Application Example • 07/2017

TIA Portal Openness: Generating a Modular Machine with S7-1500

TIA Portal Openness V14 SP1

https://support.industry.siemens.com/cs/ww/en/view/109739678

Warranty and Liability

Note

The Application Examples are not binding and do not claim to be complete regarding the circuits shown, equipping and any eventuality. The Application Examples do not represent customer-specific solutions. They are only intended to provide support for typical applications. You are responsible for ensuring that the described products are used correctly. These Application Examples do not relieve you of the responsibility to use safe practices in application, installation, operation and maintenance. When using these Application Examples, you recognize that we cannot be made liable for any damage/claims beyond the liability clause described. We reserve the right to make changes to these Application Examples at any time without prior notice.
If there are any deviations between the recommendations provided in these Application Examples and other Siemens publications – e.g. Catalogs – the contents of the other documents have priority.

We do not accept any liability for the information contained in this document.
Any claims against us – based on whatever legal reason – resulting from the use of the examples, information, programs, engineering and performance data etc., described in this Application Example shall be excluded. Such an exclusion shall not apply in the case of mandatory liability, e.g. under the German Product Liability Act (“Produkthaftungsgesetz”), in case of intent, gross negligence, or injury of life, body or health, guarantee for the quality of a product, fraudulent concealment of a deficiency or breach of a condition which goes to the root of the contract (“wesentliche Vertragspflichten”). The damages for a breach of a substantial contractual obligation are, however, limited to the foreseeable damage, typical for the type of contract, except in the event of intent or gross negligence or injury to life, body or health. The above provisions do not imply a change of the burden of proof to your detriment.
Any form of duplication or distribution of these Application Examples or excerpts hereof is prohibited without the expressed consent of the Siemens AG.

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 only form one element of such a concept.
Customer is responsible to prevent unauthorized access to its plants, systems, machines and networks. Systems, machines and components should only be connected to the enterprise network or the internet if and to the extent necessary and with appropriate security measures (e.g. use of firewalls and network segmentation) in place.
Additionally, Siemens’ guidance on appropriate security measures should be taken into account. For more information about industrial security, please visit http://www.siemens.com/industrialsecurity.

Siemens’ products and solutions undergo continuous development to make them more secure. Siemens strongly recommends to apply product updates as soon as available and to always use 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 http://www.siemens.com/industrialsecurity.

Table of Contents

Warranty and Liability 2

1 Task 4

2 Solution 5

2.1 Overview 5
2.2 Hardware and software components 7
2.2.1 Validity 7
2.2.2 Components used 7

3 Basics 10

3.1 Basics on TIA Portal Openness 10
3.2 Basics on the “option handling” configuration control 10
3.3 Basics on “Apache Tomcat” web server 11

4 Mode of Operation 12

4.1 General overview 12
4.2 Modular project 13
4.2.1 Organization block “Main” [OB 1] 14
4.2.2 “Conveyor” [FB 1] function block 15
4.2.3 “Drive” [FB 2] function block 16
4.3 Initial project 17
4.3.1 “MainOptionHandling” [OB 125] organization block 17
4.3.2 “ActDeactDrive” [FB 15] function block 18
4.3.3 “PlantConfigData” [DB 7] global data block 19
4.3.4 „SpeedAxisData“ [DB 114] global data block 19

5 Installation and Commissioning 20

5.1 Installing the software (download) 20
5.2 Installation of the “Apache Tomcat” web server optional 21
5.2.1 Installation of the “Apache Tomcat” web server 21
5.2.2 Setting the Windows Firewall 23
5.2.3 Installing the openness web application „WebAppOpenness“ 26
5.2.4 Security setting in the “Apache Tomcat” directory 27
5.3 Creating the configuration file 28
5.4 Installation of the OpennessApplication “TIA Openness Project” 28
5.5 Managing the user rights 31
5.6 Granting access 32

6 Operating the Application Example 33

6.1 Scenario A 34
6.2 Scenario B 37

7 Links & Literature 39

8 History 39

1 Task

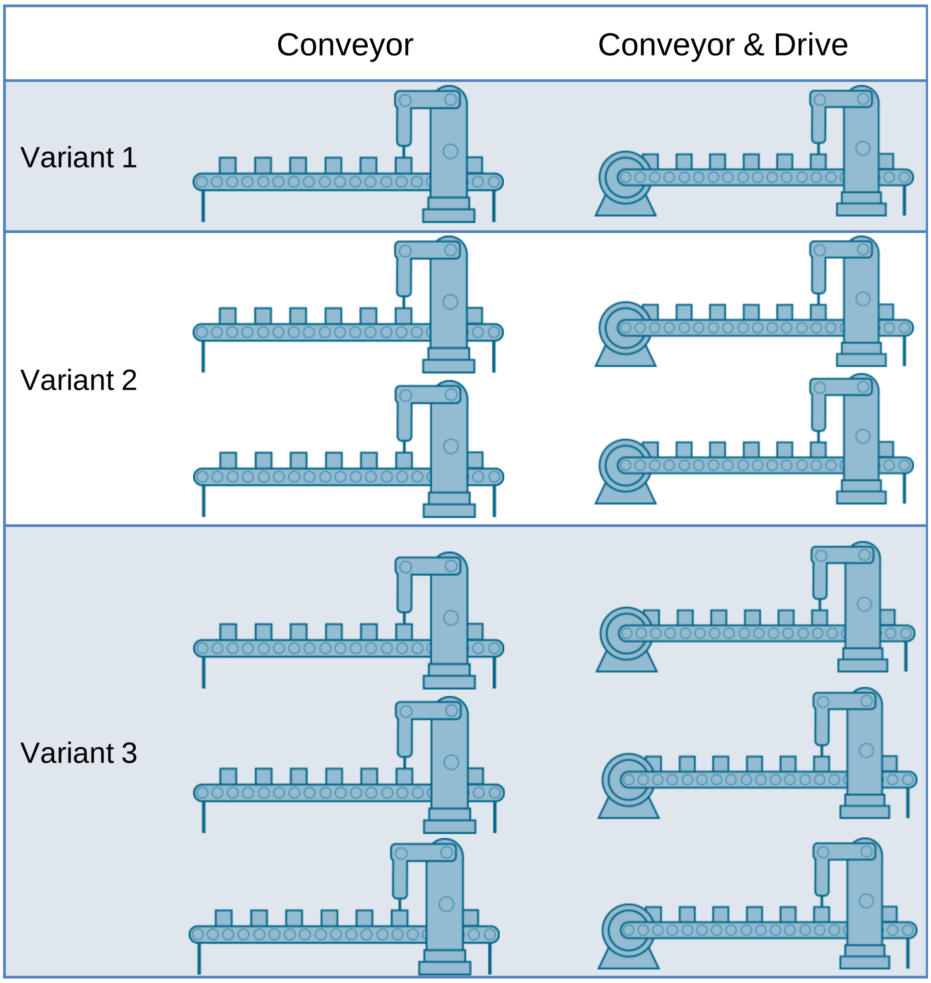
Introduction

TIA Portal Openness V14 SP1 is an open interface to TIA Portal V14 SP1. With TIA Portal Openness you can automatize and remote control your engineering tasks in the TIA Portal Openness with independent scripts or programs.

Overview of the application example

The following figure gives you an overview of the application example. In the right column the modular system “Conveyor” is supplemented by a “Drive”. In the variants the quantity is supplemented by identical systems.

Figure 1-1



Description of the application example

The aim of the application example is to start the TIA Portal via a remote control and to configure and generate a STEP 7 project for a modular machine. All variants of the system are automatized with the same STEP 7 project and without reconfiguration.

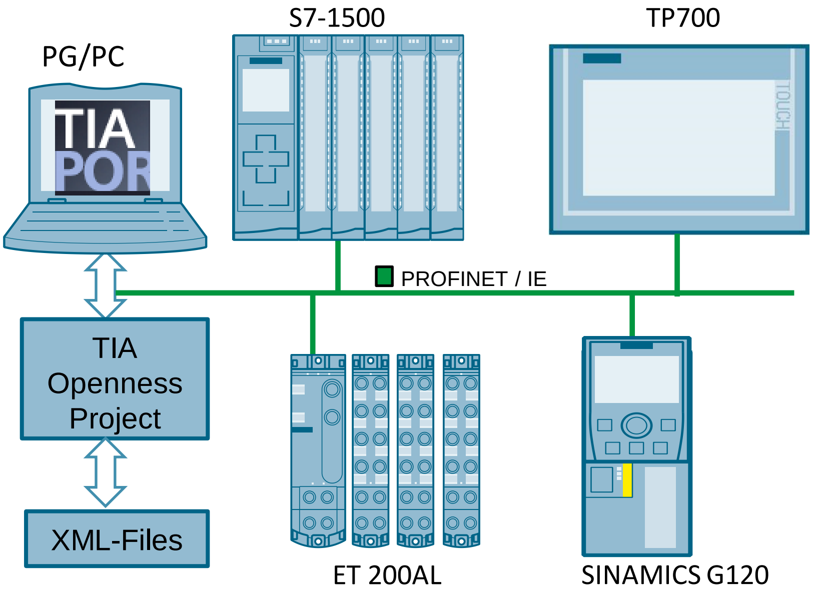
2 Solution

2.1 Overview

Schematic layout

The diagrammatic representation below shows what components are involved in the maximum configuration of the solution:

Figure 2-1



Configuration

The "TIA Openness Project" starts the TIA Portal and opens the initial project. This project consists of a pre-configured maximum configuration, which includes all variants of the machine. Depending on variant the „TIA Openness Project" changes the previously created XML files (program blocks) and imports this in the project. The optional operator panel visualizes the plant.

Advantages

- Remote control of the TIA Portal in order to create projects.
- Provision of project data of the TIA Portal for external applications.
- Easy project development due to the use of a single STEP 7 project for all variants.
- Small engineering effort for configuration, construction and wiring.
- Standardization of hardware and software for the various variants of a machine.

Assumed knowledge

Basic knowledge for the following issues is assumed:

- STEP 7 (TIA Portal)
- Programming environment, for example, Microsoft Visual Studio
- Object-oriented programming, for example, C# and JSP (JavaServer Pages)

2.2 Hardware and software components

2.2.1 Validity

The application example was tested with

- STEP 7 (TIA Portal) V14 SP1

2.2.2 Components used

The hardware configuration can be carried out with the TIA Selection Tool: http://www.siemens.en/tia-selection-tool.

The application example has been created with the following components:

Hardware components

Table 2-1

Component	Qty.	Article number	Note
CPU 1511C-1 PN	1	6ES7511-1CK00-0AB0	-
MEMORY CARD, 24 MB	-	6ES7954-8LF02-0AA0	-
DI16/DQ16X24VDC	2	6ES7523-1BL00-0AA0	-
SCALANCE X208	1	GK5 208-0BA10-2AA3	-
TP700 COMFORT	1	6AV2124-0GC01-0AX0	-
ET 200AL IM157-1 PN	1	6ES7157-1AB00-0AB0	-
ET 200AL, DI 4+DQ 4X24VDC/0.5A, 8XM8	3	6ES7143-5BF00-0BA0	-
SINAMICS G120 CU240E-2 PN-F E-TYP	1	6SL3244-0BB13-1FA0	-
SINAMICS POWER MODULE PM240-2	1	6SL3210-1PB13-0AL0	-
DIN RAIL 245MM (9.6")	1	6ES7590-1AC40-0AA0	-
POWER SUPPLY S7-1500 PM1507	1	6EP1332-4BA00	-
BUS CABLE ET CONNECTION, 0.19M	1	6ES7194-2LH02-0AA0	-
POWER LINE M8, 0.19M	1	6ES7194-2LH02-1AA0	-
SINAMICS G120 INTELLIGENT OPERATOR PANEL(IOP)	1	6SL3 255-0AA00-4JA0	-

Software components

Table 2-2

Component	Qty.	Article number	Note
STEP 7 Professional V14 SP1	1	6ES7822-1..04-..	-
WinCC Advanced V14 SP1	1	6AV2102-0AA04-0AA5	-
TIA Portal Openness V14 SP1	1		Included in the scope of delivery of STEP 7 Professional V14 or WinCC Advanced V14
Microsoft Visual Studio 2015	1	-	.NET Framework 4.6.1

Example files and projects

The following list contains all files and projects that are used in this application example.

Table 2-3

No.	Component	Note
1	109739678_TIAV14SP1_OpennessModularMachine_CODE_v10.zip	This zip file includes:
	TIA_OpennessModularMachine_Project	The TIA Portal STEP 7 project
	TIA_OpennessModularMachine_Library	The TIA Portal library
	SourceCode OpennessApp	The Microsoft Visual Studio “OpennessApplication” project
	WebAppOpenness	The openness web application “WebAppOpenness”
	XML	The XML files for the configuration of individual variants
	config.xml	The paths for the web application, the STEP 7 project and TIA Portal library
	TIA_OpennessModularMachine.exe	The executable file for OpennessApplication “TIA Openness Project”
2	109739678_TIA_OpennessModularMachine_PRODUCTS_v10.zip	The zipped file includes the TIA Selection Tool file with hardware and software products.
3	109739678_TIAV14SP1_OpennessModularMachine_DOC_v10_en.pdf	This document

Note The XML structure has been changed in TIA Portal V14 SP1. For this reason the project "TIA_OpennessModularMachine_Project" of V13 and V14 is not the same project in V14 SP1.

The TIA Portal projects in V13 and V14 are different in their structures to the TIA Portal project in V14 SP1.

3 Basics

3.1 Basics on TIA Portal Openness

In STEP 7 V14 SP1 and WinCC V14 SP1, TIA Portal Openness V14 SP1 is available for free on the respective product DVDs. For you to be able to use TIA Portal Openness V14, STEP 7 or WinCC V14 has to be installed.

TIA Portal Openness V14 SP1 provides DLLs via which you can access the TIA Portal platform. These DLLs are based on .NET Framework 4.6.1

With TIA Portal Openness you automatize engineering by controlling the TIA Portal from a program created by you.

With TIA Portal Openness you can perform the following actions:

- Create project data
- Modify project data
- Delete project data
- Read project data
- Provide projects and project data for other applications.

Note

More information on TIA Portal Openness can be found in the manual “Automating SIMATIC Projects via Scripts” and in the application example “TIA Portal Openness Introduction and Demo Application”.

https://support.industry.siemens.com/cs/de/en/view/109477163/72488745483

https://support.industry.siemens.com/cs/de/en/view/108716692

3.2 Basics on the “option handling” configuration control

The “option handling” configuration control enables flexible configuration steps for several stations within in a project and can be adjusted to customer requirements.

In a single STEP 7 project the maximum configuration of the stations is configured that comprises all module variants of a machine. This project can be used for several configuration steps of I/O stations.

To be able to use configuration control, the following steps are required:

1. Enable configuration control in the module.
2. Configure maximum configuration or program and write control data record that is valid as basis for the configuration control.

Note

More information on configuration control “option handling” can be found in the application example “Variants Management with SIMATIC S7”. https://support.industry.siemens.com/cs/ww/en/view/29430270

3.3 Basics on “Apache Tomcat” web server

Apache Tomcat is an open source web server that consists of Java Servlets of the JSP engine (JavaServer pages). With Apache Tomcat, web applications that are written with Java can be executed.

Apache Tomcat can be operated as independent web server with the HTTP connector or integrated with AJP connector in other web servers, for example, Microsoft IIS (Internet Information Services) or Apache web server.

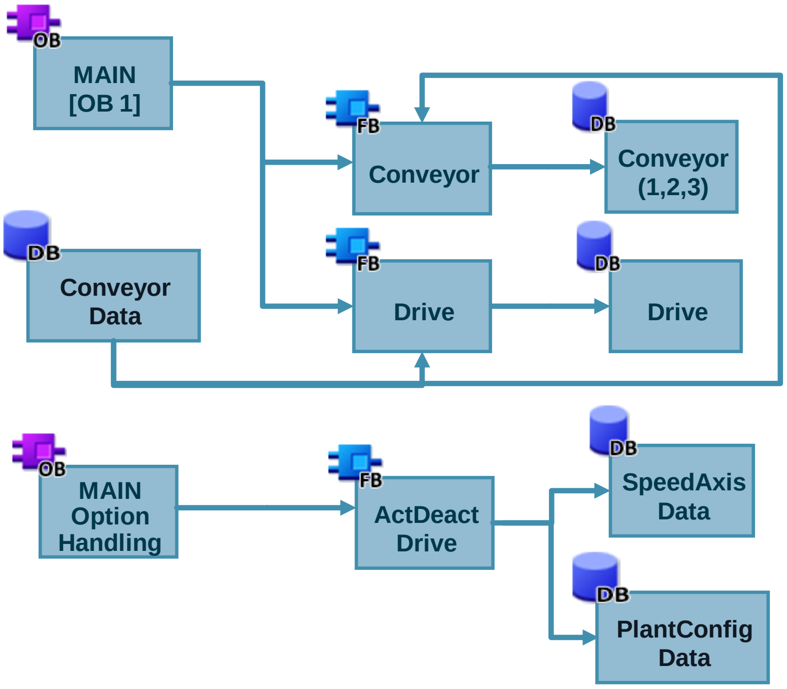
Note More information on the “Apache Tomcat” web server can be found on the Apache Tomcat website http://tomcat.apache.org/.

4 Mode of Operation

4.1 General overview

The figure below shows the program structure of the entire STEP 7 project.

Figure 4-1



Description of the core functionality

- The control with the modules for the maximum configuration is configured in the STEP 7 project.
- For all variants of the machine an independent data structure has been created in a database (XML files).
- To be able to display the selected variant, the user program transfers the required data record into the controller.

This application example consists of the following blocks:

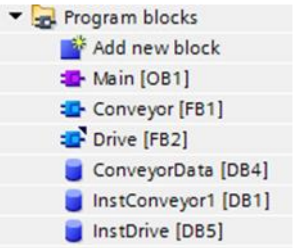
Table 4-1

Symbolic name	Description
Modular project	
Main [OB 1]	The OB 1 block contains the cyclic program. For the variant change the modules are created from the hardware catalog and for the drive the option handling is used.
Conveyor [FB 1]	The function block FB 1 includes the configuration of the "Conveyor".
ConveyorData [DB 4]	Data block DB 4 includes the tags for the maximum configuration.
Drive [FB 2]	The function block FB 2 includes the configuration of the "Drive".
Initial project	
MainOptionHandling [OB 125]	Block OB 125 creates a defined initial state for configuration of the "Drive".
ActDeactDrive [FB 15]	Function block FB 15 defines the states of the drive.
PlantConfigData [DB 7]	Data block DB 7 includes the status information of the "Conveyor" and the "Drive".
SpeedAxisData [DB 114]	Data block DB 114 includes the database of the "Drive".

4.2 Modular project

The initial configuration for this display is variant 1 "Conveyor & Drive".

Figure 4-2



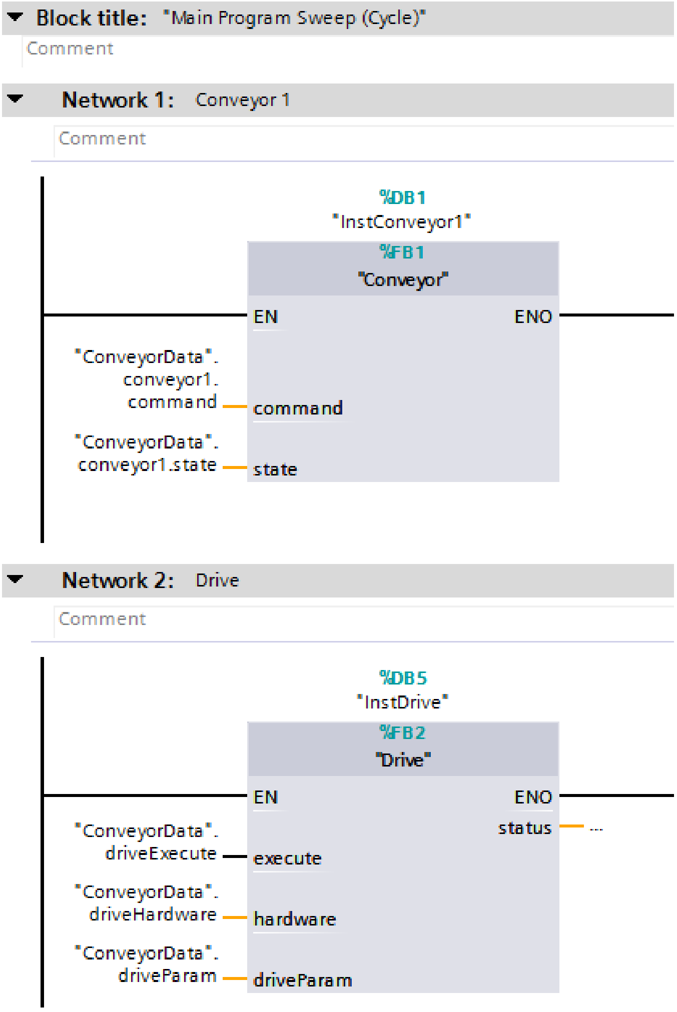
- Program blocks
  - Add new block
  - Main [OB1]
  - Conveyor [FB1]
  - Drive [FB2]
  - ConveyorData [DB4]
  - InstConveyor1 [DB1]
  - InstDrive [DB5]

4.2.1 Organization block “Main” [OB 1]

OB 1 block contains the cyclic program and is used to operate the application. If the variant is changed, the hardware configuration of the modules is created from the hardware catalog via Openness and for the drive the “OptionHandling” is used.

The configuration basis for OB 1 is the “XML > Main.xml” file.

Figure 4-3



Block title: "Main Program Sweep (Cycle)"

Comment

Network 1: Conveyor 1

Comment

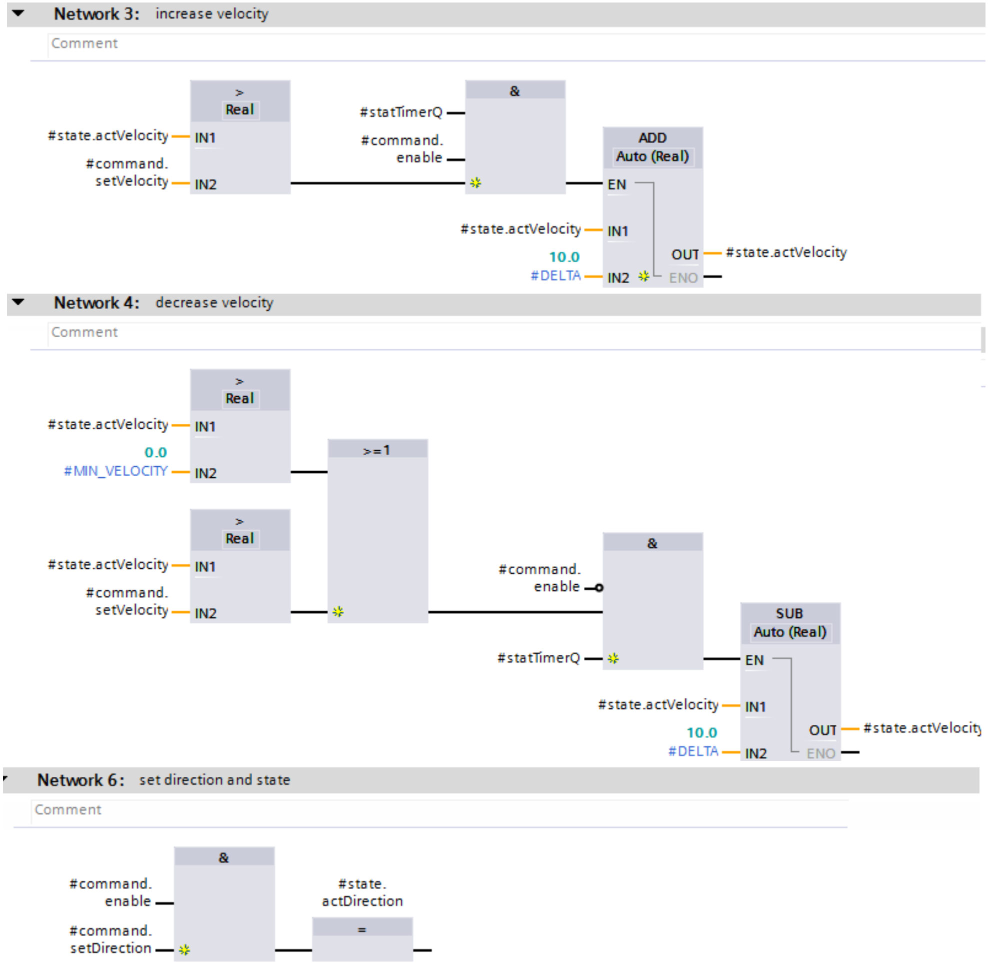
Network 2: Drive

Comment

4.2.2 “Conveyor” [FB 1] function block

The function block “conveyor” includes the parameters “velocity and direction” as well as the status information of the “conveyor”. The configuration basis for FB 1 is the “XML > Conveyor.xml” file.

Figure 4-4



Network 3: increase velocity

Comment

Network 4: decrease velocity

Comment

Network 6: set direction and state

Comment

The following table includes all tags that are used with function block FB 1.

Table 4-2

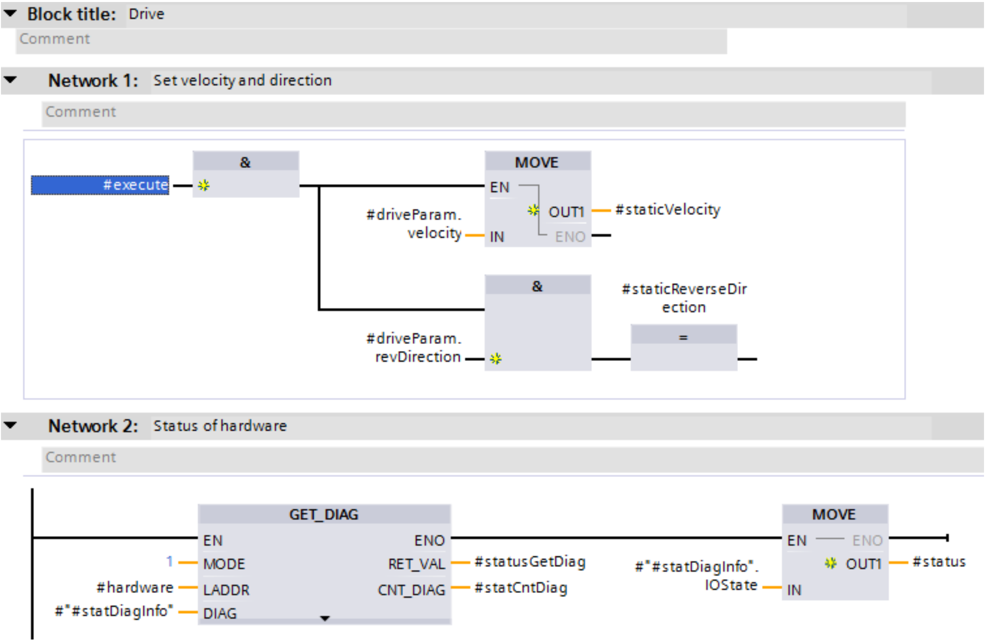
Typ	Name	Parameter	Datentyp	Beschreibung
Input	command		PLC-data types	Structur “ConveyorCommands”
		enable	BOOL	Activate/deactivate
		setMode	BOOL	
		conveyorID	INT	Hardware ID module (conveyor number)
		hardware	HW_SUBMODULE	Hardware ID sub module
		setDirection	BOOL	Direction conveyor
		setVelocity	REAL	Velocity conveyor

InOut	state		PLC-data types	Structure „ConveyorState“
		enabled	BOOL	Activate
		actState	INT	State activate
		actMode	BOOL	Mode activate
		actDirection	BOOL	Direction conveyor activate
		actVelocitiy	REAL	Velocity conveyor activate

4.2.3 “Drive” [FB 2] function block

The “drive” function block includes the parameters “velocity and direction” as well as the status information of the “Drive”. The configuration basis for FB 2 is the “TIA_OpennessModularMachine_Library” library.

Figure 4-5



The following table includes all tags that are used with function block FB 7.

Table 4-3

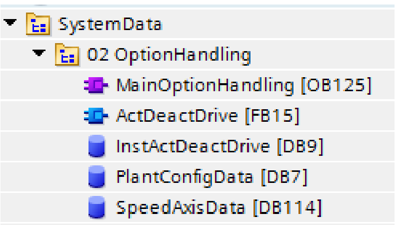
Type	Parameter	Data type	Description
Input	excute	BOOL	Executing in the TRUE cycle
	hardware	HW_SUBMODULE	Hardware ID of the module (drive)
Output	status	WORD	I/O status of the module
InOut	driveParam	typeDriveParam	Velocity and direction (drive)
Static	staticVelocity	REAL	Velocity status interim result
	staticReverseDirection	BOOL	Changing direction
	#statDiagInfo	DIS	Diagnostic information at

Type	Parameter	Data type	Description
			Mode = 1
	statusGetDiag	WORD	Status of instruction
	statCntDiag	WORD	Reserved (always “0”)

4.3 Initial project

The initial project includes the system data “SystemData” that serves as basis for the configuration of the “Drive”. The structure is specified for all variants.

Figure 4-6

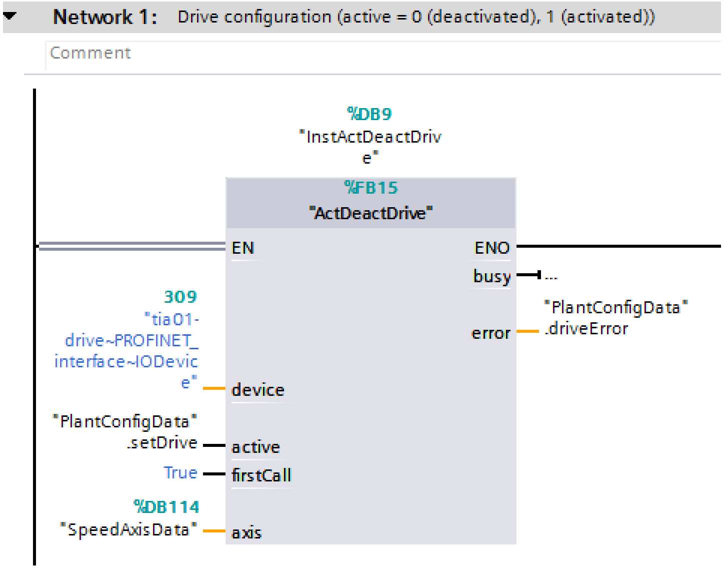


- SystemData
  - 02 OptionHandling
    - MainOptionHandling [OB125]
    - ActDeactDrive [FB15]
    - InstActDeactDrive [DB9]
    - PlantConfigData [DB7]
    - SpeedAxisData [DB114]

4.3.1 “MainOptionHandling” [OB 125] organization block

The “MainOptionHandling” block creates a defined initial state for configuration of the “Drive”. In the user interface of the “TIA Openness Project” application or the “WebAppOpenness” web application, the various variants of the machine can be selected.

Figure 4-7



Network 1: Drive configuration (active = 0 (deactivated), 1 (activated))

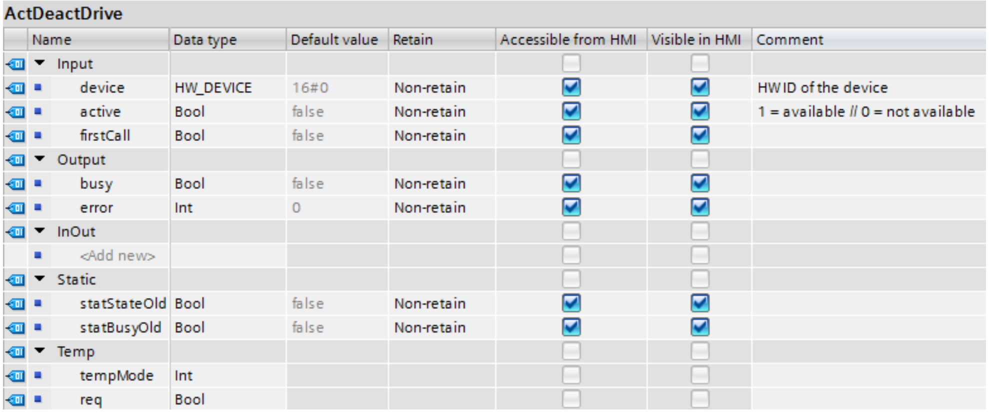
Comment

Note More information on the configuration control “option handling” can be found in the application example “Variants Management with SIMATIC S7”.
https://support.industry.siemens.com/cs/ww/en/view/29430270

4.3.2 “ActDeactDrive” [FB 15] function block

The “ActDeactDrive” function block includes all status information of the drive.

Figure 4-8



ActDeactDrive

Name	Data type	Default value	Retain	Accessible from HMI	Visible in HMI	Comment
▼ Input				□	□	
device	HW_DEVICE	16#0	Non-retain	☑	☑	HW ID of the device
active	Bool	false	Non-retain	☑	☑	1 = available // 0 = not available
firstCall	Bool	false	Non-retain	☑	☑	
▼ Output				□	□	
busy	Bool	false	Non-retain	☑	☑	
error	Int	0	Non-retain	☑	☑	
▼ InOut				□	□	
<Add new>				□	□	
▼ Static				□	□	
statStateOld	Bool	false	Non-retain	☑	☑	
statBusyOld	Bool	false	Non-retain	☑	☑	
▼ Temp				□	□	
tempMode	Int			□	□	
req	Bool			□	□	

The following table includes all tags that are used with function block FB 15.

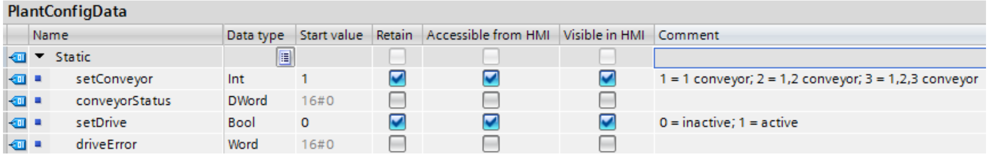
Table 4-4

Type	Parameter	Data type	Description
Input	device	HW_DEVICE	Hardware ID device
	active	Bool	Activation: 1 = available 0 = not available
	firstCall	Bool	First call
Output	busy	Bool	Busy = 1: Job is still active Busy = 0: Job was completed
	error	Int	Device activate/deactivate
Static	statStateOld	Bool	Saving status interim result
	statBusyOld	Bool	
Temp	tempMode	Int	1 = activate 2 = deactivate
	req	Bool	Carrying out activation or deactivation

4.3.3 “PlantConfigData” [DB 7] global data block

The following figure shows the structure of the global “PlantConfigData” data block. Data block DB 7 includes the status information of the conveyor (conveyor 1, 2 and 3) and the drive (drive 0 and 1).

Figure 4-9



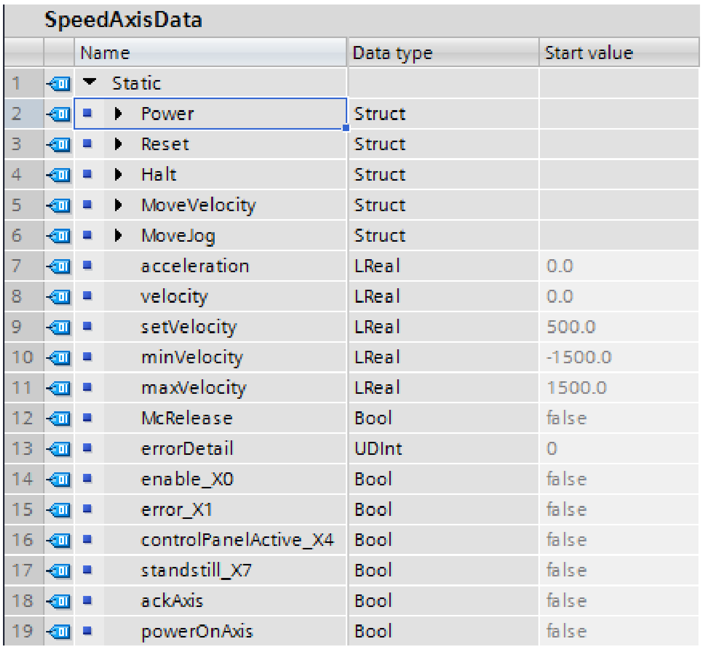
PlantConfigData

Name	Data type	Start value	Retain	Accessible from HMI	Visible in HMI	Comment
Static			□	□	□	
setConveyor	Int	1	☑	☑	☑	1 = 1 conveyor; 2 = 1,2 conveyor; 3 = 1,2,3 conveyor
conveyorStatus	DWord	16#0	□	□	□	
setDrive	Bool	0	☑	☑	☑	0 = inactive; 1 = active
driveError	Word	16#0	□	□	□	

4.3.4 „SpeedAxisData“ [DB 114] global data block

The “SpeedAxisData” data block includes the database of the drive.

Abbildung 4-10



SpeedAxisData

	Name	Data type	Start value
1	Static		
2	Power	Struct	
3	Reset	Struct	
4	Halt	Struct	
5	MoveVelocity	Struct	
6	MoveJog	Struct	
7	acceleration	LReal	0.0
8	velocity	LReal	0.0
9	setVelocity	LReal	500.0
10	minVelocity	LReal	-1500.0
11	maxVelocity	LReal	1500.0
12	McRelease	Bool	false
13	errorDetail	UDInt	0
14	enable_X0	Bool	false
15	error_X1	Bool	false
16	controlPanelActive_X4	Bool	false
17	standstill_X7	Bool	false
18	ackAxis	Bool	false
19	powerOnAxis	Bool	false

5 Installation and Commissioning

This chapter shows you how to install and commission the application example.

5.1 Installing the software (download)

Load the following download file in a directory of your choice and unzip the zipped folder.

„109739678_TIAV14SP1_OpennessModularMachine_CODE_v10.zip“

The zipped download file includes:

- SourceCode OpennessApp
- TIA_OpennessModularMachine_Library
- TIA_OpennessModularMachine_Project
- WebAppOpenness
- XML
- config.xml
- TIA_OpennessModularMachine.exe

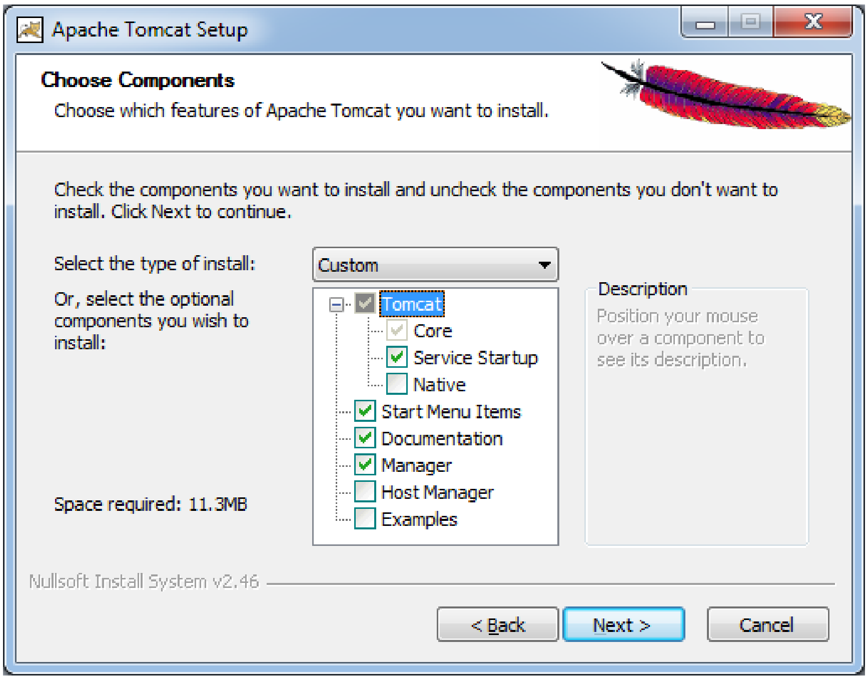
5.2 Installation of the “Apache Tomcat” web server optional

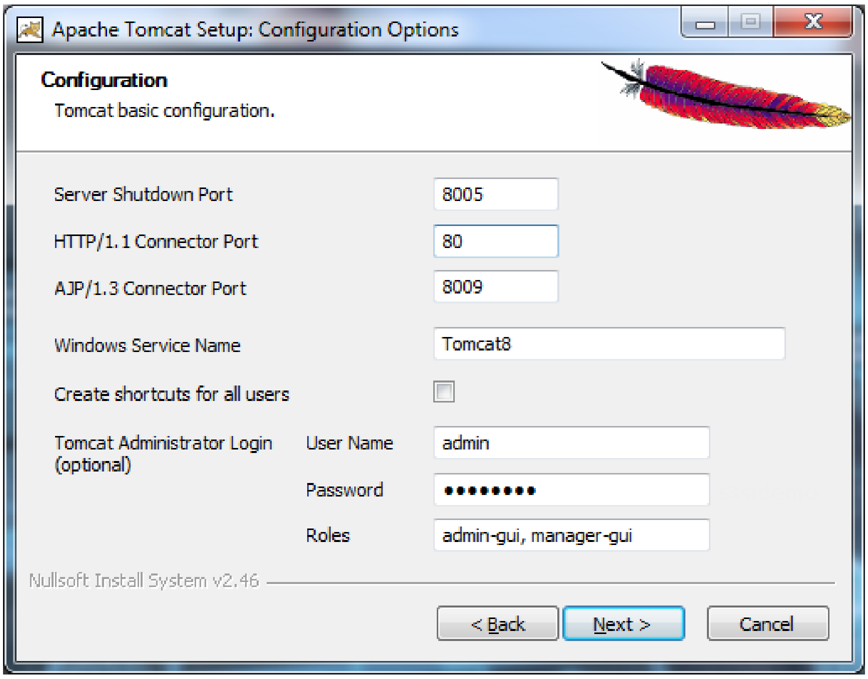
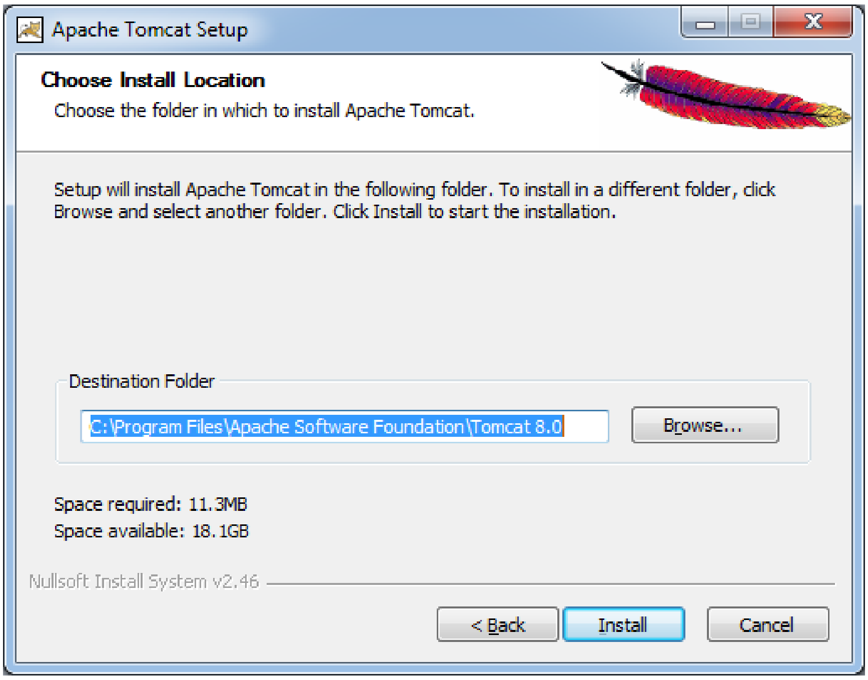
This chapter shows you the steps to be able to work with the “Apache Tomcat” web server.

5.2.1 Installation of the “Apache Tomcat” web server

You can use the “Apache Tomcat” web server for dynamic web applications.

Table 5-1

No.	Action
1.	Open the Apache Tomcat website http://tomcat.apache.org/
2.	Select the "32-bit/64-bit Windows Service Installer” in “Download > Tomcat 8 > Binary Distributions".
3.	Load the executable file “apache-tomcat-8.5.5.exe” onto your device.
4.	Open the storage folder of the downloaded file and start the Apache Tomcat setup. Follow the instructions of the installation setup.
5.	Select the functions of the web server that you want to install by ticking or unticking “Choose Components”.  Apache Tomcat Setup Choose Components Choose which features of Apache Tomcat you want to install. Check the components you want to install and uncheck the components you don't want to install. Click Next to continue. Select the type of install: Custom Or, select the optional components you wish to install: ☑ Tomcat ☑ Core ☑ Service Startup □ Native ☑ Start Menu Items ☑ Documentation ☑ Manager □ Host Manager □ Examples Space required: 11.3MB Description Position your mouse over a component to see its description. Nullsoft Install System v2.46

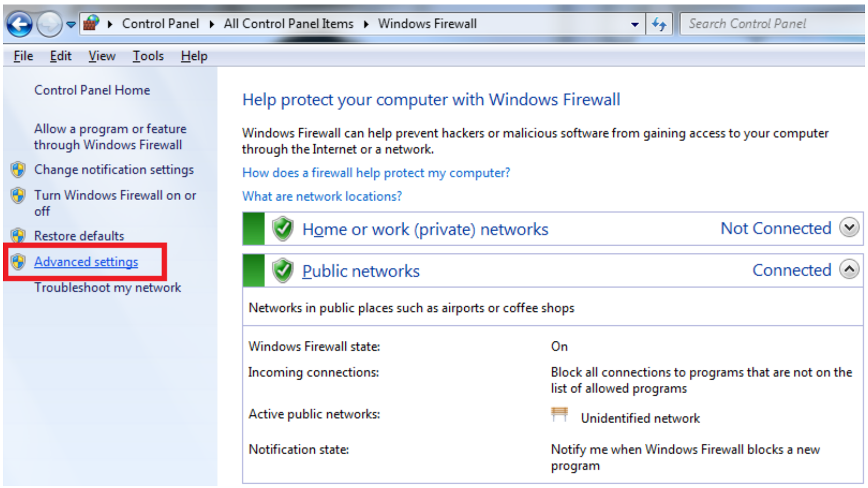
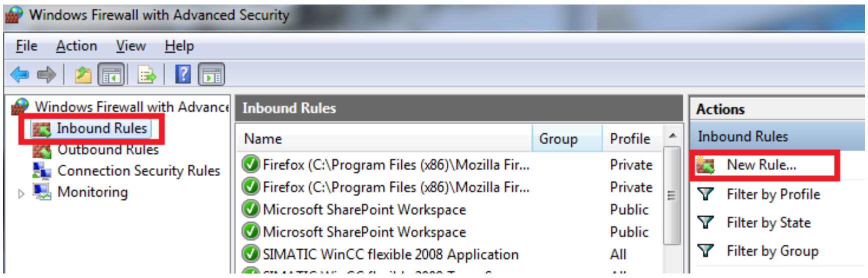
No.	Action
6.	Set the entries in the configuration settings “Configuration Options” as shown in the following figure. Enter a user name and password.  Apache Tomcat Setup: Configuration Options **Configuration** Tomcat basic configuration. Server Shutdown Port: 8005 HTTP/1.1 Connector Port: 80 AJP/1.3 Connector Port: 8009 Windows Service Name: Tomcat8 Create shortcuts for all users: □ Tomcat Administrator Login (optional) User Name: admin Password: ●●●●●●●● Roles: admin-gui, manager-gui Nullsoft Install System v2.46
7.	Select the folder in which you want to install “Apache Tomcat”. Then click the “Install” button.  Apache Tomcat Setup **Choose Install Location** Choose the folder in which to install Apache Tomcat. Setup will install Apache Tomcat in the following folder. To install in a different folder, click Browse and select another folder. Click Install to start the installation. Destination Folder: C:\Program Files\Apache Software Foundation\Tomcat 8.0 Space required: 11.3MB Space available: 18.1GB Nullsoft Install System v2.46

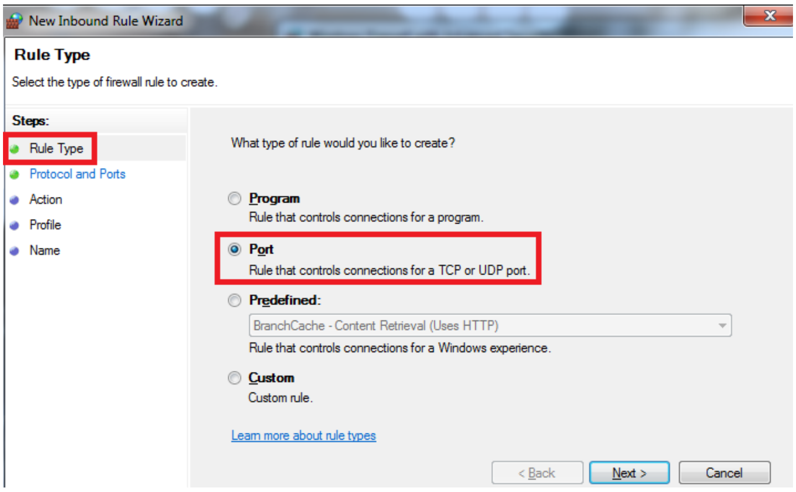
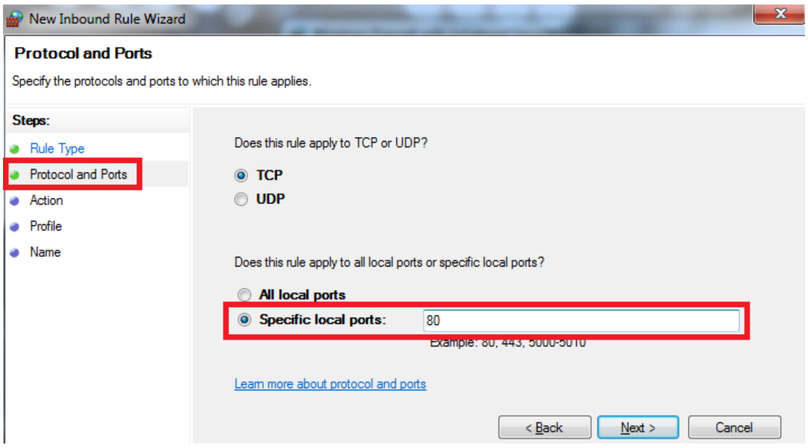
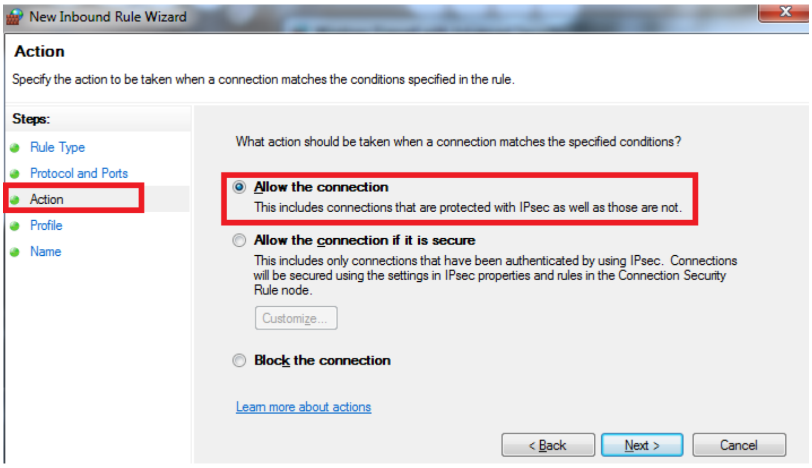
5.2.2 Setting the Windows Firewall

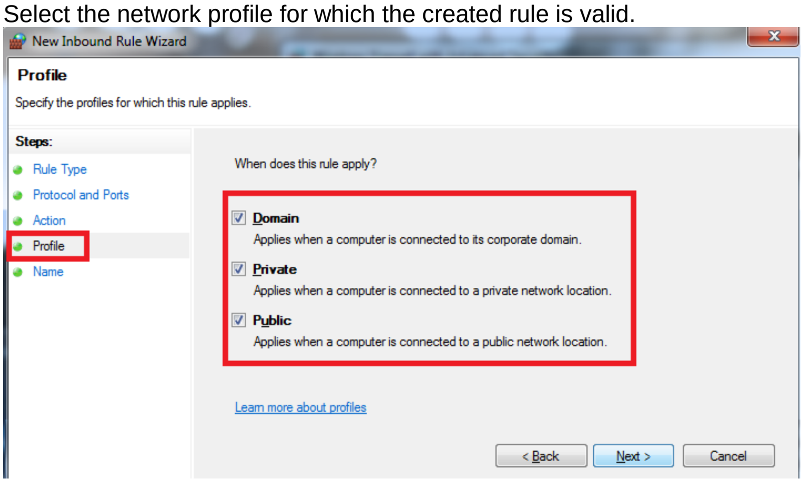
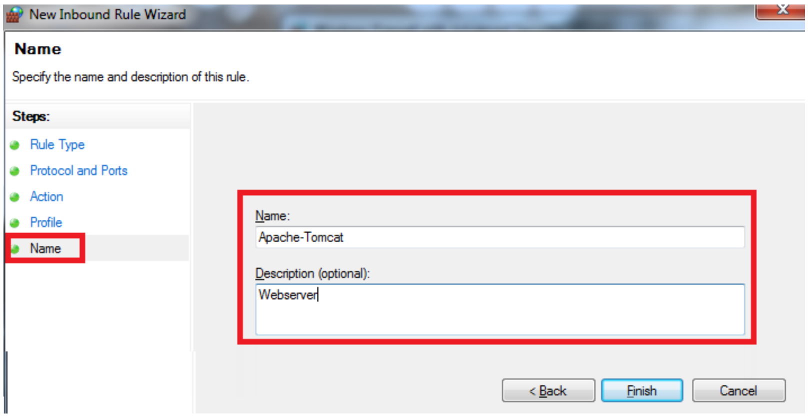
If the network connection between the individual components is blocked by a firewall, there may be problems with the data transmission.

This chapter shows you how you have to configure the firewall to ensure smooth communication between the components.

Table 5-2

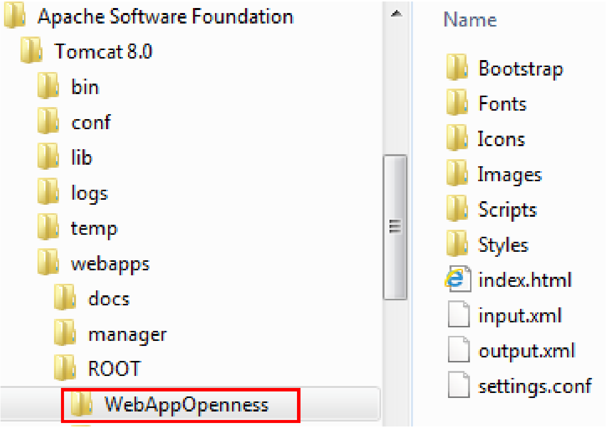
No.	Action
1.	Select "Start > Control Panel > All Control Panel Items > Windows Firewall > Advanced settings".  Control Panel ▸ All Control Panel Items ▸ Windows Firewall Search Control Panel File Edit View Tools Help Control Panel Home Allow a program or feature through Windows Firewall Change notification settings Turn Windows Firewall on or off Restore defaults Advanced settings Troubleshoot my network Help protect your computer with Windows Firewall Windows Firewall can help prevent hackers or malicious software from gaining access to your computer through the Internet or a network. How does a firewall help protect my computer? What are network locations? Home or work (private) networks — Not Connected Public networks — Connected Networks in public places such as airports or coffee shops Windows Firewall state: On Incoming connections: Block all connections to programs that are not on the list of allowed programs Active public networks: Unidentified network Notification state: Notify me when Windows Firewall blocks a new program
2.	First of all create a new “Inbound Rules” in order to open the port for the web server in inbound direction.  Windows Firewall with Advanced Security File Action View Help Windows Firewall with Advance Inbound Rules Outbound Rules Connection Security Rules Monitoring Actions Inbound Rules New Rule... Filter by Profile Filter by State Filter by Group Inbound Rules Name | Group | Profile Firefox (C:\Program Files (x86)\Mozilla Fir... | | Private Firefox (C:\Program Files (x86)\Mozilla Fir... | | Private Microsoft SharePoint Workspace | | Public Microsoft SharePoint Workspace | | Public SIMATIC WinCC flexible 2008 Application | | All

No.	Action
3.	Select the type of rule, by enabling “Port”.  New Inbound Rule Wizard **Rule Type** Select the type of firewall rule to create. Steps: Rule Type Protocol and Ports Action Profile Name What type of rule would you like to create? ○ **Program** Rule that controls connections for a program. ● **Port** Rule that controls connections for a TCP or UDP port. ○ **Predefined:** BranchCache - Content Retrieval (Uses HTTP) Rule that controls connections for a Windows experience. ○ **Custom** Custom rule. Learn more about rule types
4.	Specify the respective port (80) and in addition select the “TCP” protocol.  New Inbound Rule Wizard **Protocol and Ports** Specify the protocols and ports to which this rule applies. Steps: Rule Type Protocol and Ports Action Profile Name Does this rule apply to TCP or UDP? ● **TCP** ○ **UDP** Does this rule apply to all local ports or specific local ports? ○ **All local ports** ● **Specific local ports:** 80 Example: 80, 443, 5000-5010 Learn more about protocol and ports
5.	Enable the “Allow the connection” check box.  New Inbound Rule Wizard **Action** Specify the action to be taken when a connection matches the conditions specified in the rule. Steps: Rule Type Protocol and Ports Action Profile Name What action should be taken when a connection matches the specified conditions? ● **Allow the connection** This includes connections that are protected with IPsec as well as those are not. ○ **Allow the connection if it is secure** This includes only connections that have been authenticated by using IPsec. Connections will be secured using the settings in IPsec properties and rules in the Connection Security Rule node. ○ **Block the connection** Learn more about actions

No.	Action
6.	Select the network profile for which the created rule is valid.  New Inbound Rule Wizard **Profile** Specify the profiles for which this rule applies. When does this rule apply? ☑ **Domain** Applies when a computer is connected to its corporate domain. ☑ **Private** Applies when a computer is connected to a private network location. ☑ **Public** Applies when a computer is connected to a public network location. Steps: Rule Type Protocol and Ports Action Profile Name Learn more about profiles < Back Next > Cancel
7.	Name your rule. The rule is shown in the list of outbound rules.  New Inbound Rule Wizard **Name** Specify the name and description of this rule. Name: Apache-Tomcat Description (optional): Webserver Steps: Rule Type Protocol and Ports Action Profile Name < Back Finish Cancel

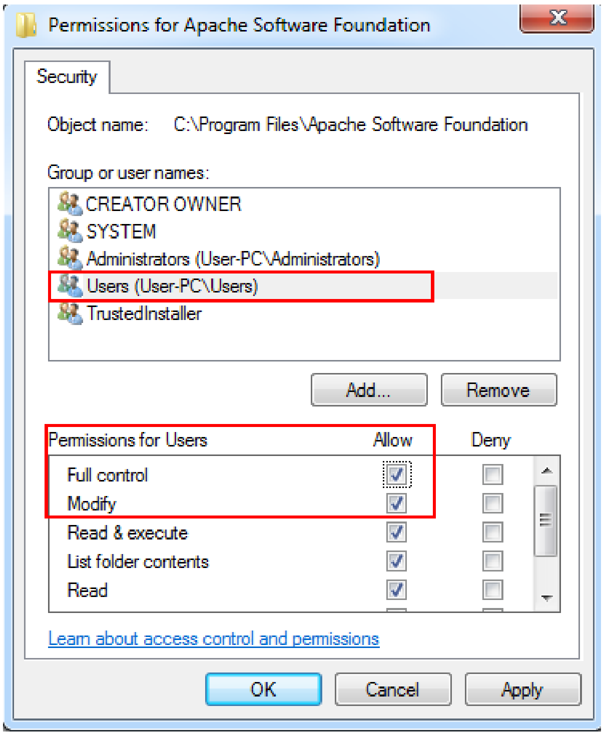
5.2.3 Installing the openness web application „WebAppOpenness“

Table 5-3

No.	Action
1.	Take the “WebAppOpenness” folder from the download file that you have downloaded in chapter 5.1.
2.	Copy the “WebAppOpenness” folder into the following directory: “C: > Program Files > Apache Software Foundation > Tomcat 8.0 > webapps > ROOT”.  Apache Software Foundation Tomcat 8.0 bin conf lib logs temp webapps docs manager ROOT WebAppOpenness Name Bootstrap Fonts Icons Images Scripts Styles index.html input.xml output.xml settings.conf **Note:** The storage path for the “WebAppOpenness” web application is specified in the configuration file “config.xml” from chapter 5.3. If you have not installed “Apache Tomcat” locally, you have to change the link in the “config.xml” accordingly.
3.	Once you have installed “Apache Tomcat” you can start the “WebAppOpenness” web application in your browser. Enter “http://localhost/WebAppOpenness/” in the address line to call the web application.

5.2.4 Security setting in the “Apache Tomcat” directory

Table 5-4

No.	Action
1.	Open the “C:\Program Files” directory.
2.	Right-click the “Apache Software Foundation” folder and select “Properties > Security > Edit”.
3.	For the “Users” tick “Allow” for “Full control” and “Modify”.  Permissions for Apache Software Foundation Security Object name: C:\Program Files\Apache Software Foundation Group or user names: CREATOR OWNER SYSTEM Administrators (User-PC\Administrators) Users (User-PC\Users) TrustedInstaller Permissions for Users — Allow / Deny Full control ☑ □ Modify ☑ □ Read & execute ☑ □ List folder contents ☑ □ Read ☑ □ Learn about access control and permissions

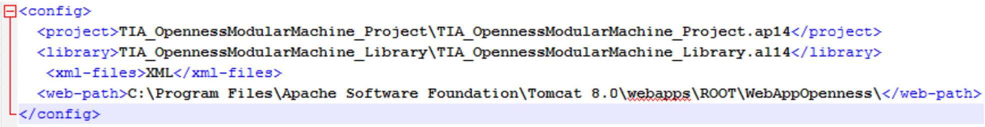
5.3 Creating the configuration file

To determine the path, you can use the configuration file “config.xml” from the download file from chapter 5.1 .

The “config.xml” includes the paths of:

- TIA Portal STEP 7 project
- TIA Portal STEP 7 library
- XML files
- Web application “WebAppOpenness”

Figure 5-1



```
<config>
  <project>TIA_OpennessModularMachine_Project\TIA_OpennessModularMachine_Project.ap14</project>
  <library>TIA_OpennessModularMachine_Library\TIA_OpennessModularMachine_Library.al14</library>
   <xml-files>XML</xml-files>
  <web-path>C:\Program Files\Apache Software Foundation\Tomcat 8.0\webapps\ROOT\WebAppOpenness\</web-path>
</config>
```

Note

> The “config.xml” has to be stored in the same directory as the OpennessApplication “TIA_OpennessModularMachine.exe”.
>
> You have to enter the correct path in “config.xml”, otherwise the OpennessApplication will not work.

5.4 Installation of the OpennessApplication “TIA Openness Project”

Note

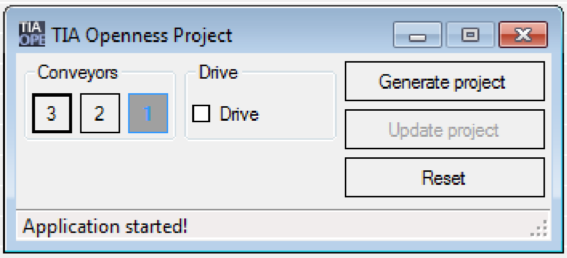
> In STEP 7 V14 or WinCC V14, TIA Portal Openness V14 is included in the delivery.

The Microsoft Visual Studio project with fully programmed TIA Portal Openness functions is available in the download file from chapter 5.1 in the “SourceCode OpennessApp” folder.

The executable file “TIA_OpennessModularMachine.exe” can be found in the download file.

The following user interface “TIA Openness Project” is shown.

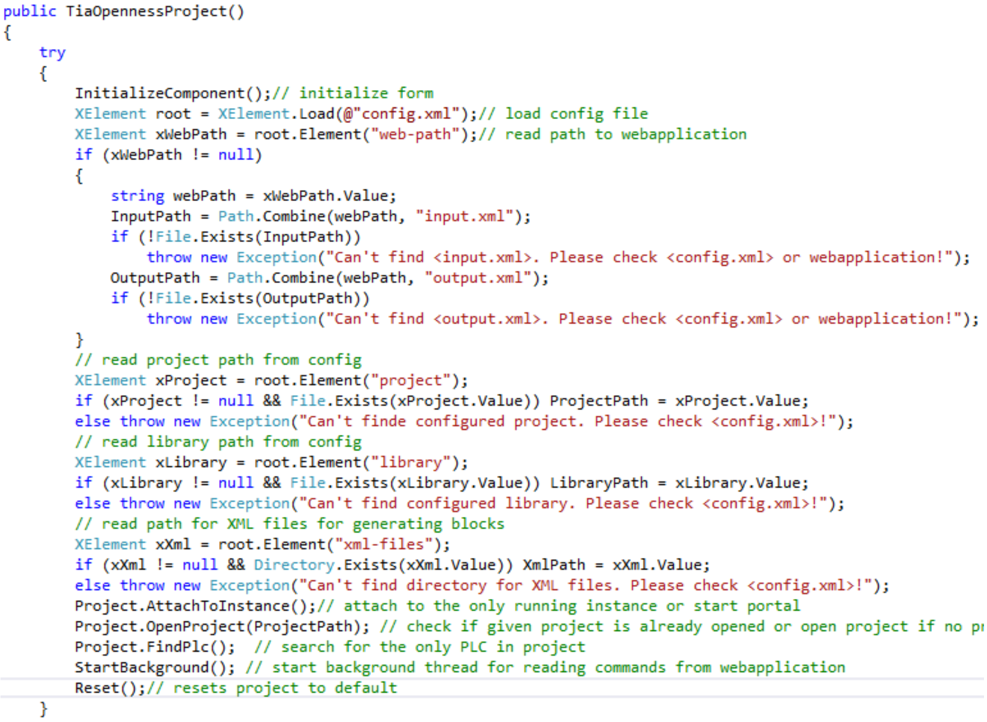
Figure 5-2



Application started!

The “TIA Openness Project” accesses the “config.xml” file to be able to read the path of the initial project, the library and the XML files and calls the TIA Portal project accordingly.

Figure 5-3

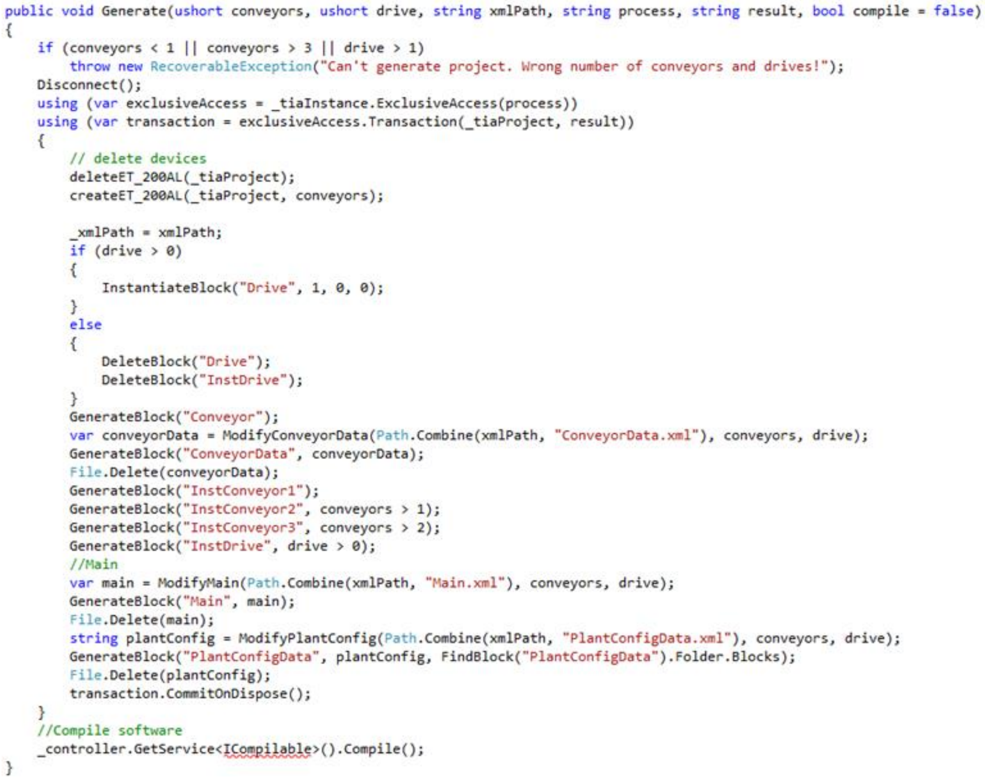


```
public TiaOpennessProject()
{
    try
    {
        InitializeComponent();// initialize form
        XElement root = XElement.Load(@"config.xml");// load config file
        XElement xWebPath = root.Element("web-path");// read path to webapplication
        if (xWebPath != null)
        {
            string webPath = xWebPath.Value;
            InputPath = Path.Combine(webPath, "input.xml");
            if (!File.Exists(InputPath))
                throw new Exception("Can't find <input.xml>. Please check <config.xml> or webapplication!");
            OutputPath = Path.Combine(webPath, "output.xml");
            if (!File.Exists(OutputPath))
                throw new Exception("Can't find <output.xml>. Please check <config.xml> or webapplication!");
        }
        // read project path from config
        XElement xProject = root.Element("project");
        if (xProject != null && File.Exists(xProject.Value)) ProjectPath = xProject.Value;
        else throw new Exception("Can't finde configured project. Please check <config.xml>!");
        // read library path from config
        XElement xLibrary = root.Element("library");
        if (xLibrary != null && File.Exists(xLibrary.Value)) LibraryPath = xLibrary.Value;
        else throw new Exception("Can't find configured library. Please check <config.xml>!");
        // read path for XML files for generating blocks
        XElement xXml = root.Element("xml-files");
        if (xXml != null && Directory.Exists(xXml.Value)) XmlPath = xXml.Value;
        else throw new Exception("Can't find directory for XML files. Please check <config.xml>!");
        Project.AttachToInstance();// attach to the only running instance or start portal
        Project.OpenProject(ProjectPath); // check if given project is already opened or open project if no pr
        Project.FindPlc();  // search for the only PLC in project
        StartBackground(); // start background thread for reading commands from webapplication
        Reset();// resets project to default
    }
```

The project is updated by the “TIA_OpennessModularMachine_Library” library (update project)

The configuration data of the machine variants are imported from the XML folder via XML files in the TIA Portal. Afterwards the program blocks are generated and configured.

Figure 5-4

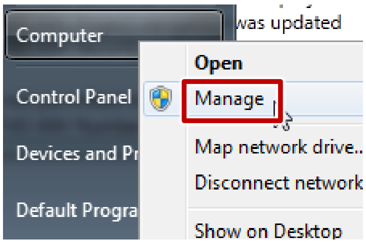
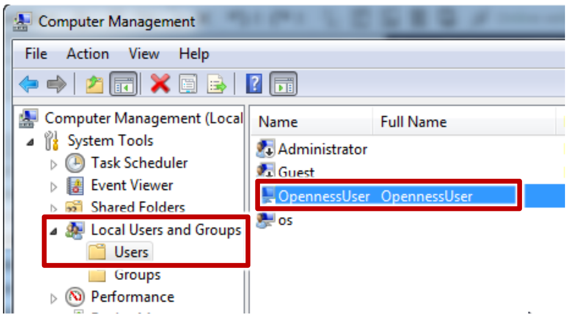
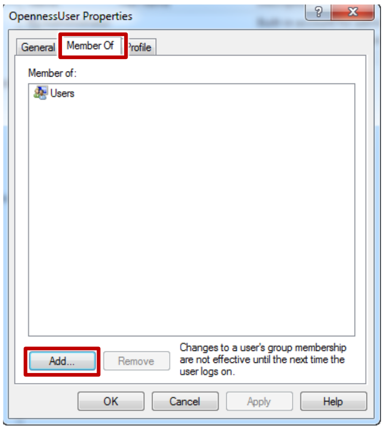


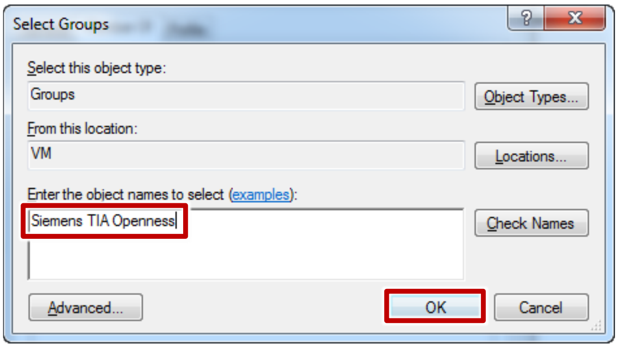
```
public void Generate(ushort conveyors, ushort drive, string xmlPath, string process, string result, bool compile = false)
{
    if (conveyors < 1 || conveyors > 3 || drive > 1)
        throw new RecoverableException("Can't generate project. Wrong number of conveyors and drives!");
    Disconnect();
    using (var exclusiveAccess = _tiaInstance.ExclusiveAccess(process))
    using (var transaction = exclusiveAccess.Transaction(_tiaProject, result))
    {
        // delete devices
        deleteET_200AL(_tiaProject);
        createET_200AL(_tiaProject, conveyors);

        _xmlPath = xmlPath;
        if (drive > 0)
        {
            InstantiateBlock("Drive", 1, 0, 0);
        }
        else
        {
            DeleteBlock("Drive");
            DeleteBlock("InstDrive");
        }
        GenerateBlock("Conveyor");
        var conveyorData = ModifyConveyorData(Path.Combine(xmlPath, "ConveyorData.xml"), conveyors, drive);
        GenerateBlock("ConveyorData", conveyorData);
        File.Delete(conveyorData);
        GenerateBlock("InstConveyor1");
        GenerateBlock("InstConveyor2", conveyors > 1);
        GenerateBlock("InstConveyor3", conveyors > 2);
        GenerateBlock("InstDrive", drive > 0);
        //Main
        var main = ModifyMain(Path.Combine(xmlPath, "Main.xml"), conveyors, drive);
        GenerateBlock("Main", main);
        File.Delete(main);
        string plantConfig = ModifyPlantConfig(Path.Combine(xmlPath, "PlantConfigData.xml"), conveyors, drive);
        GenerateBlock("PlantConfigData", plantConfig, FindBlock("PlantConfigData").Folder.Blocks);
        File.Delete(plantConfig);
        transaction.CommitOnDispose();
    }
    //Compile software
    _controller.GetService<ICompilable>().Compile();
}
```

5.5 Managing the user rights

In order to use the application example, the user has to be added to the “Siemens TIA Openness” user group.

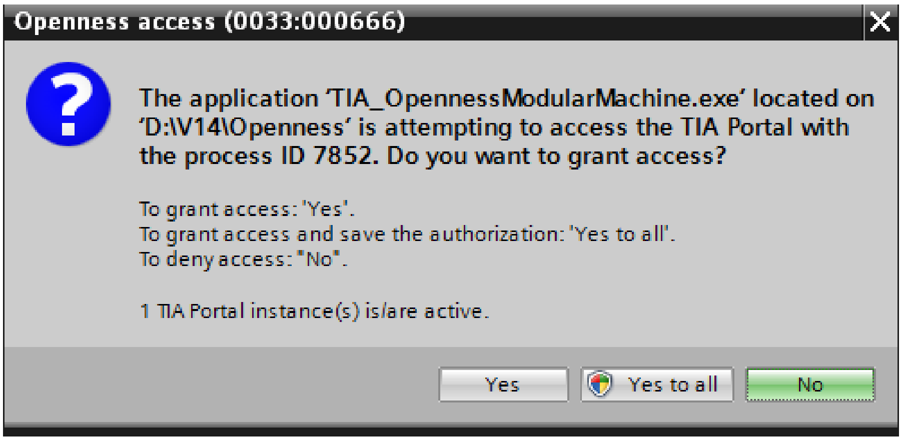
Table 5-5

No.	Action
1.	Right-click on “Computer” in the windows taskbar and select “Manage”.  was updated Computer Control Panel Devices and Pr Default Progra Open Manage Map network drive.. Disconnect network Show on Desktop
2.	Open “Local Users and Groups > Users” and double click the user.  Computer Management File Action View Help Computer Management (Local System Tools Task Scheduler Event Viewer Shared Folders Local Users and Groups Users Groups Performance Name Full Name Administrator Guest OpennessUser OpennessUser os
3.	Go to the “Member Of” tab and click the “Add” button.  OpennessUser Properties General Member Of Profile Member of: Users Add... Remove OK Cancel Apply Help Changes to a user's group membership are not effective until the next time the user logs on.

No.	Action
4.	Enter “Siemens TIA Openness” and confirm by pressing “OK”.  Select Groups Select this object type: Groups From this location: VM Enter the object names to select (examples): Siemens TIA Openness Object Types... Locations... Check Names Advanced... OK Cancel
5.	Close the opened dialog boxes and log in again. **Note:** When the dialog is not closed and is reopened, the changes are not validly accepted.

5.6 Granting access

Table 5-6

No.	Action
1.	Double-click on the “TIA_OpennessModularMachine.exe” application. When you start the application for the first time, the following security message appears.  Openness access (0033:000666) The application ‘TIA_OpennessModularMachine.exe’ located on ‘D:\V14\Openness’ is attempting to access the TIA Portal with the process ID 7852. Do you want to grant access? To grant access: 'Yes'. To grant access and save the authorization: 'Yes to all'. To deny access: "No". 1 TIA Portal instance(s) is/are active.
2.	Confirm the message with “Yes” to allow the access once and with “Yes to all” to always permit access to this application.

Note

If you are working with Microsoft Visual Studio, it can happen that you will receive the message, although you have already clicked “Yes to all”. To avoid this, follow the instructions in the FAQ “Tips for TIA Portal Openness”.
https://support.industry.siemens.com/cs/ww/en/view/109251656

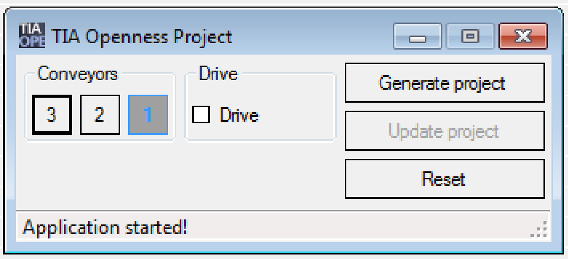
6 Operating the Application Example

At this point, it is assumed that the necessary software from chapter 5.1 has been installed on your computer and that you are familiar with the software.

To start “OpennessApplication”, double-click on the following “TIA_OpennessModularMachine.exe” file.

First of all a TIA Portal instance is opened and then a default project. The following user interface “TIA Openness Project” is shown.

Figure 6-1

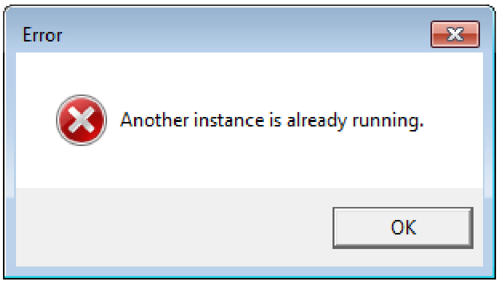


Application started!

Note

When the “TIA_OpennessModularMachine.exe” is running in the background it cannot be restarted. The following error message is displayed “Another instance is already running”.

Figure 6-2



Another instance is already running.

You have two options to operate the application:

- Scenario A: Operation via the user interface “TIA Openness Project”
- Scenario B: Operation via the “WebAppOpenness” web browser

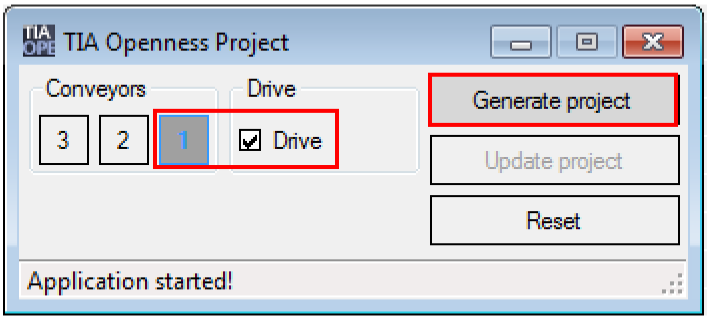
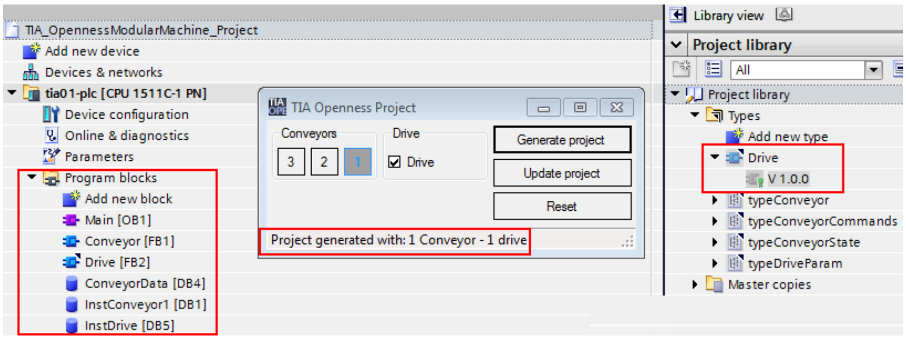
6.1 Scenario A

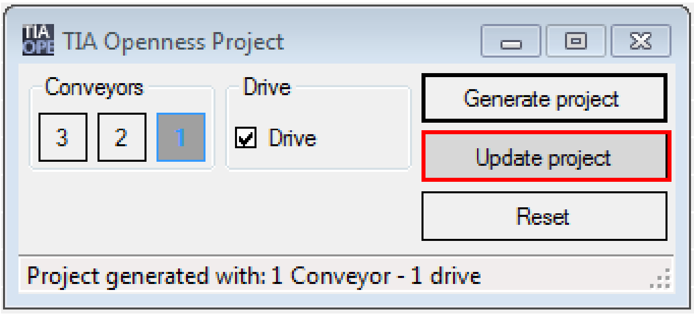
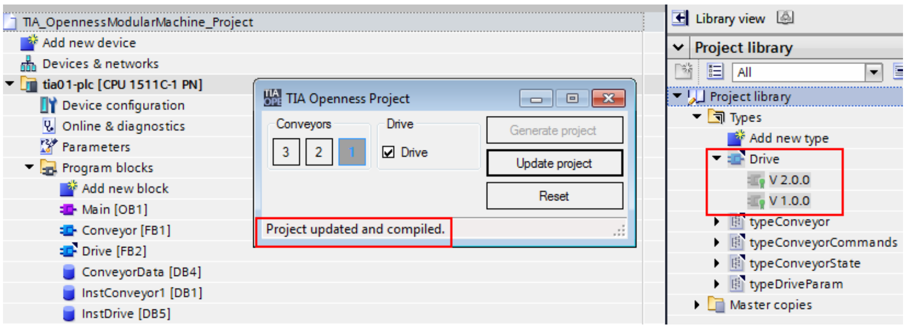
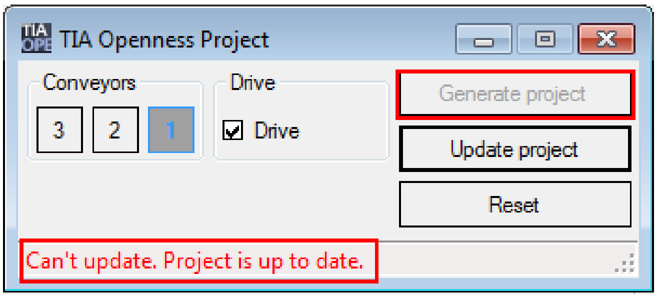
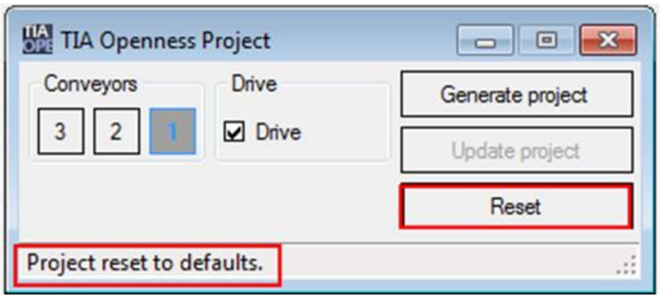
In scenario A you operate the application via the user interface “TIA Openness Project”.

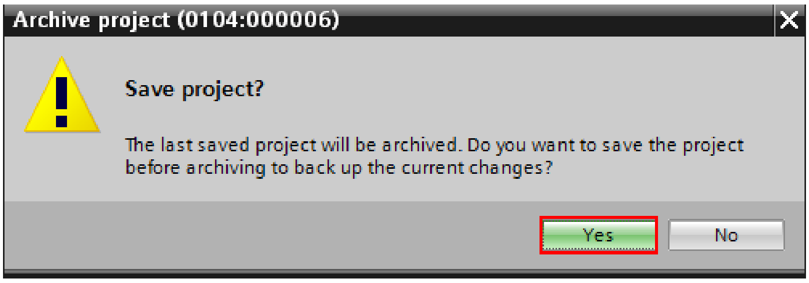
Note

In order to save your changes permanently, you have to archive the project. This is also valid when the project is saved under a different name, since the project is reset to default after closing.

Table 6-1

No.	Action
1.	Select the desired variant of your machine and then click on “Generate project”. In the example variant 1 with drive (Conveyor & Drive) is shown.  TIA Openness Project Conveyors Drive 3 2 1 ☑ Drive Generate project Update project Reset Application started!
2.	As initial configuration, the hardware in the plant is setup according to variant 1. The program blocks of the selected modules are generated (generating) and the configuration is compiled (Compiling). The version of the drive (Drive V1.0.0) can be seen in the project library.  TIA_OpennessModularMachine_Project Add new device Devices & networks tia01-plc [CPU 1511C-1 PN] Device configuration Online & diagnostics Parameters Program blocks Add new block Main [OB1] Conveyor [FB1] Drive [FB2] ConveyorData [DB4] InstConveyor1 [DB1] InstDrive [DB5] TIA Openness Project Conveyors Drive 3 2 1 ☑ Drive Generate project Update project Reset Library view Project library All Project library Types Add new type Drive V 1.0.0 typeConveyor typeConveyorCommands typeConveyorState typeDriveParam Master copies Project generated with: 1 Conveyor - 1 drive The variant of the machine can be changed as often as desired when you click onto the desired variant and the “Generate project” button.

No.	Action
3.	To update the variant of your machine, click on “Update project”. The libraries are loaded and the project is updated.  TIA Openness Project Conveyors 3 2 1 Drive ☑ Drive Generate project Update project Reset Project generated with: 1 Conveyor - 1 drive
4.	The update of the project can be seen based on the version of the drive (Drive V 2.0.0) in the project library.  TIA_OpennessModularMachine_Project Add new device Devices & networks tia01-plc [CPU 1511C-1 PN] Device configuration Online & diagnostics Parameters Program blocks Add new block Main [OB1] Conveyor [FB1] Drive [FB2] ConveyorData [DB4] InstConveyor1 [DB1] InstDrive [DB5] TIA Openness Project Conveyors 3 2 1 Drive ☑ Drive Generate project Update project Reset Library view Project library All Project library Types Add new type Drive V 2.0.0 V 1.0.0 typeConveyor typeConveyorCommands typeConveyorState typeDriveParam Master copies Project updated and compiled.
5.	After updating the project (Update project) no generation or renewed update is possible.  TIA Openness Project Conveyors 3 2 1 Drive ☑ Drive Generate project Update project Reset Can't update. Project is up to date.
6.	Reset the project to default to be able to allow another generation. Click the “Reset” box. After the reset, the following message appears: “Project reset to defaults”.  TIA Openness Project Conveyors 3 2 1 Drive ☑ Drive Generate project Update project Reset Project reset to defaults.

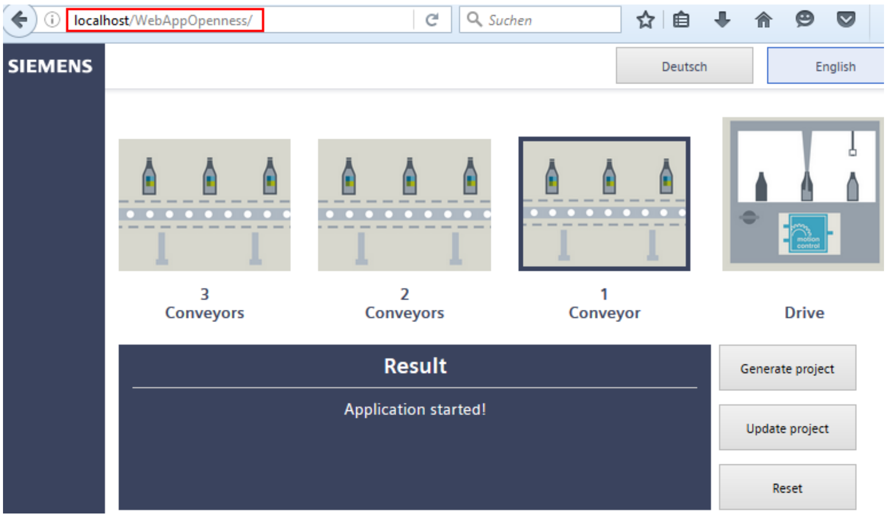
No.	Action
7.	In order to save your changes, you have to archive the project, by confirming the following message.  Archive project (0104:000006) Save project? The last saved project will be archived. Do you want to save the project before archiving to back up the current changes?

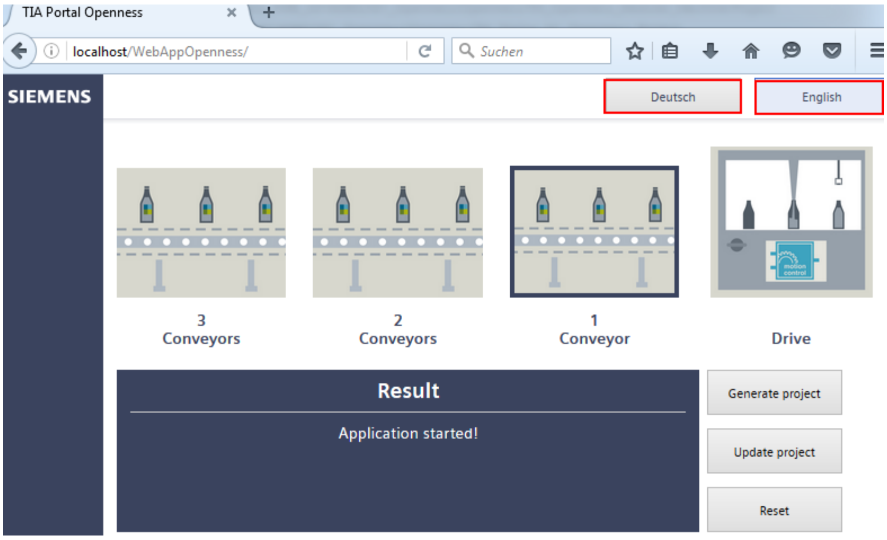
6.2 Scenario B

In scenario B you operate the application via the “WebAppOpenness” web browser.

Note In order to save your changes permanently, you have to archive the project. This is also valid when the project is saved under a different name, since the project is reset to default after closing.

Table 6-2

No.	Action
1.	Start the “WebAppOpenness” web application in a browser. In this application example the “Firefox” browser it is shown.
2.	Enter “http://localhost/WebAppOpenness/” in the address line and the web application is accessed.  localhost/WebAppOpenness/ Suchen SIEMENS Deutsch English 3 Conveyors 2 Conveyors 1 Conveyor motion control Drive Generate project Update project Reset Result Application started! **Note:** The web application interface can **only** be operated, when “TIA_OpennessModularMachine.exe” is running in the background and a default project has been created.

No.	Action
3.	Click on German or English, in order to operate the web application interface in the selected language.  TIA Portal Openness localhost/WebAppOpenness/ Suchen SIEMENS Deutsch English 3 Conveyors 2 Conveyors 1 Conveyor Drive Result Application started! Generate project Update project Reset
4.	The operation of the web application interface “WebAppOpenness” is identical with the operation of the “TIA Openness Project” user interface. See chapter 6.1.

7 Links & Literature

Table 7-1

	Topic
\1\	Siemens Industry Online Support https://support.industry.siemens.com
\2\	Download page of the entry https://support.industry.siemens.com/cs/ww/en/view/109739678
\3\	Technical Support http://www.siemens.com/automation/support-request
\4\	TIA Selection Tool http://www.siemens.com/tia-selection-tool
\5\	Automating SIMATIC projects via scripts https://support.industry.siemens.com/cs/de/en/view/109477163/72488745483
\6\	TIA Portal Openness: Introduction and Demo Application https://support.industry.siemens.com/cs/de/en/view/108716692
\7\	Variants Management with SIMATIC S7 https://support.industry.siemens.com/cs/ww/en/view/29430270
\8\	Apache Tomcat http://tomcat.apache.org/
\9\	Tips for TIA Portal Openness https://support.industry.siemens.com/cs/en/en/view/109251656

8 History

Table 8-1

Version	Date	Modifications
V1.0	01/2017	First version
V1.0	07/2017	Project upgrade from TIA Portal V14 to V14 SP1