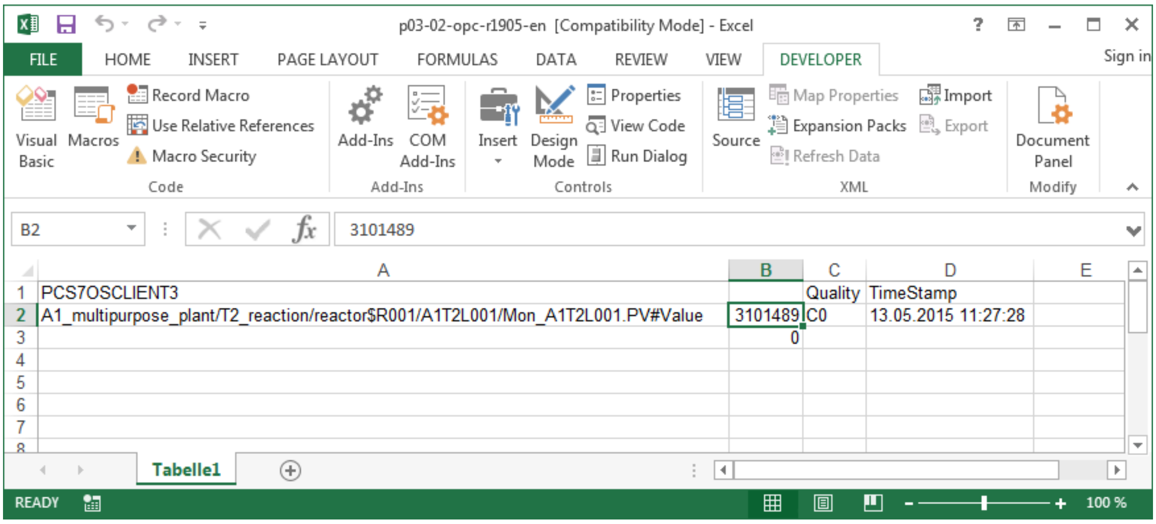


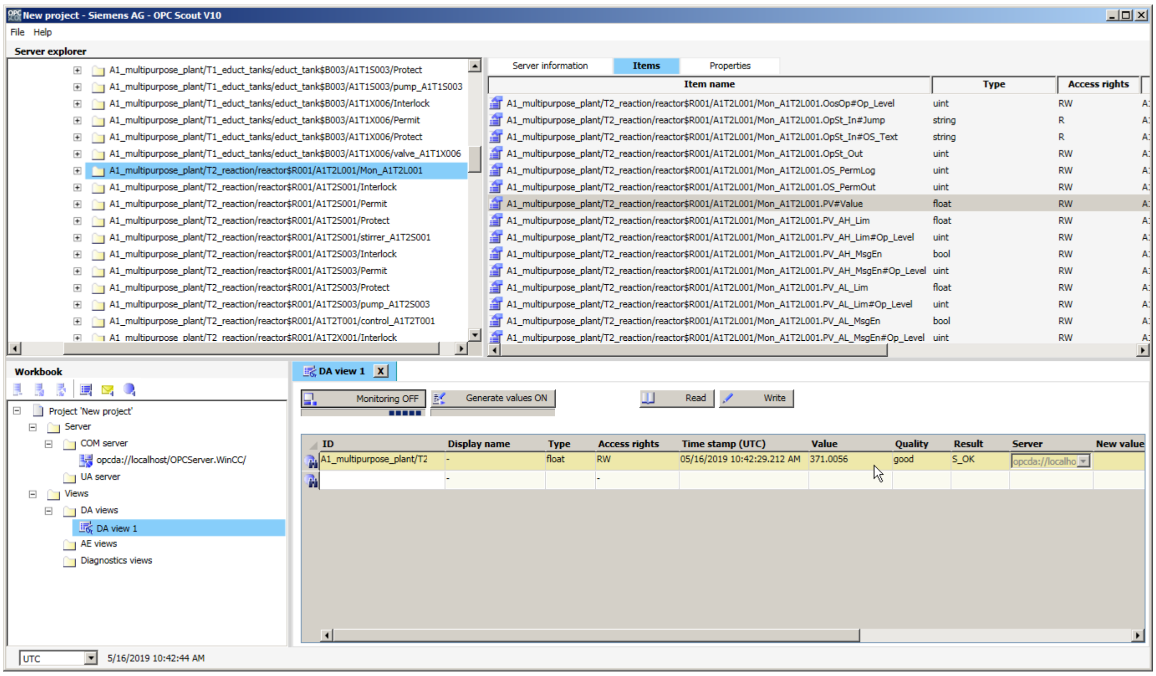
8.4 Tag access in Excel

1. When Excel opens, take note of the security warning and enable the macros for the following session. (→ Enable Content)

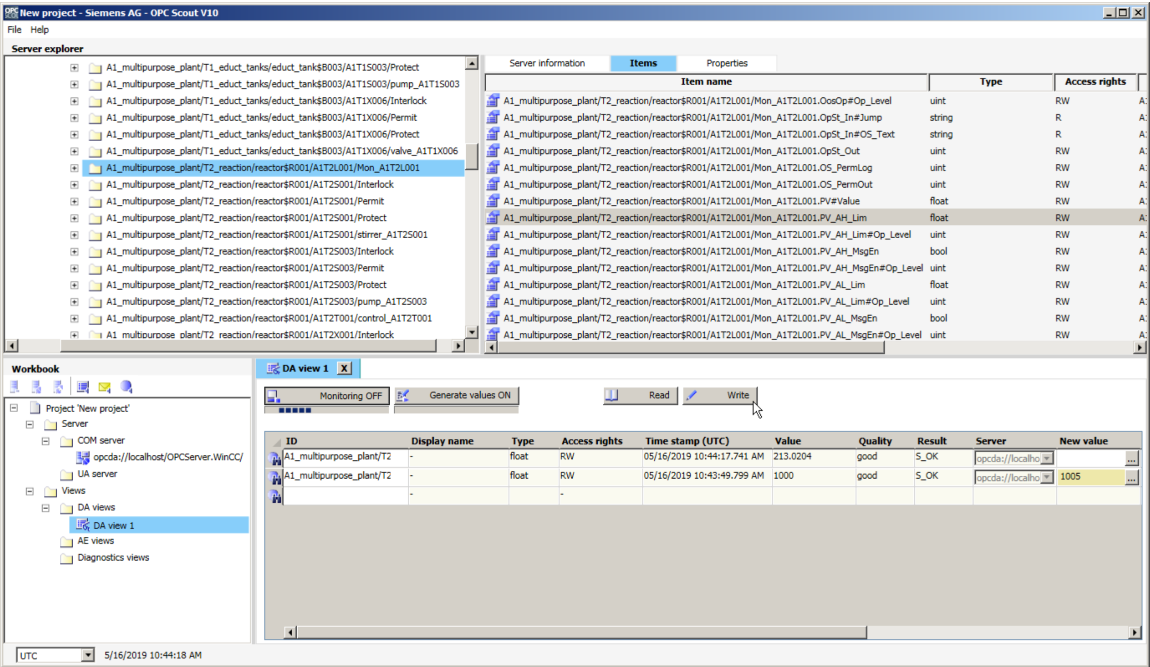


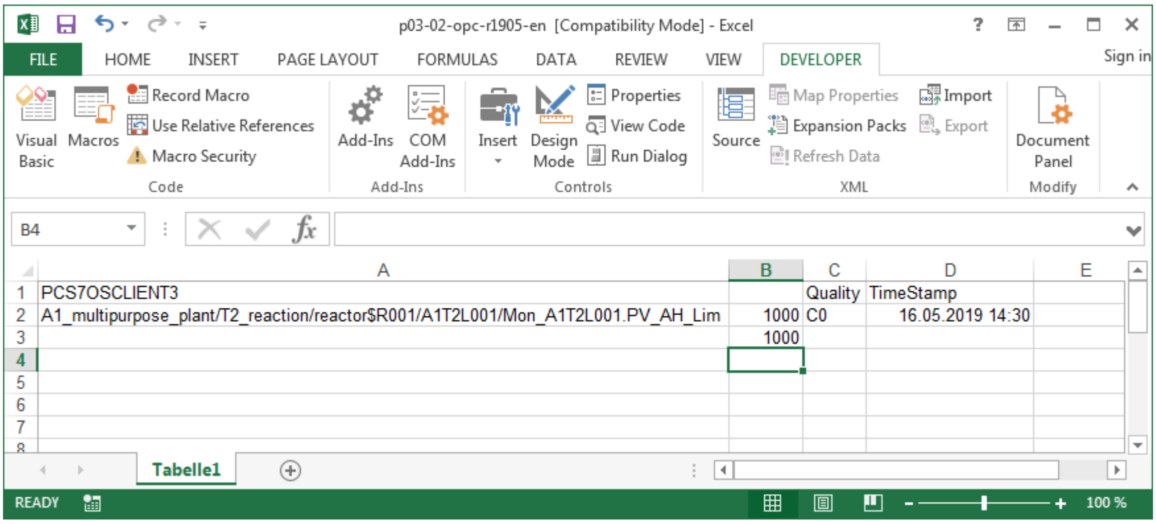
2. You can now monitor the tag in cell B2.





3. You can write a new value for the tag in cell B3. You cannot use the process value for this as above because it would be overwritten again immediately. For this reason, use a fixed parameter, for example, A1_multipurpose_plant/T2_reaction/reactor$R001/ A1T2L001/monitor_A1T2L001.PV_AH_Lim, to test this functionality. Change the value to 1005 with OPC Scout and then back to 1000 with the Excel file.





8.5 Checklist – step-by-step instruction

The following checklist helps students to independently check whether all steps of the step-by-step instruction have been carefully completed and enables them to successfully complete the module on their own.

No.	Description	Checked
1	Simulation and WinCC Runtime started	
2	Tag access with OPC Scout successful	
3	Excel configured	
4	Tag access with Excel successful	

Table 1: Checklist for step-by-step instructions

9 Exercises

In the exercises, you apply what you learned in the theory section and in the step-by-step instructions.

The existing multiproject p03-01-exercise-r1905-en.zip and the table p03-02-opc-template-r1905-en.xls (in zip file "p03-02-files-r1905-en.zip') will be used for this. The download of the projects/files is stored as a zip file on the SCE Internet for the respective module.

The objective of this exercise is to identify and read out a measured value via an OPC item.

9.1 Tasks

1. Using PCS 7 and OPC Scout, identify the corresponding tag name of the temperature measurement in Reactor R001.
2. Modify the tag name in the table from the step-by-step instructions with the new name and read out the value.

9.2 Checklist – exercise

The following checklist helps students to independently check whether all steps of the exercise have been carefully completed and enables them to successfully complete the module on their own.

No.	Description	Checked
1	Tag name of temperature measurement A1T2T001 identified	
2	Temperature A1T2T001 successfully read out	

Table 2: Checklist for exercises

10 Additional information

More information for further practice and consolidation is available as orientation, for example: Getting Started, videos, tutorials, apps, manuals, programming guidelines and trial software/ firmware, under the following link:

siemens.com/sce/pcs7

Preview "Additional information"

Getting Started, Videos, Tutorials, Apps, Manuals, Trial-SW/Firmware

- SIMATIC PCS 7 Overview
- SIMATIC PCS 7 Videos
- Getting Started
- Application Examples
- Download Software/Firmware
- SIMATIC PCS 7 Website
- SIMATIC S7-400 Website

Further Information

Siemens Automation Cooperates with Education
siemens.com/sce

Siemens SIMATIC PCS 7
siemens.com/pcs7

SCE Learn-/Training Documents
siemens.com/sce/documents

SCE Trainer Packages
siemens.com/sce/tp

SCE Contact Partners
siemens.com/sce/contact

Digital Enterprise
siemens.com/digital-enterprise

Industrie 4.0
siemens.com/future-of-manufacturing

Totally Integrated Automation (TIA)
siemens.com/tia

TIA Portal
siemens.com/tia-portal

SIMATIC Controller
siemens.com/controller

SIMATIC Technical Documentation
siemens.com/simatic-docu

Industry Online Support
support.industry.siemens.com

Product catalogue and online ordering system Industry Mall
mall.industry.siemens.com

Siemens
Digital Industries, FA
P.O. Box 4848
90026 Nuremberg
Germany

siemens.com/sce