

Industry Online Support

Home

TIA Add-Ins Getting Started

TIA Portal as of V16 / SINAMICS Startdrive as of V16

<https://support.industry.siemens.com/cs/ww/en/view/109779415>

Siemens In-
dustry
Online Sup-
port



Legal information

Use of application examples

Application examples illustrate the solution of automation tasks through an interaction of several components in the form of text, graphics and/or software modules. The application examples are a free service by Siemens AG and/or a subsidiary of Siemens AG ("Siemens"). They are non-binding and make no claim to completeness or functionality regarding configuration and equipment. The application examples merely offer help with typical tasks; they do not constitute customer-specific solutions. You yourself are responsible for the proper and safe operation of the products in accordance with applicable regulations and must also check the function of the respective application example and customize it for your system.

Siemens grants you the non-exclusive, non-sublicensable and non-transferable right to have the application examples used by technically trained personnel. Any change to the application examples is your responsibility. Sharing the application examples with third parties or copying the application examples or excerpts thereof is permitted only in combination with your own products. The application examples are not required to undergo the customary tests and quality inspections of a chargeable product; they may have functional and performance defects as well as errors. It is your responsibility to use them in such a manner that any malfunctions that may occur do not result in property damage or injury to persons.

Disclaimer of liability

Siemens shall not assume any liability, for any legal reason whatsoever, including, without limitation, liability for the usability, availability, completeness and freedom from defects of the application examples as well as for related information, configuration and performance data and any damage caused thereby. This shall not apply in cases of mandatory liability, for example under the German Product Liability Act, or in cases of intent, gross negligence, or culpable loss of life, bodily injury or damage to health, non-compliance with a guarantee, fraudulent non-disclosure of a defect, or culpable breach of material contractual obligations. Claims for damages arising from a breach of material contractual obligations shall however be limited to the foreseeable damage typical of the type of agreement, unless liability arises from intent or gross negligence or is based on loss of life, bodily injury or damage to health. The foregoing provisions do not imply any change in the burden of proof to your detriment. You shall indemnify Siemens against existing or future claims of third parties in this connection except where Siemens is mandatorily liable.

By using the application examples you acknowledge that Siemens cannot be held liable for any damage beyond the liability provisions described.

Other information

Siemens reserves the right to make changes to the application examples at any time without notice. In case of discrepancies between the suggestions in the application examples and other Siemens publications such as catalogs, the content of the other documentation shall have precedence.

The Siemens terms of use (<https://support.industry.siemens.com>) shall also apply.

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 constitute one element of such a concept.

Customers are responsible for preventing unauthorized access to their plants, systems, machines and networks. Such systems, machines and components should only be connected to an enterprise network or the Internet if and to the extent such a connection is necessary and only when appropriate security measures (e.g. firewalls and/or network segmentation) are in place.

For additional information on industrial security measures that may be implemented, please visit <https://www.siemens.com/industrialsecurity>.

Siemens' products and solutions undergo continuous development to make them more secure. Siemens strongly recommends that product updates are applied as soon as they are available and that the latest product versions are used. Use of product versions that are no longer supported, and failure to apply the latest updates may increase customer's exposure to cyber threats. To stay informed about product updates, subscribe to the Siemens Industrial Security RSS Feed at: <https://www.siemens.com/industrialsecurity>.

Contents

Legal information	2
1 Definition of TIA Add-Ins.....	4
1.1 Fundamentals of TIA Portal Openness	4
1.2 Fundamentals of TIA Add-Ins	4
1.3 Debug TIA Add-Ins.....	5
1.4 Content and structure of Getting Started	5
2 Development preconditions	8
2.1 Software preconditions.....	8
2.2 Licenses	8
3 Creating the Visual Studio project.....	10
3.1 Creating a new project	10
3.2 Creating class "AddInProvider"	15
3.3 Programming class "AddIn".....	21
3.4 Creating config.xml.....	25
3.5 Creating the .addin file	29
3.6 Installing the .addin file in the TIA Portal.....	31
3.7 Joining the "TIA Portal Openness" user group.....	33
3.8 Activating and running the Add-In in the TIA Portal	36
3.9 Note on the template of the current status	39
4 Startdrive Openness – interconnecting drive parameters	40
5 Design template for a user interface	41
5.1 Template of a window	41
5.2 Using the template	52
5.3 Creating the .addin file for TIA Portal	54
6 Additional reference recommendation.....	55
6.1 TIA Portal Openness Manual	55
6.2 Programming support TIA Portal Openness Explorer.....	55
6.3 Example of a more comprehensive TIA Add-In	55
7 FAQ	56
7.1 Why can I not access a certain SINAMICS parameter?	56
8 Appendix	57
8.1 Application support.....	57
8.2 Links and references	57
8.3 History	58

1 Definition of TIA Add-Ins

1.1 Fundamentals of TIA Portal Openness

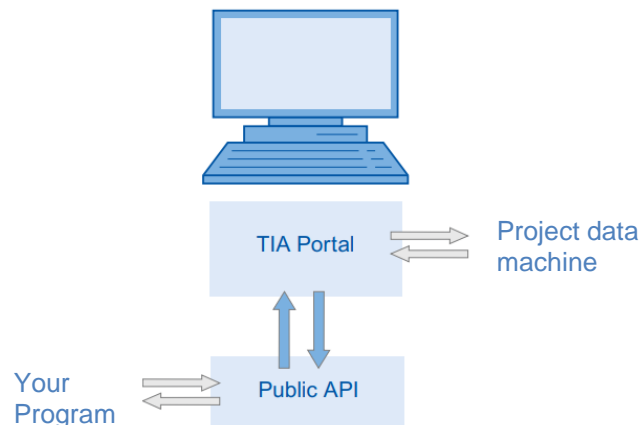
Steps that are repeated in TIA Portal projects can be automated via the TIA Portal Openness API, which saves a lot of time. TIA Portal Openness allows functions to be selected for defined tasks in the TIA Portal, e.g. opening projects, adapting PLC data and changing drive parameters. This means that TIA Portal Openness allows external applications to access the TIA Portal and its various functions.

The following TIA Portal areas can be accessed via the TIA Portal Openness API interface:

- Portal data
- Project data
- PLC data
- HMI data
- Drive data

How the TIA Portal can be accessed via the TIA Portal Openness API interface is schematically shown in Fig. 1-1.

Fig. 1-1 Description of the TIA Portal Openness

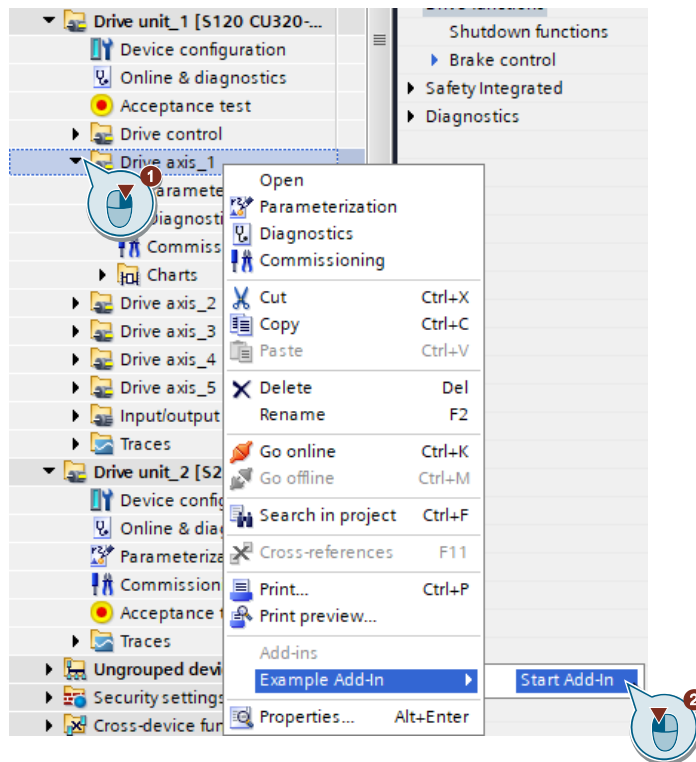


1.2 Fundamentals of TIA Add-Ins

TIA Add-Ins provide a user-friendly way of extending the TIA Portal functionality via Openness API. TIA Add-Ins use the Openness interface to control TIA Portal functions, and therefore automate steps that are repeated. No additional external applications are required to run TIA Add-Ins, as the Add-Ins can be directly started on the required elements in the TIA Portal.

As shown in Fig. 1-2, a TIA Add-In can be directly started on a drive axis for example, by simply right clicking on the appropriate axis. The Add-In that can be run is displayed in the shortcut menu under "Add-Ins". The Add-In can be started by left clicking on "Start TIA Add-In".

Fig. 1-2 Example: Starting a TIA Add-In



This means that TIA Add-Ins provide a high level of user friendliness as external applications do not have to be called.

1.3 Debug TIA Add-Ins

Information about debugging a TIA Add-In can be found in the following SIOS-entry:

<https://support.industry.siemens.com/cs/ww/en/view/109790393>

1.4 Content and structure of Getting Started

This description is intended to make it easy to become familiar with programming TIA Add-Ins. Users are navigated in a user-friendly fashion through the workflow of creating a TIA Add-In based on step-by-step instructions with all the relevant mouse clicks. The necessary settings are explained, based on program examples.

This description has been created so that even users with little programming experience can simply create Add-Ins.

If complex applications that go beyond the information provided in this description are to be created, then you should be familiar with C#/VB.NET or the Openness API.

A summary of the content of Getting Started is provided in the following table.

Table 1-1 Content and structure of Getting Started

Chapter	Content
Chapter 1	The purpose of TIA Portal Openness and the structure of Getting Started are explained.
1.1 Fundamentals of TIA Portal Openness	The purpose of TIA Portal Openness is specified.
1.2 Fundamentals of TIA Add-Ins	The interrelationship between TIA Portal Openness and TIA Add-Ins is explained as well as the advantages of TIA Add-Ins.
1.4 Content and structure of Getting Started	This chapter describes the content and structure of the document.
Chapter 2	All the preconditions required for programming TIA Add-Ins is specified.
2.1 Software preconditions	The software preconditions required for programming TIA Add-Ins is specified.
2.2 Licenses	The licenses that are required for programming and running Add-Ins are specified here.
Chapter 3	Step-by-step instructions for creating a TIA Add-In.
3.1 Creating a new project	How a Visual Studio project is created from scratch to program a TIA Add-In is described here.
3.2 Creating class "AddInProvider"	Class "AddInProvider" is used to define in which area of the TIA Portal the Add-In is displayed. The program code of the class is specified and explained.
3.3 Programming class "AddIn"	Class "AddIn" includes the structure of the Add-In, i.e. whether a button/radio button/checkbox can be selected in the TIA Portal under the "Add-Ins" tab. This class also includes the Add-In code to be executed, i.e. the functions that you wish to automate in the TIA Portal. The program code of the class is specified and explained.
3.4 Creating config.xml	To create the .addin file, which then must be subsequently saved to the TIA Portal installation folder, a "Config.xml" configuration file must be created in the Visual Studio project. This XML file can be considered to be a type of construction plan. Information is saved here, for example, a brief description of the TIA Add-In, the version of the TIA Add-In and/or the name of the Add-In author. Special rights for the Add-In sequence are also specified in this configuration file. For example, whether the TIA Portal can only be read-accessed or also write-accessed. The program code of the class is specified and explained.
3.5 Creating the .addin file	Here, additional settings are specified in the Visual Studio project, which are necessary to generate an .addin file from the Visual Studio Solution.
3.6 Installing the .addin file in the TIA Portal	How the generated .addin file must be installed is specified here.
3.7 Joining the "TIA Portal Openness" user group	To run TIA Add-Ins, you must be a member of the Windows "TIA Portal Openness" user group. An appropriate instruction is provided.
3.8 Activating and running the Add-In in the TIA Portal	How the TIA Add-In can be activated in the TIA Portal and subsequently started is explained here.

1 Definition of TIA Add-Ins

Chapter	Content
3.9 Note on the template of the current status	The Visual Studio project created from Chapter 3 is archived and made available so that the steps outlined in the chapter do not have to be manually performed. It serves as template for developing your own Add-Ins.
Chapter 4	In this chapter, a reference to the second document in the article is made. This document
Chapter 5	A WPF design template is specified for creating a window that looks like TIA Portal
5.1 Template of a window	The XAML code of the WPF is specified here.
5.2 Using the template	Using the template to interconnect drive parameters with subsequent output of a log text is explained.
Chapter 6	Useful references that extend this Getting Started are specified here. This document deals with Openness functions, which can be used for SINAMICS drives.
6.1 TIA Portal Openness Manual	The TIA Portal Openness Manual contains all TIA Portal Openness functions.
6.2 Programming support TIA Portal Openness Explorer	The TIA Portal Openness Explorer simplifies programming as the Openness object model of a TIA project can be read-out here in a user-friendly fashion.
6.3 Example of a more comprehensive TIA Add-In	Reference is made here to a larger TIA Add-In, for which the Visual Studio project is also published.
Chapter 7	FAQs are listed here
7.1 Why can I not access a certain SINAMICS parameter?	TIA Portal Openness can only access SINAMICS parameters that are in a white list.

2 Development preconditions

2.1 Software preconditions

The following development environment is required when programming TIA Add-Ins:

	Microsoft Visual Studio 2015 Update 1 or higher	Microsoft Visual Studio 2017 or higher	.Net Framework As of 4.6 to 4.8	.Net Framework 4.8
TIA Portal Openness V16	✓	✓	✓	
TIA Portal Openness V17		✓		✓

Add-Ins which were programmed with TIA Portal Openness V16 also run in the V17 of the TIA Portal. However, there is no downward compatibility with Add-Ins programmed with TIA Portal Openness V17.

To run TIA Add-Ins (with Startdrive functionality), the following software components must be installed in the TIA Portal:

	STEP 7	Startdrive	TIA Portal Openness
	As of V16	As of V16	As of V16
TIA Portal	✓	✓	✓

2.2 Licenses

- No special Siemens licenses are required for the "pure" execution of TIA Add-Ins in the TIA Portal.
- If you wish to develop your own TIA Add-Ins, then a Microsoft license is required for the Visual Studio programming environment.

Note

The usual TIA Portal engineering licenses (STEP 7 Professional / STEP 7 Safety/ Startdrive Advanced / DCC Combo / etc.) must also be taken into account when running the Add-Ins. This means that it is not possible to use Add-Ins without having these licenses.

3 Creating the Visual Studio project

Note

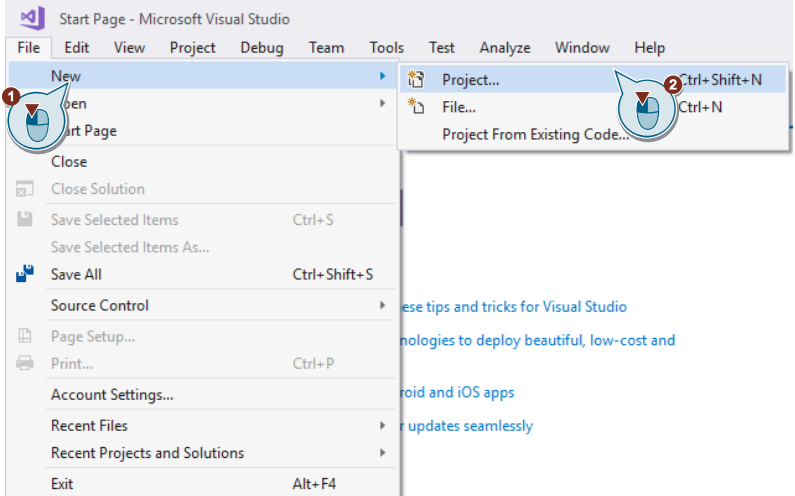
The following instructions show how to create a new Visual Studio project step by step to program a TIA Add-In. At the end of the chapter, the Visual Studio project created in Chapter 3 is archived, so that theoretically, the subsequent steps do not have to be independently carried out. More detailed information is provided in Chapter 3.9 Note on the template of the current status.

3.1 Creating a new project

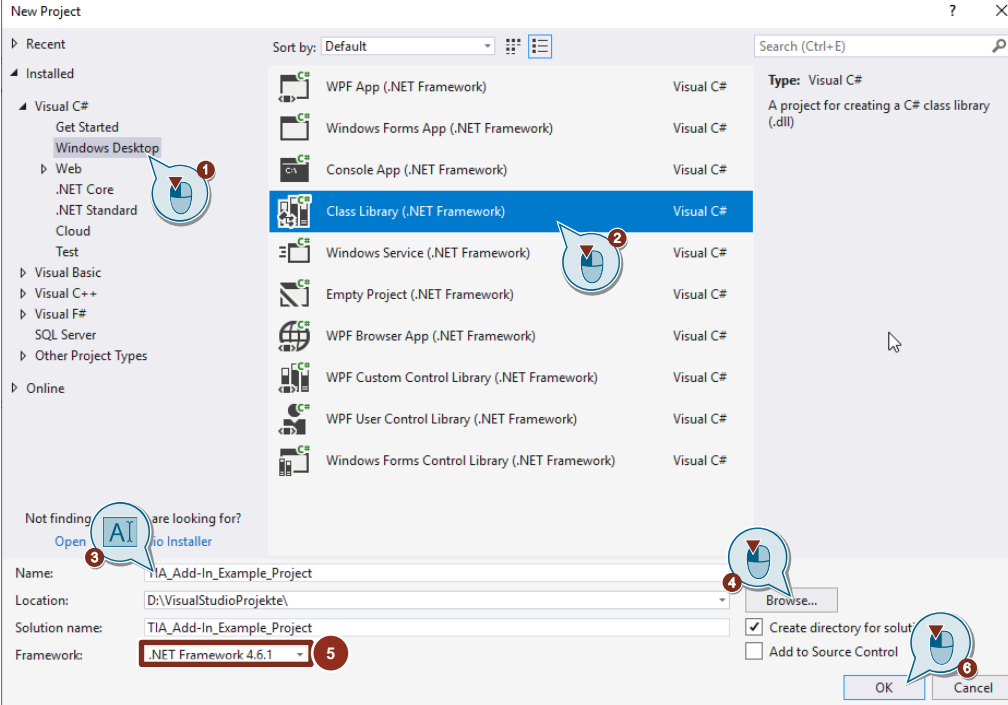
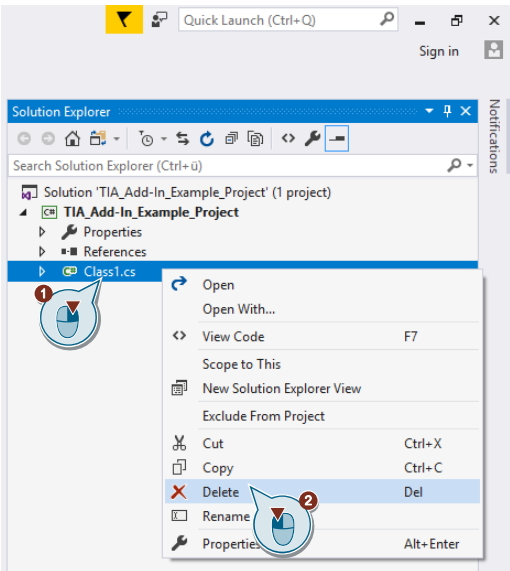
Note

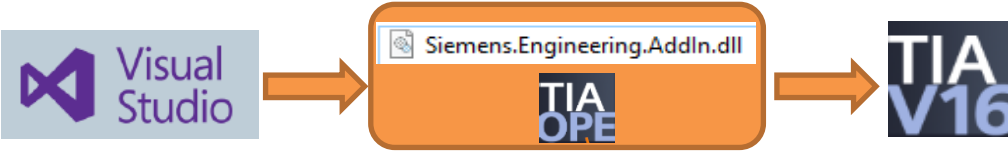
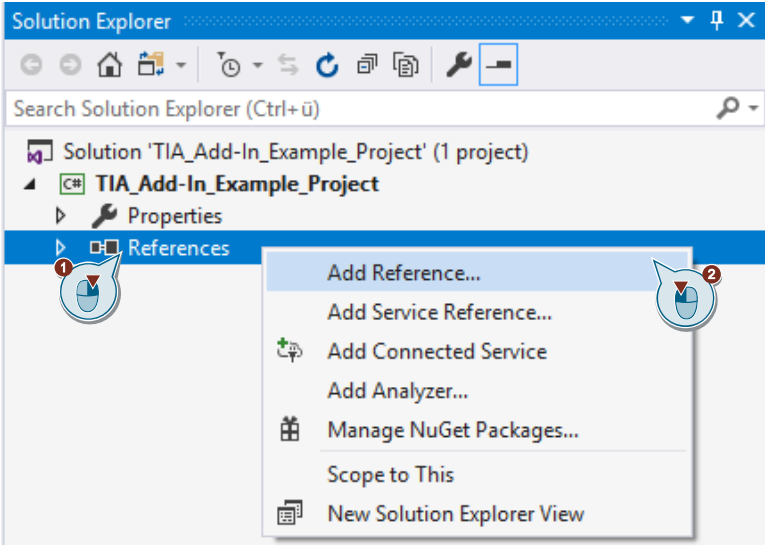
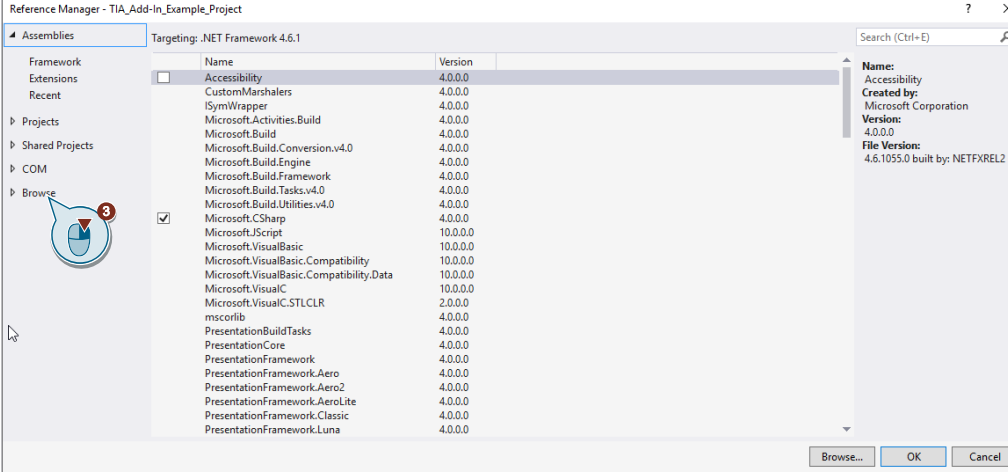
To program a TIA Add-In, a new project must first be created in Visual Studio. If nothing different is specified in the notes, then the clicks must be executed one after the other. More detailed information is provided about the individual clicks in some instances. These notes are numbered corresponding to the click number, and are specified in red. For example: "this is the note for click number two (2)". If there are no notes associated with the click, then the click is self-explanatory, and can be understood based on the texts displayed in Visual Studio.

Table 3-1 Instructions to create a new Visual Studio project

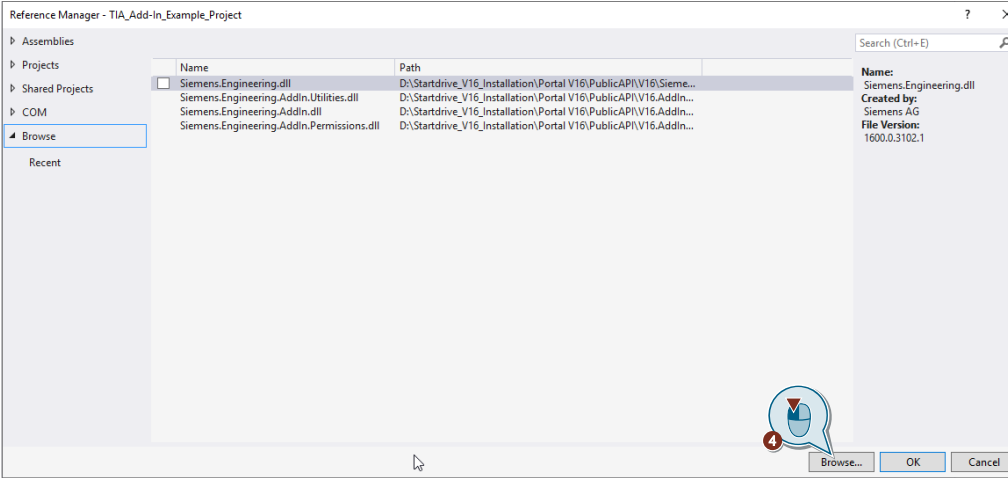
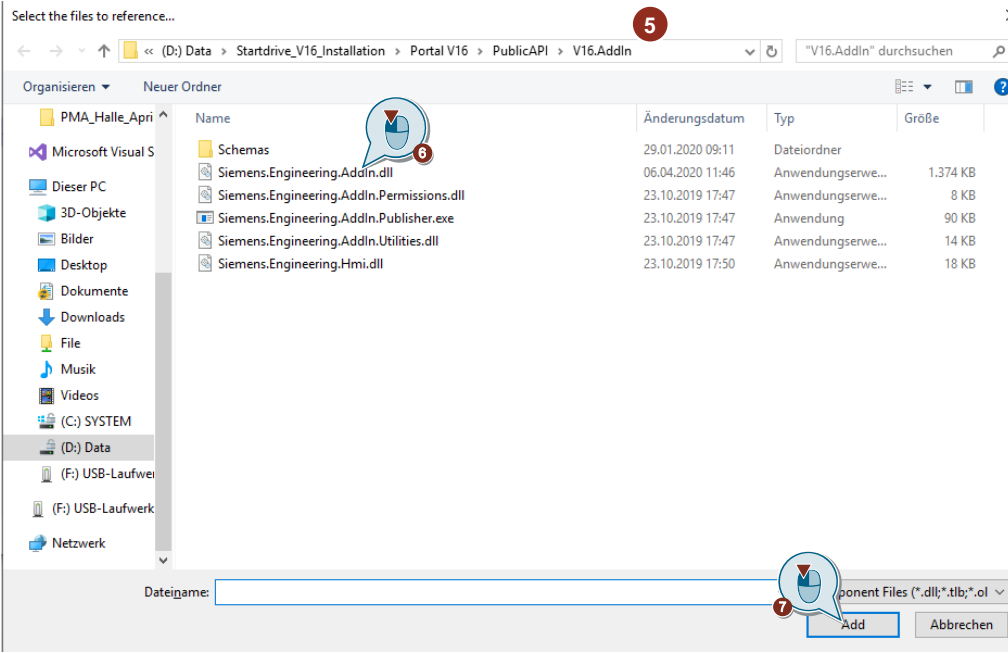
No.	Action
1.	<p>Open the Microsoft Visual Studio, and select File → New → Project.</p> 

3 Creating the Visual Studio project

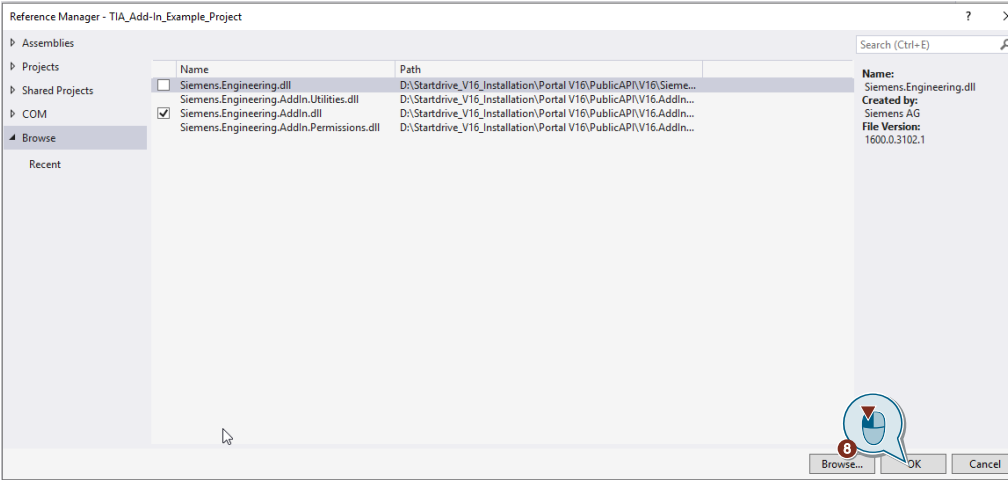
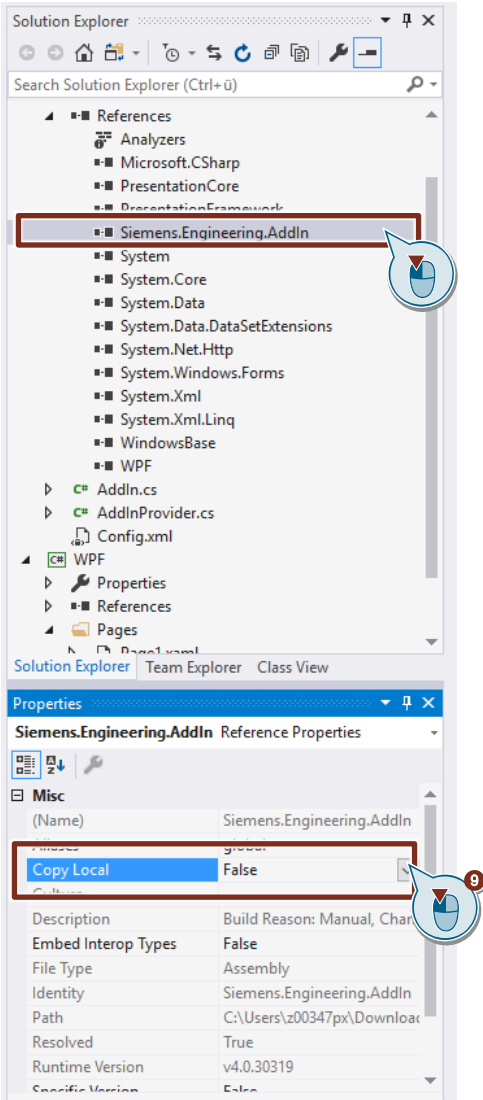
No.	Action
2.	<p>Here, "Class Library" must be selected, so that an Add-In file for the TIA Portal can be created from the Visual Studio project. (2)</p> <p>TIA Portal Openness V16 uses .NET Framework 4.6.x, in this case version 4.6.1 or 4.6.2 must be selected here. TIA Portal Openness V17 doesn't run until .NET Framework version 4.8. The selection of the .NET Framework version can also be located at a different position in this window (5)</p> 
3.	<p>To start, standard class "Class1.cs" is deleted here. (2)</p> 

No.	Action
4.	<p>The Siemens.Engineering.AddIn.dll required must now be integrated into the project. This file, with extension .dll ("Dynamic Link Library"), provides the interface to the TIA Portal. The Dynamic Link Library acts as official interface to the TIA Portal. This interface is also called TIA Portal Openness. The Siemens.Engineering.AddIn.dll also includes the Class TIA Portal, to be able to open a TIA Portal project, for example.</p>  <p>The following steps must be executed to register the Siemens.Engineering.AddIn.dll with the Visual Studio project.</p> <p>For Openness V16 TIA Portal V16 or Startdrive V16 must be installed! (Openness V17: TIA Portal V17 or Startdrive V17)</p>  

3 Creating the Visual Studio project

No.	Action
	 <p>Reference Manager - TIA_Add-In_Example_Project</p> <p>Under certain circumstances, the path to the Siemens.Engineering.AddIn.dll is different than specified here. A search must be made for the path to the TIA Portal, and then in folders "Portal V16" or "Portal V17" → "PublicAPI" → "V16.AddIn" or "V17.AddIn" (5) file Siemens.Engineering.AddIn.dll must be selected (6) and then "Add" clicked (7).</p>  <p>Select the files to reference...</p> <p>Path: D:\Startdrive_V16_Installation > Portal V16 > PublicAPI > V16.AddIn</p> <p>File: Siemens.Engineering.AddIn.dll</p> <p>Buttons: Add, Abbrechen</p>

3 Creating the Visual Studio project

No.	Action
	
	<p>The „Copy Local“ property in the Siemens.Engineering.AddIn.dll has to be set to „false“:</p> 

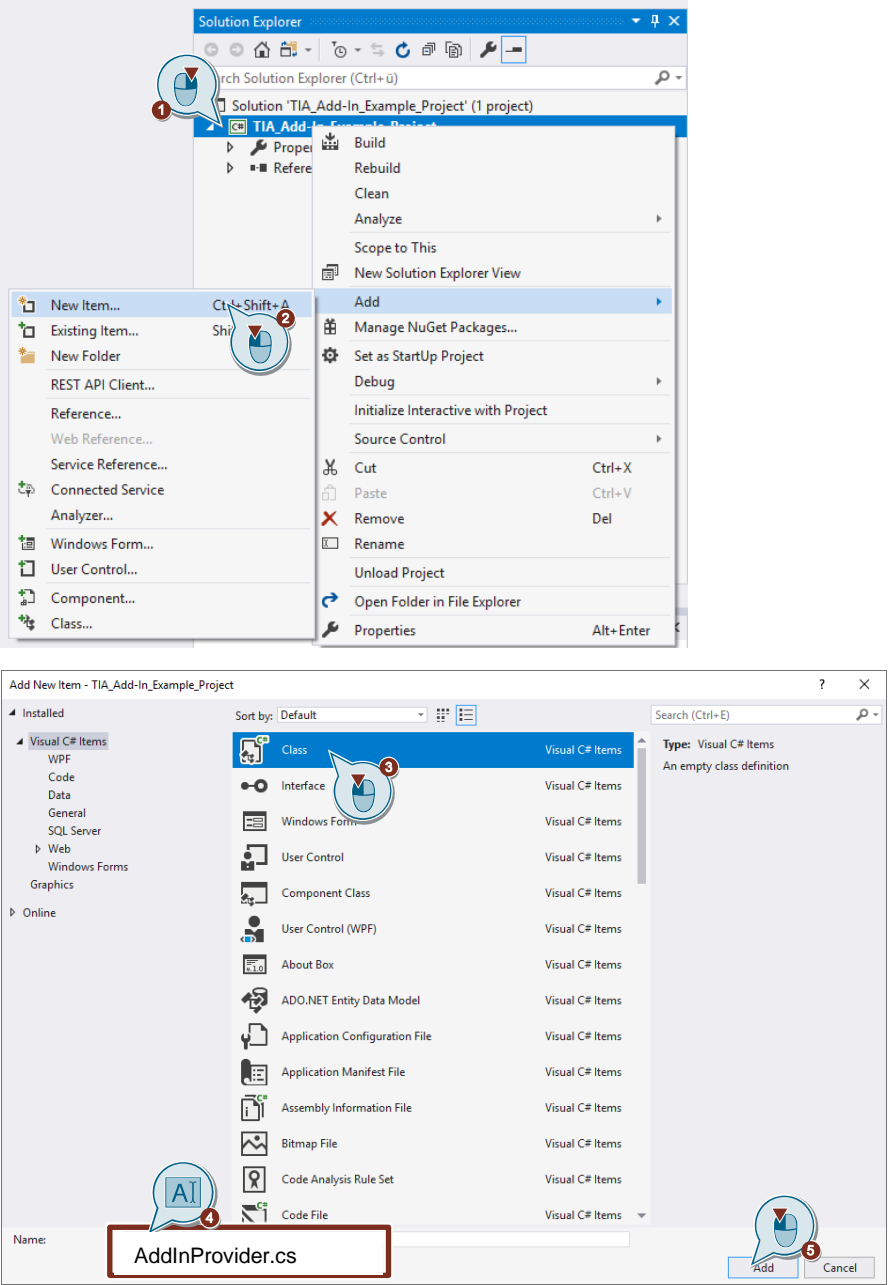
3.2 Creating class "AddInProvider"

Class "AddInProvider" is used to define in which area of the TIA Portal the Add-In is displayed. In principle, a dedicated class must be created for each provider type. The following AddInProviders can be created:

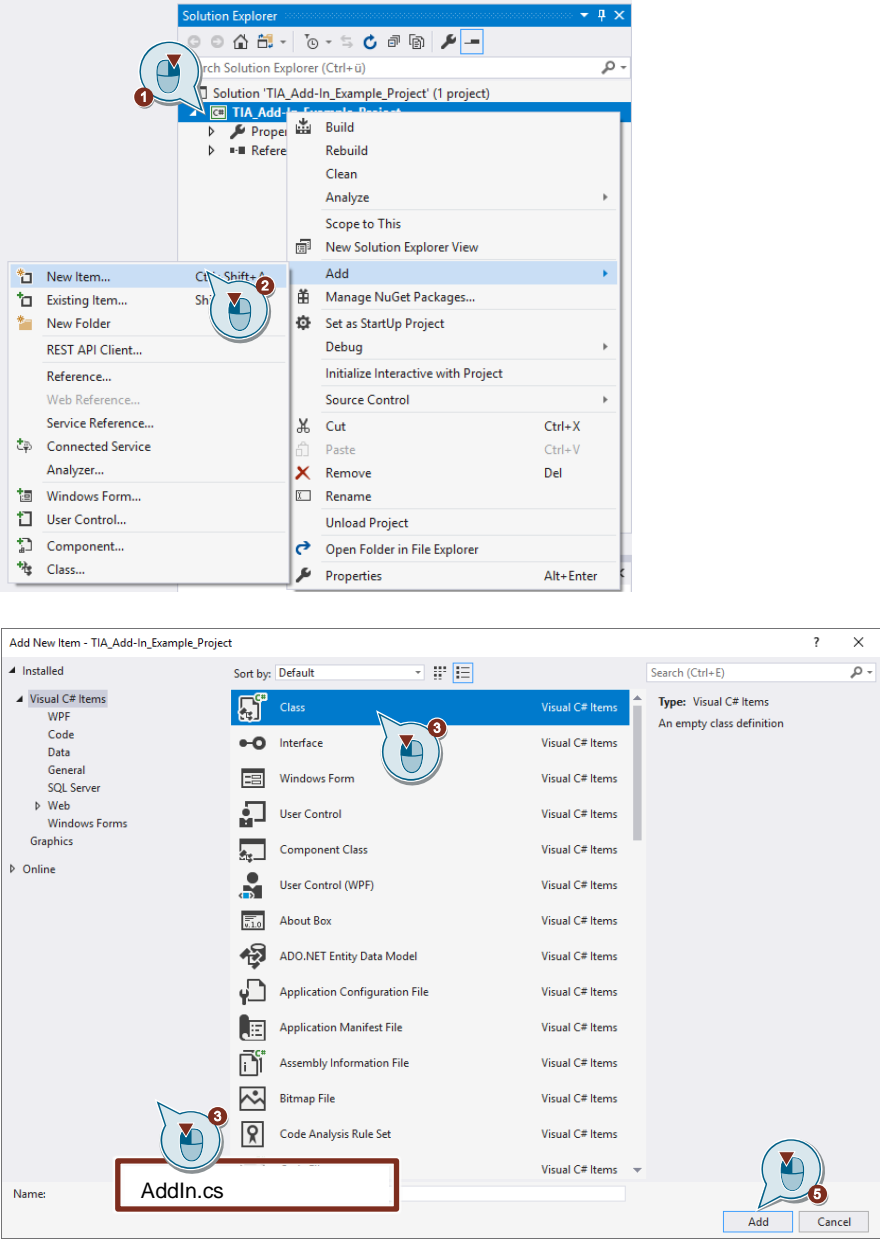
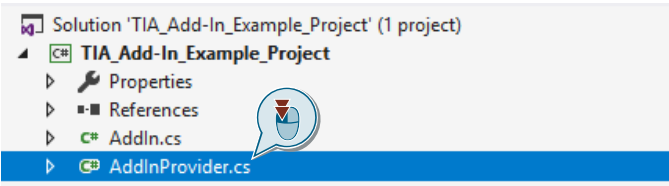
- "ProjectTreeAddInProvider":
Displays the Add-In in the project tree.
- "GlobalLibraryTreeAddInProvider":
Displays the Add-In in the global libraries.
- "ProjectLibraryTreeAddInProvider":
Displays the Add-In in the project library.
- "DevicesAndNetworksAddInProvider"
Displays the Add-In in devices and the network editor. However, please note that the Add-In cannot be displayed in the following areas of the device and network editor:
 - "I/O communication" in the network view
 - "Connections" table in the network view
- "VciAddInProvider":
Displays the Add-In in a work area of the version control interface.

3 Creating the Visual Studio project

Table 3-2 Instructions to create the AddInProvider class

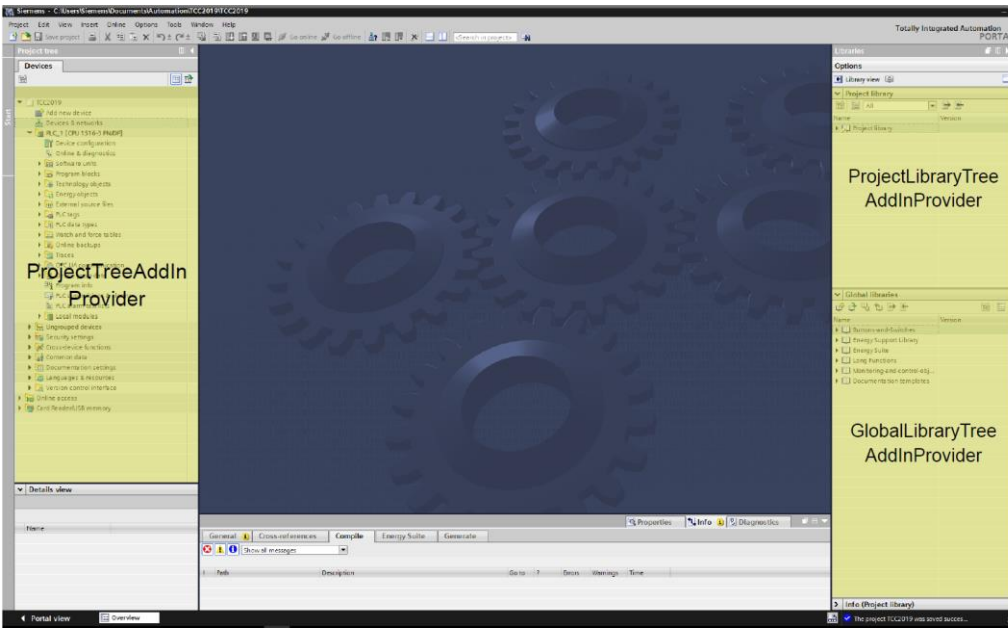
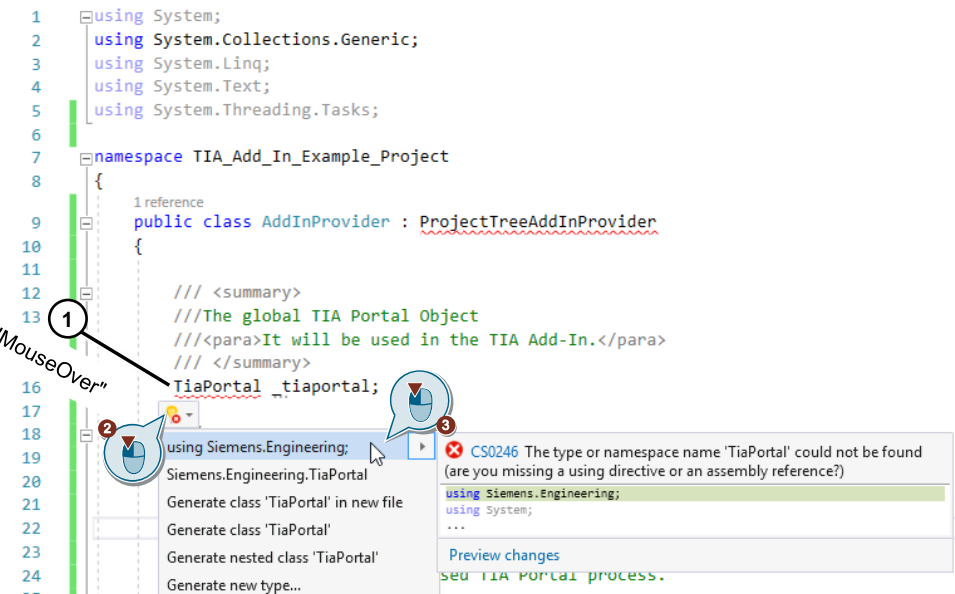
No.	Action
1.	<p>A new class is then added to the project. This class is called "AddInProvider". (4) The AddInProvider is responsible for linking the Add-In to the TIA Portal process.</p>  <p>The screenshot shows the Visual Studio interface. In the top window, the 'Solution Explorer' shows a project named 'TIA_Add-In_Example_Project'. A right-click context menu is open over the project, and the 'Add' option is selected. This opens the 'Add New Item' dialog. In this dialog, the 'Visual C# Items' category is expanded, and the 'Class' item is selected. The 'Name' field at the bottom contains the text 'AddInProvider.cs'. The 'Add' button is highlighted with a red circle and the number 5. A red circle with the number 4 is also present near the 'Name' field.</p>

3 Creating the Visual Studio project

No.	Action
2.	<p>An additional new class is then added to the project. This class is called "AddIn". (4) Class AddIn contains the functionality of the Add-In.</p> 
3.	<p>Class "AddInProvider" is now programmed. To do this, the class must be opened by double-clicking on it.</p> 

No.	Action
	<p>The program code of class "AddInProvider" is adapted as follows (by copying [CTRL + C] & pasting [CTRL + V] the following code in class AddInProvider). More detailed information about the individual variables, methods etc., is provided directly in the code, and displayed as comments in a green font.</p> <p>It is important to create a "public class", as otherwise the TIA Portal will not be able to find these classes at a later point in time. Under certain circumstances, namespace "TIA_Add_In_Example_Project" must be adapted to the solution names used.</p> <pre> using System; using System.Collections.Generic; using System.Linq; using System.Text; using System.Threading.Tasks; namespace TIA_Add_In_Example_Project { public class AddInProvider : ProjectTreeAddInProvider //or ProjectLibraryTreeAddInProvider or GlobalLibraryTreeAddInProvider { /// <summary> ///The global TIA Portal Object ///<para>It will be used in the TIA Add-In.</para> /// </summary> TiaPortal _tiaportal; /// <summary> /// The constructor of the AddInProvider. /// <para>- Creates an object of the class AddInProvider</para> /// <para>- Called when a right-click is performed in TIA</para> /// </summary> /// <param name="tiaportal"> /// Represents the actual used TIA Portal process. /// </param> public AddInProvider(TiaPortal tiaportal) { /* * The acutal TIA Portal process is saved in the * global TIA Portal variable _tiaportal */ _tiaportal = tiaportal; } /// <summary> /// The method is supplemented to include the Add-In /// in the Context Menu of TIA Portal. /// </summary> /// <typeparam name="ContextMenuAddIn"> /// The Add-In will be displayed in /// the Context Menu of TIA Portal. /// </typeparam> /// <returns> /// A new instance of the class AddIn will be created /// which contains the main functionality of the Add-In /// </returns> protected override IEnumerable<ContextMenuAddIn> GetContextMenuAddIns() { yield return new AddIn(_tiaportal); } } } </pre>

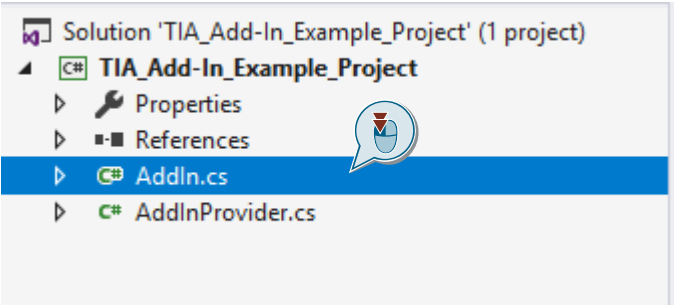
3 Creating the Visual Studio project

No.	Action
	<p>The following diagram shows where Add-Ins can be called in the TIA Portal.</p> <p>"ProjectTreeAddInProvider", "ProjectLibraryTreeAddInProvider" or "GlobalLibraryTreeAddInProvider" can be selected.</p> 
4.	<p>You can now see the program locations marked in red (underlined).</p> <p>This indicates that in the current AddInProvider class, access to Siemens.Engineering.dll still must be activated.</p> <p>To do this, place the mouse on class "TiaPortal" (1) and open the menu with a mouse over. (2)</p>  <pre> 1 using System; 2 using System.Collections.Generic; 3 using System.Linq; 4 using System.Text; 5 using System.Threading.Tasks; 6 7 namespace TIA_Add_In_Example_Project 8 { 9 1 reference 10 public class AddInProvider : <u>ProjectTreeAddInProvider</u> 11 { 12 /// <summary> 13 ///The global TIA Portal Object 14 ///<para>It will be used in the TIA Add-In.</para> 15 /// </summary> 16 <u>TiaPortal</u> _tiaportal; 17 18 2 using Siemens.Engineering; 19 Siemens.Engineering.TiaPortal 20 Generate class 'TiaPortal' in new file 21 Generate class 'TiaPortal' 22 Generate nested class 'TiaPortal' 23 Generate new type... 24 25 } </pre> <p>CS0246 The type or namespace name 'TiaPortal' could not be found (are you missing a using directive or an assembly reference?)</p> <p>using Siemens.Engineering; using System; ...</p> <p>Preview changes sed TIA Portal process.</p>

No.	Action
5.	<p>Repeat step No. 4 for locations "ProjectTreeAddInProvider" and "ContextMenuAddIn" for "GetContextMenuAddIns", so that class AddInProvider looks like this:</p> <pre> using Siemens.Engineering; using Siemens.Engineering.AddIn; using Siemens.Engineering.AddIn.Menu; using System; using System.Collections.Generic; using System.Linq; using System.Text; using System.Threading.Tasks; namespace TIA_Add_In_Example_Project { public class AddInProvider : ProjectTreeAddInProvider //or ProjectLibraryTreeAddInProvider or GlobalLibraryTreeAddInProvider { /// <summary> ///The global TIA Portal Object ///<para>It will be used in the TIA Add-In.</para> /// </summary> TiaPortal _tiaportal; /// <summary> /// The constructor of the AddInProvider. /// <para>- Creates an object of the class AddInProvider</para> /// <para>- Called when a right-click is performed in TIA</para> /// </summary> /// <param name="tiaportal"> /// Represents the actual used TIA Portal process. /// </param> public AddInProvider(TiaPortal tiaportal) { /* * The actual TIA Portal process is saved in the * global TIA Portal variable _tiaportal */ _tiaportal = tiaportal; } /// <summary> /// The method is supplemented to include the Add-In /// in the Context Menu of TIA Portal. /// </summary> /// <typeparam name="ContextMenuAddIn"> /// The Add-In will be displayed in /// the Context Menu of TIA Portal. /// </typeparam> /// <returns> /// A new instance of the class AddIn will be created /// which contains the main functionality of the Add-In /// </returns> protected override IEnumerable<ContextMenuAddIn> GetContextMenuAddIns() { yield return new AddIn(_tiaportal); } } } </pre>

3.3 Programming class "AddIn"

Table 3-3 Instructions to create the AddIn class

No.	Action
1.	<p>Class "AddInProvider" is now programmed. This class includes the structure of the Add-In (button/radio button/checkbox, and at which level the Add-In is called) and the Add-In code to be run.</p> <p>To program the class, the class must be opened by double-clicking on it.</p> 

No.	Action
2.	<p>The program code of class "AddIn" is adapted as follows (by copying [CTRL + C] & pasting [CTRL + V] the following code of steps 2. - 4. in class AddIn). More detailed information about the individual variables, methods etc., is provided directly in the code, and displayed as comments in a green font.</p> <p>It is important to create a "public class", as otherwise the TIA Portal will not be able to find these classes at a later point in time.</p> <pre> using Siemens.Engineering; using Siemens.Engineering.AddIn.Menu; using Siemens.Engineering.HW; using Siemens.Engineering.Settings; using System; using System.Collections.Generic; using System.Linq; using System.Text; using System.Threading.Tasks; using System.Windows.Forms; namespace TIA_Add_In_Example_Project { public class AddIn: ContextMenuAddIn { /// <summary> ///The global TIA Portal Object ///<para>It will be used in the TIA Add-In.</para> /// </summary> TiaPortal _tiaportal; /// <summary> /// The display name of the Add-In. /// </summary> private const string s_DisplayNameOfAddIn = " Add-In Example"; /// <summary> /// The constructor of the AddIn. /// Creates an object of the class AddIn /// Called from AddInProvider, when the first /// right-click is performed in TIA /// Motherclass' constructor of ContextMenuAddin /// will be executed, too. /// </summary> /// <param name="tiaportal"> /// Represents the actual used TIA Portal process. /// </param> public AddIn(TiaPortal tiaportal) : base(s_DisplayNameOfAddIn) { /* * The acutal TIA Portal process is saved in the * global TIA Portal variable _tiaportal * tiaportal comes as input Parameter from the * AddInProvider */ _tiaportal = tiaportal; } } } </pre>

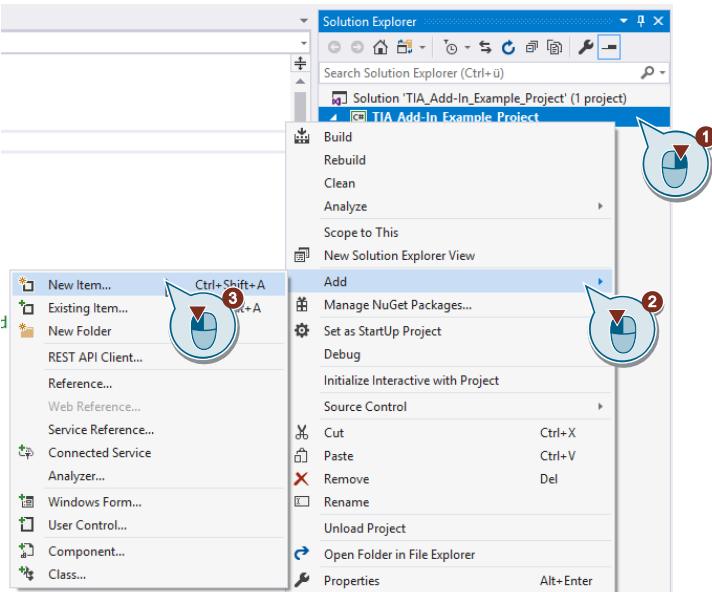
No.	Action
3.	<pre> /// <summary> /// The method is supplemented to include the Add-In /// in the Context Menu of TIA Portal. /// Called when a right-click is performed in TIA /// and a mouse-over is performed on the name of the Add-In. /// </summary> /// <typeparam name="addInRootSubmenu"> /// The Add-In will be displayed in /// the Context Menu of TIA Portal. /// </typeparam> /// <example> /// ActionItems like Buttons/Checkboxes/Radiobuttons /// are possible. In this example, only Buttons will be created /// which will start the Add-In program code. /// </example> protected override void BuildContextMenuItems(ContextMenuAddInRoot addInRootSubmenu) { /* Method addInRootSubmenu.Items.AddActionItem * Will Create a Pushbutton with the text 'Start Add-In Code' * 1st input parameter of AddActionItem is the text of the * button * 2nd input parameter of AddActionItem is the clickDelegate, * which will be executed in case the button 'Start * Add-In Code' will be clicked/pressed. * 3rd input parameter of AddActionItem is the * updateStatusDelegate, which will be executed in * case there is a mouseover the button 'Start * Add-In Code'. * in <placeholder> the type of AddActionItem will be * specified, because AddActionItem is generic * AddActionItem<DeviceItem> will create a button that will be * displayed if a rightclick on a DeviceItem will be * performed in TIA Portal * AddActionItem<Project> will create a button that will be * displayed if a rightclick on the project name * will be performed in TIA Portal */ addInRootSubmenu.Items.AddActionItem<DeviceItem>(("Start Add-In"), OnDoSomething, OnCanSomething); addInRootSubmenu.Items.AddActionItem<Project>("Not Available here", OnClickProject, OnStatusUpdateProject); } /// <summary> /// The method contains the program code of the TIA Add-In. /// Called when the button 'Start Add-In Code' will be pressed. /// </summary> /// <typeparam name="menuSelectionProvider"> /// here, the same type as in addInRootSubmenu.Items.AddActionItem /// must be used -> here it is <DeviceItem> /// </typeparam> private void OnDoSomething(MenuSelectionProvider<DeviceItem> menuSelectionProvider) { //Program of AddIn MessageBox.Show("Hello World"); } </pre>

No.	Action
4.	<pre> /// <summary> /// Called when there is a mousover the button at a DeviceItem. /// It will be used to enable the button. /// </summary> /// <typeparam name="menuSelectionProvider"> /// here, the same type as in addInRootSubmenu.Items.AddActionItem /// must be used -> here it is <DeviceItem> /// </typeparam> private MenuStatus OnCanSomething(MenuSelectionProvider <DeviceItem> menuSelectionProvider) { //enable the button return MenuStatus.Enabled; } /// <summary> /// The method contains the program code of the TIA Add-In. /// Called when the button will be pressed on project level. /// </summary> /// <typeparam name="menuSelectionProvider"> /// here, the same type as in addInRootSubmenu.Items.AddActionItem /// must be used -> here it is <Project> /// </typeparam> private void OnClickProject(MenuSelectionProvider<Project> menuSelectionProvider) { //Do Nothing on Project level } /// <summary> /// Called when there is a mousover the button at the Project /// Level. It will be used to disable the button because no /// action should be performed on project level. /// </summary> /// <typeparam name="menuSelectionProvider"> /// here, the same type as in addInRootSubmenu.Items.AddActionItem /// must be used -> here it is <Project> /// </typeparam> private MenuStatus OnStatusUpdateProject(MenuSelectionProvider <Project> menuSelectionProvider) { //disable the button return MenuStatus.Disabled; } } </pre>

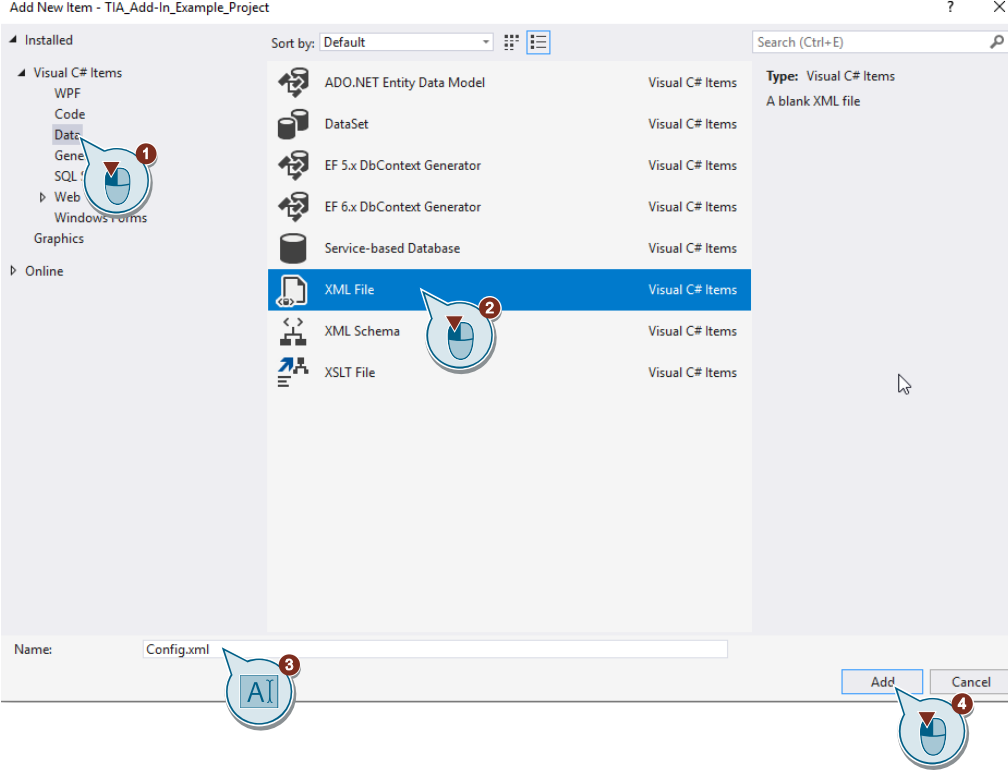
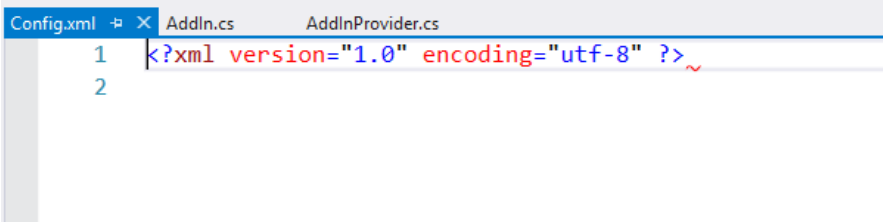
3.4 Creating config.xml

To create the .addin file, a "Config.xml" configuration file must be created in the Visual Studio project. This file can be considered to be a type of construction plan. Information is saved here about the Add-In, for example a brief description, the version and/or the name of the Add-In author. Special rights for the Add-In sequence are also specified in this configuration file. For example, whether the TIA Portal can only be read-accessed or also write-accessed.

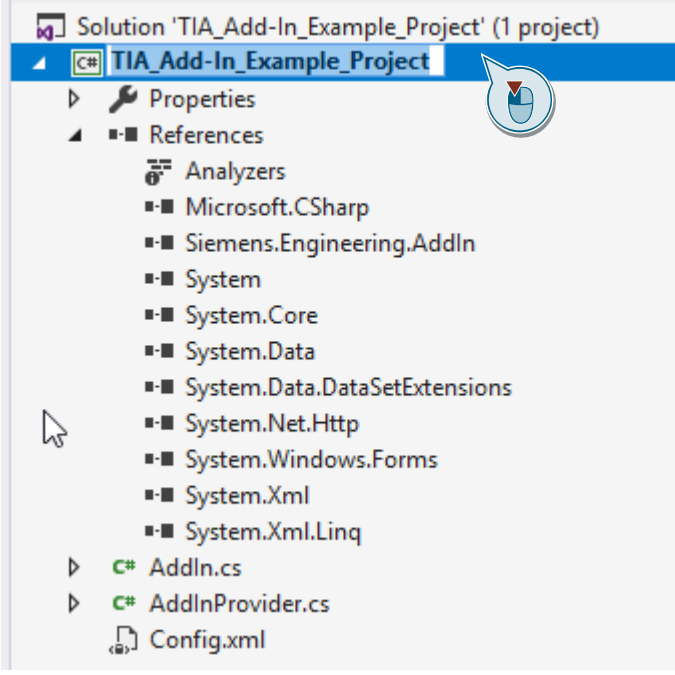
Table 3-4 Instructions to create the configuration file

No.	Action
1.	

3 Creating the Visual Studio project

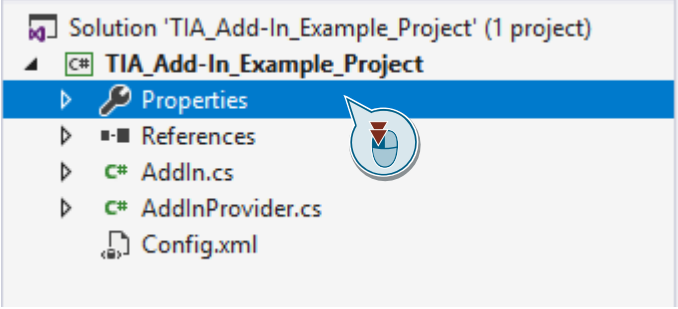
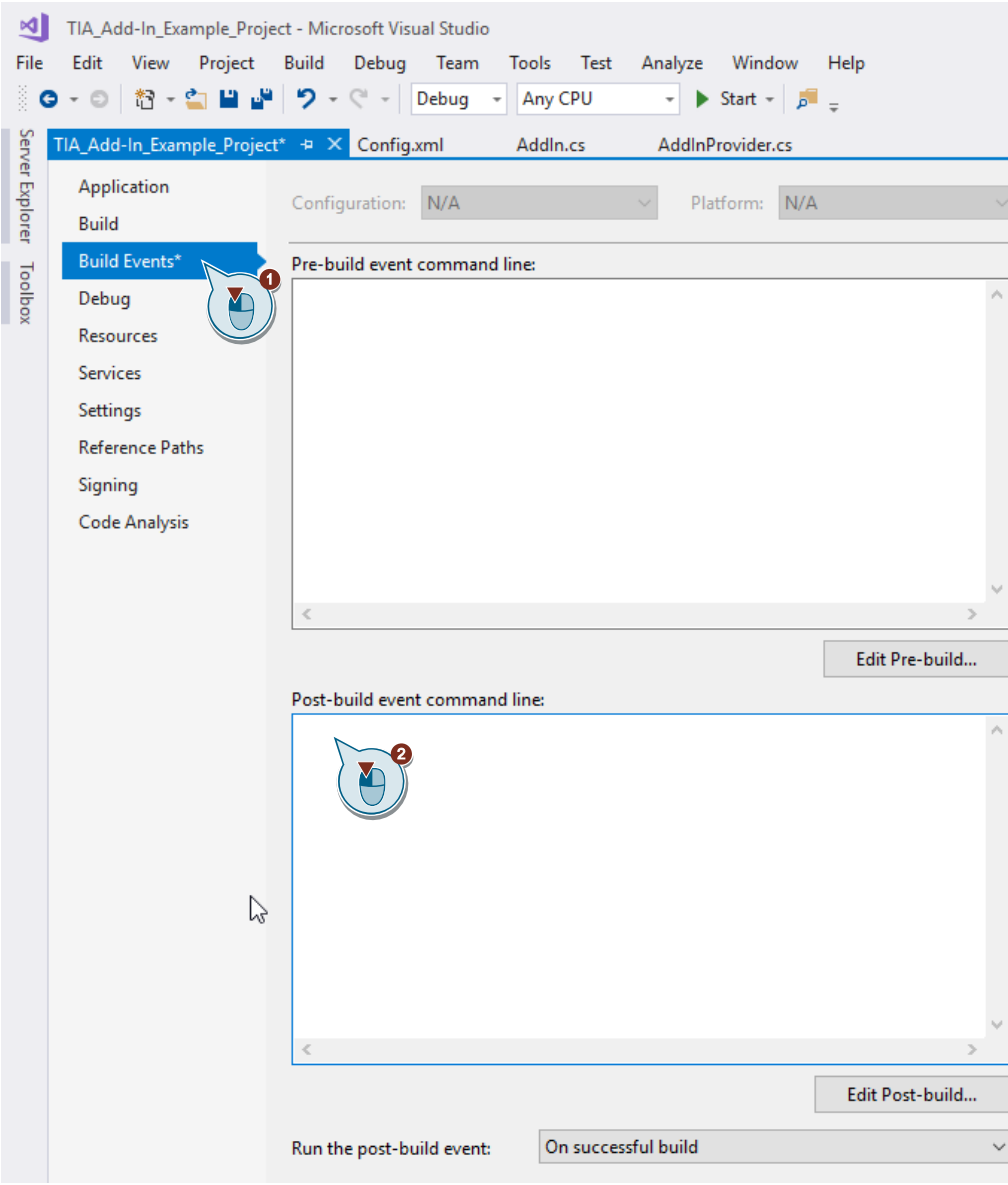
No.	Action
2.	
3.	<p>Delete the following line</p> 

No.	Action
4.	<p>Then insert the following code and populate the fields that are marked:</p> <p>Comments are provided after this table for steps (1), (8) and (9)</p> <pre> <?xml version="1.0" encoding="utf-8" ?> <PackageConfiguration xmlns= "http://www.siemens.com/automation/Openness/AddIn/Publisher/V.."> <Author>Your Name</Author> <Description>The Add-In short description.</Description> <AddInVersion>V0.1</AddInVersion> <Product> <Name>Add-In Name</Name> <Id>123456789</Id> <Version>0.1</Version> </Product> <FeatureAssembly> <AssemblyInfo> <Assembly>bin\Debug\TIA_Add-In_Example_Project.dll</Assembly> <Pdb>bin\Debug\TIA_Add-In_Example_Project.pdb</Pdb> </AssemblyInfo> </FeatureAssembly> <RequiredPermissions> <TIAPermissions> <TIA.ReadWrite /> </TIAPermissions> <SecurityPermissions> <System.Security.Permissions.SecurityPermission.UnmanagedCode/> <System.Security.Permissions.FileDialogPermission/> <System.Security.Permissions.UIPermission/> <System.Security.Permissions.MediaPermission/> <Siemens.Engineering.AddIn.Permissions.ProcessStartPermission/> <System.Net.WebPermission/> <System.Security.Permissions.KeyContainerPermission/> <System.Net.SocketPermission/> <System.Data.Odbc.OdbcPermission/> <System.Security.Permissions.WebBrowserPermission/> <System.Net.NetworkInformation.NetworkInformationPermission/> <System.Security.Permissions.StorePermission/> <System.Configuration.ConfigurationPermission/> <System.Security.Permissions.FileIOPermission/> <System.Diagnostics.EventLogPermission/> <System.Drawing.Printing.PrintingPermission/> <System.Security.Permissions.IsolatedStorageFilePermission/> <System.Data.OleDb.OleDbPermission/> <System.Security.Permissions.RegistryPermission/> <System.Data.SqlClient.SqlClientPermission/> <System.Security.Permissions.EnvironmentPermission/> <System.Net.Mail.SmtpPermission/> </SecurityPermissions> </RequiredPermissions> </PackageConfiguration> </pre> <p>Note Not all security permissions have to be granted, only those that are required for the Add-In! All available security permissions are listed in this table.</p>

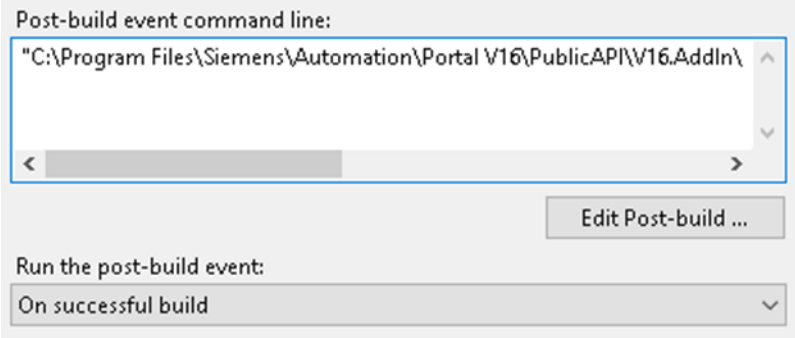
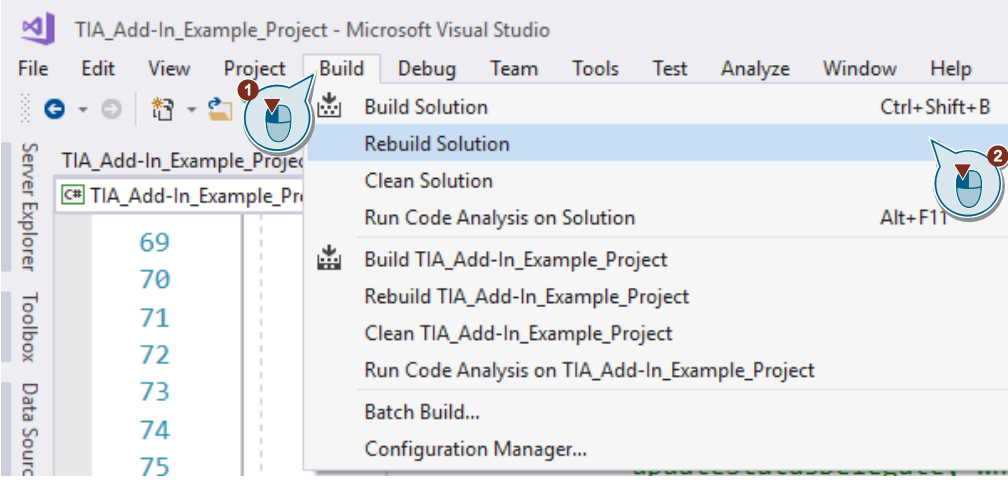
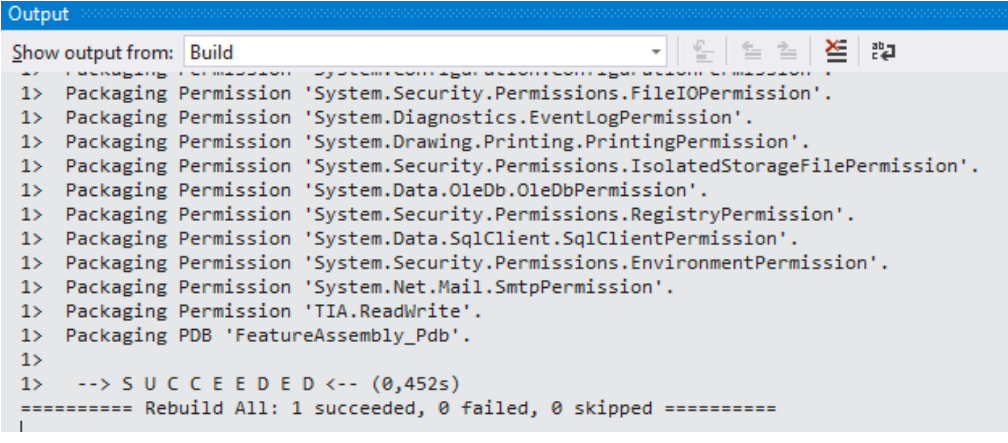
No.	Action
5.	<p>Step (1): Enter here the version of the used Publisher.exe of the Tia portal (either V16 or V17).</p> <p>Step (8): The content of the "Assembly" tag comprises the following: bin\Debug\projectname.dll</p> <p>The projectname is located in the Visual Studio project.:</p>  <p>Step (9): The content of the "Pdb" tag comprises the following: bin\Debug\projectname.pdb</p>

3.5 Creating the .addin file

Table 3-5 Instructions for creating the .addin file

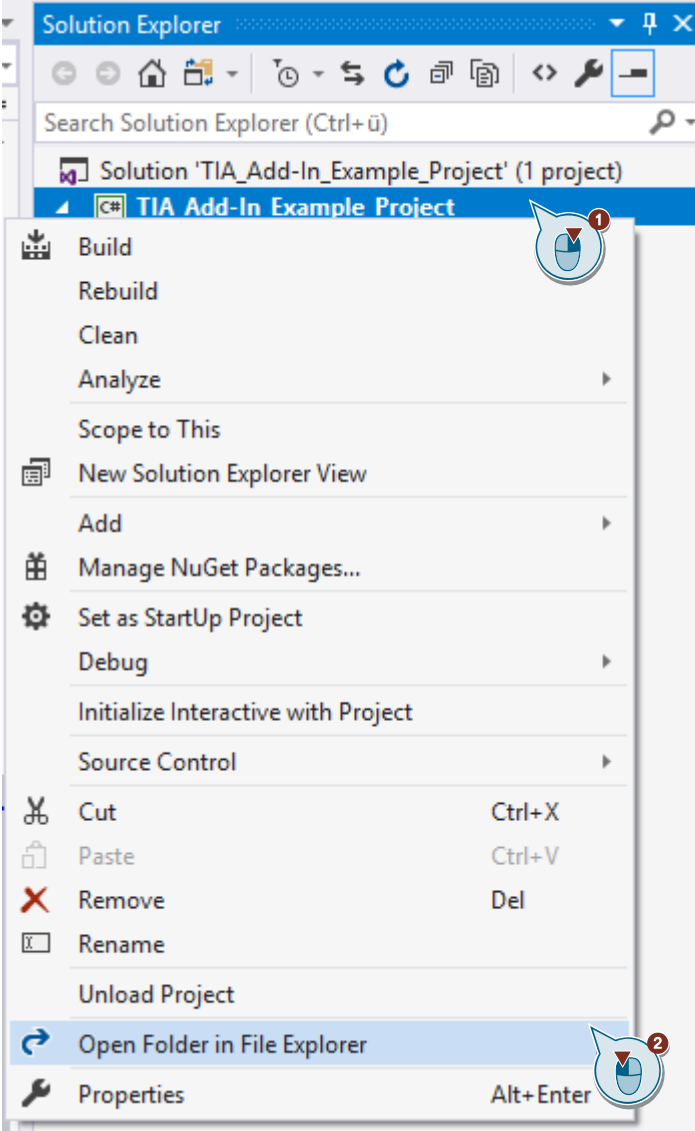
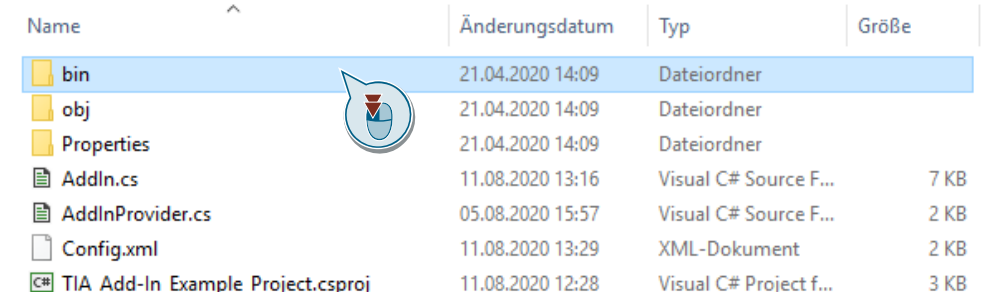
No.	Action
1.	<p>The post-build event of the Visual Studio Solution must now be adapted.</p> 
2.	

3 Creating the Visual Studio project














































No.	Action
3.	<p>Insert the following code as post-build event:</p> <pre>"C:\ProgramFiles\Siemens\Automation\Portal\16\PublicAPI\16.AddIn\Siemens.Engineering.AddIn.Publisher.exe" -f "\$(SolutionDir)\\$(ProjectName)\Config.xml" -v -c</pre>  <p>Note, the first part of the code corresponds to the path where the Siemens.Engineering.AddIn.Publisher.exe is located. This may vary depending on your TIA Portal version and the corresponding installation location.</p>
4.	<p>The Visual Studio Solution is now rebuild:</p> 
5.	<p>The output of the Visual Studio Solution should now look like this.</p> 

























3.6 Installing the .addin file in the TIA Portal

Table 3-6 Instructions for installing the .addin file

No.	Action																																
1.	<p>The .addin file just created must now be installed in the TIA Portal: Open the project path</p>  <p>The screenshot shows the Visual Studio Solution Explorer with the project 'TIA Add-In Example Project' selected. A right-click context menu is open, displaying various actions. A red circle with the number '1' highlights the 'Open Folder in File Explorer' option, which is highlighted in blue. Another red circle with the number '2' highlights the 'Properties' option at the bottom of the menu. The menu items include Build, Rebuild, Clean, Analyze, Scope to This, New Solution Explorer View, Add, Manage NuGet Packages..., Set as StartUp Project, Debug, Initialize Interactive with Project, Source Control, Cut (Ctrl+X), Paste (Ctrl+V), Remove (Del), Rename, Unload Project, Open Folder in File Explorer, and Properties (Alt+Enter).</p>																																
2.	 <p>The screenshot shows a file explorer window with the following contents:</p> <table><thead><tr><th>Name</th><th>Änderungsdatum</th><th>Typ</th><th>Größe</th></tr></thead><tbody><tr><td>bin</td><td>21.04.2020 14:09</td><td>Dateiordner</td><td></td></tr><tr><td>obj</td><td>21.04.2020 14:09</td><td>Dateiordner</td><td></td></tr><tr><td>Properties</td><td>21.04.2020 14:09</td><td>Dateiordner</td><td></td></tr><tr><td>AddIn.cs</td><td>11.08.2020 13:16</td><td>Visual C# Source F...</td><td>7 KB</td></tr><tr><td>AddInProvider.cs</td><td>05.08.2020 15:57</td><td>Visual C# Source F...</td><td>2 KB</td></tr><tr><td>Config.xml</td><td>11.08.2020 13:29</td><td>XML-Dokument</td><td>2 KB</td></tr><tr><td>TIA_Add-In_Example_Project.csproj</td><td>11.08.2020 12:28</td><td>Visual C# Project f...</td><td>3 KB</td></tr></tbody></table> <p>A red circle with the number '1' highlights the 'bin' folder in the file explorer.</p>	Name	Änderungsdatum	Typ	Größe	bin	21.04.2020 14:09	Dateiordner		obj	21.04.2020 14:09	Dateiordner		Properties	21.04.2020 14:09	Dateiordner		AddIn.cs	11.08.2020 13:16	Visual C# Source F...	7 KB	AddInProvider.cs	05.08.2020 15:57	Visual C# Source F...	2 KB	Config.xml	11.08.2020 13:29	XML-Dokument	2 KB	TIA_Add-In_Example_Project.csproj	11.08.2020 12:28	Visual C# Project f...	3 KB
Name	Änderungsdatum	Typ	Größe																														
bin	21.04.2020 14:09	Dateiordner																															
obj	21.04.2020 14:09	Dateiordner																															
Properties	21.04.2020 14:09	Dateiordner																															
AddIn.cs	11.08.2020 13:16	Visual C# Source F...	7 KB																														
AddInProvider.cs	05.08.2020 15:57	Visual C# Source F...	2 KB																														
Config.xml	11.08.2020 13:29	XML-Dokument	2 KB																														
TIA_Add-In_Example_Project.csproj	11.08.2020 12:28	Visual C# Project f...	3 KB																														

3 Creating the Visual Studio project

No.	Action																																								
3.	<table><thead><tr><th>Name</th><th>Änderungsdatum</th><th>Typ</th><th>Größe</th></tr></thead><tbody><tr><td> Debug</td><td>11.08.2020 13:37</td><td>Dateiordner</td><td></td></tr><tr><td> Release</td><td>21.04.2020 14:09</td><td>Dateiordner</td><td></td></tr></tbody></table>	Name	Änderungsdatum	Typ	Größe	 Debug	11.08.2020 13:37	Dateiordner		 Release	21.04.2020 14:09	Dateiordner																													
Name	Änderungsdatum	Typ	Größe																																						
 Debug	11.08.2020 13:37	Dateiordner																																							
 Release	21.04.2020 14:09	Dateiordner																																							
4.	<p>Copy the "TIA_Add-In_Example_Project.addin" file:</p> <p>Projekte > TIA_Add-In_Example_Project > TIA_Add-In_Example_Project > bin > Debug</p> <table><thead><tr><th>Name</th><th>Änderungsdatum</th><th>Typ</th><th>Größe</th></tr></thead><tbody><tr><td> Siemens.Engineering.AddIn.dll</td><td>06.04.2020 11:46</td><td>Anwendungserwe...</td><td>1.374 KB</td></tr><tr><td> Siemens.Engineering.AddIn.xml</td><td>06.04.2020 11:46</td><td>XML-Dokument</td><td>3.190 KB</td></tr><tr><td> Siemens.Engineering.Hmi.dll</td><td>23.10.2019 17:50</td><td>Anwendungserwe...</td><td>18 KB</td></tr><tr><td> Siemens.Engineering.Hmi.xml</td><td>23.10.2019 17:50</td><td>XML-Dokument</td><td>8 KB</td></tr><tr><td> TIA_Add-In_Example_Project.addin</td><td>11.08.2020 13:37</td><td>ADDIN-Datei</td><td>23 KB</td></tr><tr><td> TIA_Add-In_Example_Project.dll</td><td>11.08.2020 13:37</td><td>Anwendungserwe...</td><td>8 KB</td></tr><tr><td> TIA_Add-In_Example_Project.pdb</td><td>11.08.2020 13:37</td><td>Program Debug D...</td><td>20 KB</td></tr></tbody></table>	Name	Änderungsdatum	Typ	Größe	 Siemens.Engineering.AddIn.dll	06.04.2020 11:46	Anwendungserwe...	1.374 KB	 Siemens.Engineering.AddIn.xml	06.04.2020 11:46	XML-Dokument	3.190 KB	 Siemens.Engineering.Hmi.dll	23.10.2019 17:50	Anwendungserwe...	18 KB	 Siemens.Engineering.Hmi.xml	23.10.2019 17:50	XML-Dokument	8 KB	 TIA_Add-In_Example_Project.addin	11.08.2020 13:37	ADDIN-Datei	23 KB	 TIA_Add-In_Example_Project.dll	11.08.2020 13:37	Anwendungserwe...	8 KB	 TIA_Add-In_Example_Project.pdb	11.08.2020 13:37	Program Debug D...	20 KB								
Name	Änderungsdatum	Typ	Größe																																						
 Siemens.Engineering.AddIn.dll	06.04.2020 11:46	Anwendungserwe...	1.374 KB																																						
 Siemens.Engineering.AddIn.xml	06.04.2020 11:46	XML-Dokument	3.190 KB																																						
 Siemens.Engineering.Hmi.dll	23.10.2019 17:50	Anwendungserwe...	18 KB																																						
 Siemens.Engineering.Hmi.xml	23.10.2019 17:50	XML-Dokument	8 KB																																						
 TIA_Add-In_Example_Project.addin	11.08.2020 13:37	ADDIN-Datei	23 KB																																						
 TIA_Add-In_Example_Project.dll	11.08.2020 13:37	Anwendungserwe...	8 KB																																						
 TIA_Add-In_Example_Project.pdb	11.08.2020 13:37	Program Debug D...	20 KB																																						
5.	<p>Now open the installation folder of the TIA Portal. For example:</p> <p>D:\Startdrive_V16_Installation\Portal V16\ Or D:\Startdrive_V17_Installation\Portal V17\</p> <p>Now create a new folder and call it "AddIns" - assuming that this folder does not already exist.</p> <p>It is important that the folder is precisely assigned this name, as otherwise no Add-Ins will be displayed in the TIA Portal.</p> <p>Dieser PC > (D:) Data > Startdrive_V16_Installation > Portal V16 ></p> <table><thead><tr><th>Name</th><th>Änderungsdatum</th><th>Typ</th><th>Größe</th></tr></thead><tbody><tr><td> 109760816_TIAOpennessExplorer_CODE_V10</td><td>09.01.2020 10:43</td><td>Dateiordner</td><td></td></tr><tr><td> AddIns</td><td>11.08.2020 12:28</td><td>Dateiordner</td><td></td></tr><tr><td> Bin</td><td>24.08.2020 13:55</td><td>Dateiordner</td><td></td></tr><tr><td> Catalog</td><td>24.08.2020 13:51</td><td>Dateiordner</td><td></td></tr><tr><td> Data</td><td>29.01.2020 09:34</td><td>Dateiordner</td><td></td></tr><tr><td> EDD_WorkingDir</td><td>08.01.2020 08:28</td><td>Dateiordner</td><td></td></tr><tr><td> EditParametersInSeveralDrivesAddInAcyc_V16001720</td><td>17.02.2020 14:28</td><td>Dateiordner</td><td></td></tr><tr><td> Help</td><td>08.01.2020 08:28</td><td>Dateiordner</td><td></td></tr><tr><td> lib</td><td>29.01.2020 09:40</td><td>Dateiordner</td><td></td></tr></tbody></table>	Name	Änderungsdatum	Typ	Größe	 109760816_TIAOpennessExplorer_CODE_V10	09.01.2020 10:43	Dateiordner		 AddIns	11.08.2020 12:28	Dateiordner		 Bin	24.08.2020 13:55	Dateiordner		 Catalog	24.08.2020 13:51	Dateiordner		 Data	29.01.2020 09:34	Dateiordner		 EDD_WorkingDir	08.01.2020 08:28	Dateiordner		EditParametersInSeveralDrivesAddInAcyc_V16001720	17.02.2020 14:28	Dateiordner		Help	08.01.2020 08:28	Dateiordner		lib	29.01.2020 09:40	Dateiordner	
Name	Änderungsdatum	Typ	Größe																																						
 109760816_TIAOpennessExplorer_CODE_V10	09.01.2020 10:43	Dateiordner																																							
 AddIns	11.08.2020 12:28	Dateiordner																																							
 Bin	24.08.2020 13:55	Dateiordner																																							
 Catalog	24.08.2020 13:51	Dateiordner																																							
 Data	29.01.2020 09:34	Dateiordner																																							
 EDD_WorkingDir	08.01.2020 08:28	Dateiordner																																							
EditParametersInSeveralDrivesAddInAcyc_V16001720	17.02.2020 14:28	Dateiordner																																							
Help	08.01.2020 08:28	Dateiordner																																							
lib	29.01.2020 09:40	Dateiordner																																							

No.	Action																																				
6.	<p>Insert the "TIA_Add-In_Example_Project.addin" file that was previously copied.</p> <p>Dieser PC > (D:) Data > Startdrive_V16_Installation > Portal V16 > AddIns</p> <table><tr><th>Name</th><th>Änderungsdatum</th><th>Typ</th><th>Größe</th></tr><tr><td> CrosscutterFlyingSawApplicationAddIn.a...</td><td>16.07.2020 11:46</td><td>ADDIN-Datei</td><td>1.618 KB</td></tr><tr><td> EditParametersInSeveralDrivesAddIn.addin</td><td>09.01.2020 11:21</td><td>ADDIN-Datei</td><td>755 KB</td></tr><tr><td> Parametertest.addin</td><td>19.02.2020 09:11</td><td>ADDIN-Datei</td><td>30 KB</td></tr><tr><td> Readme.txt</td><td>23.10.2019 16:47</td><td>Textdokument</td><td>2 KB</td></tr><tr><td> SynchronismApplicationAddIn.addin</td><td>15.04.2020 14:48</td><td>ADDIN-Datei</td><td>1.562 KB</td></tr><tr><td> TIA_Add-In_Example_Project.addin</td><td>12.08.2020 09:26</td><td>ADDIN-Datei</td><td>24 KB</td></tr><tr><td> VIBXAddIn.addin</td><td>03.09.2020 17:11</td><td>ADDIN-Datei</td><td>1.542 KB</td></tr><tr><td> WinderApplicationAddIn.addin</td><td>08.04.2020 15:15</td><td>ADDIN-Datei</td><td>238 KB</td></tr></table>	Name	Änderungsdatum	Typ	Größe	 CrosscutterFlyingSawApplicationAddIn.a...	16.07.2020 11:46	ADDIN-Datei	1.618 KB	 EditParametersInSeveralDrivesAddIn.addin	09.01.2020 11:21	ADDIN-Datei	755 KB	 Parametertest.addin	19.02.2020 09:11	ADDIN-Datei	30 KB	 Readme.txt	23.10.2019 16:47	Textdokument	2 KB	 SynchronismApplicationAddIn.addin	15.04.2020 14:48	ADDIN-Datei	1.562 KB	 TIA_Add-In_Example_Project.addin	12.08.2020 09:26	ADDIN-Datei	24 KB	 VIBXAddIn.addin	03.09.2020 17:11	ADDIN-Datei	1.542 KB	 WinderApplicationAddIn.addin	08.04.2020 15:15	ADDIN-Datei	238 KB
Name	Änderungsdatum	Typ	Größe																																		
 CrosscutterFlyingSawApplicationAddIn.a...	16.07.2020 11:46	ADDIN-Datei	1.618 KB																																		
 EditParametersInSeveralDrivesAddIn.addin	09.01.2020 11:21	ADDIN-Datei	755 KB																																		
 Parametertest.addin	19.02.2020 09:11	ADDIN-Datei	30 KB																																		
 Readme.txt	23.10.2019 16:47	Textdokument	2 KB																																		
 SynchronismApplicationAddIn.addin	15.04.2020 14:48	ADDIN-Datei	1.562 KB																																		
 TIA_Add-In_Example_Project.addin	12.08.2020 09:26	ADDIN-Datei	24 KB																																		
 VIBXAddIn.addin	03.09.2020 17:11	ADDIN-Datei	1.542 KB																																		
 WinderApplicationAddIn.addin	08.04.2020 15:15	ADDIN-Datei	238 KB																																		

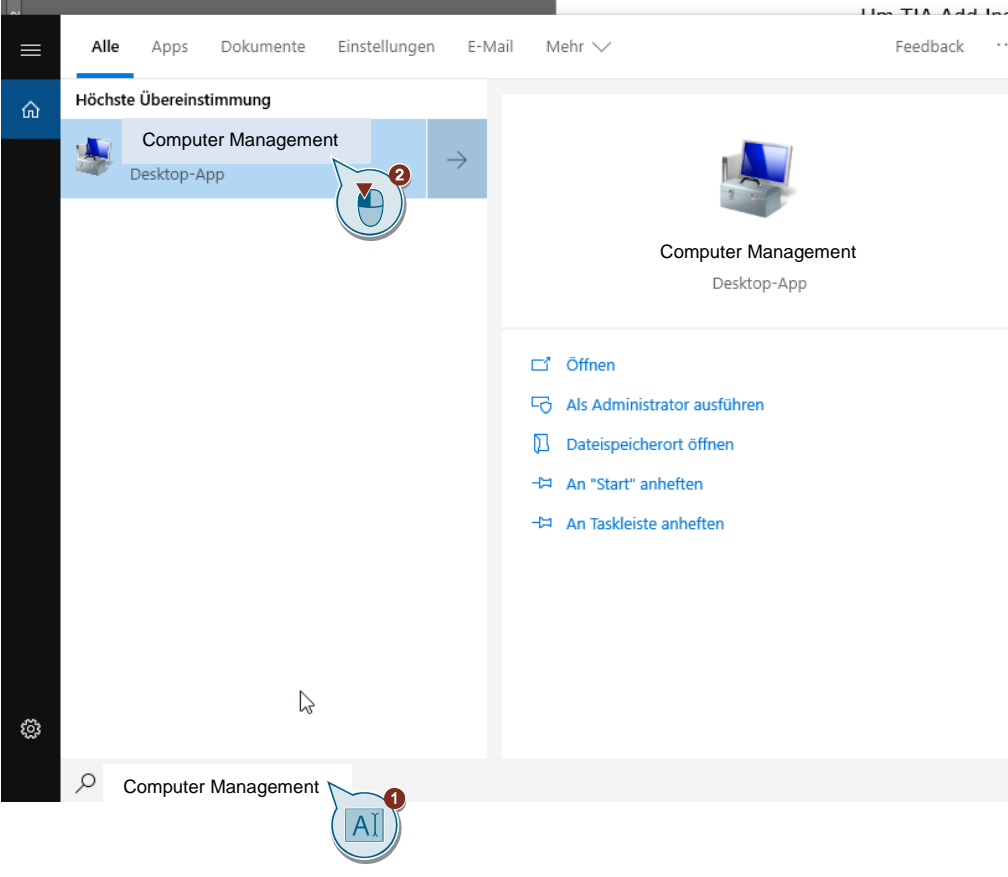
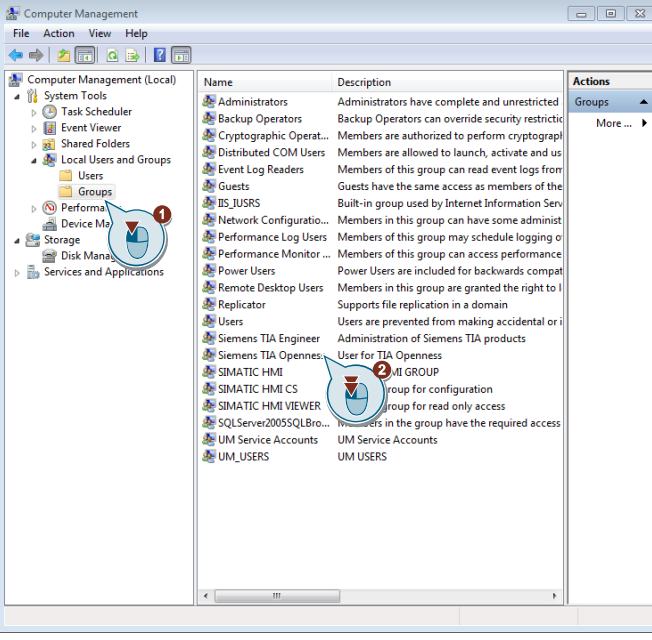
3.7 Joining the "TIA Portal Openness" user group

The current computer user must be a member of the local "Siemens TIA Portal Openness" user group in order to be able to run TIA Add-Ins or TIA Portal Openness.

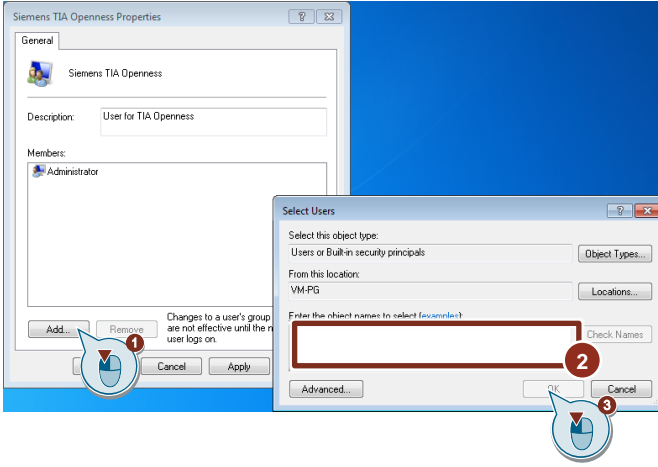
Note

The user group only has to be added once, and is then saved.

Table 3-7 Instructions on how to join the Siemens TIA Portal Openness user group

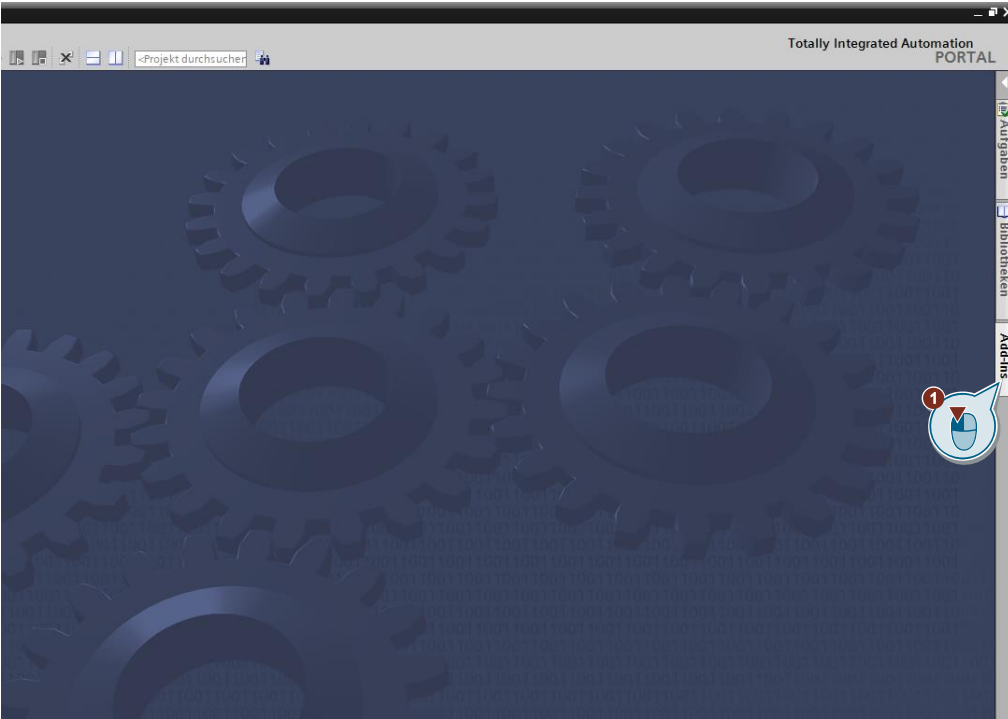
No.	Action
1.	<p>Open the Computer Management of Microsoft Windows.</p> <p>For example, using the search function -> Computer Management (1).</p> 
2.	<p>Left click on folder "Groups" under "System Tools > Local Users and Groups" (1).</p> <p>Then double-click on group "Siemens TIA Portal Openness" (2).</p> 

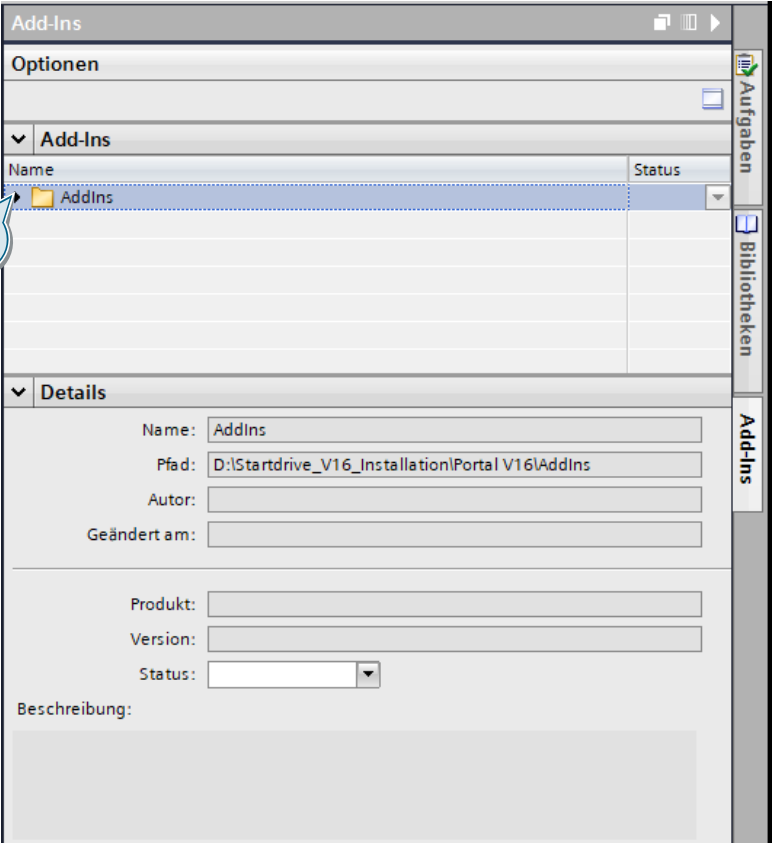
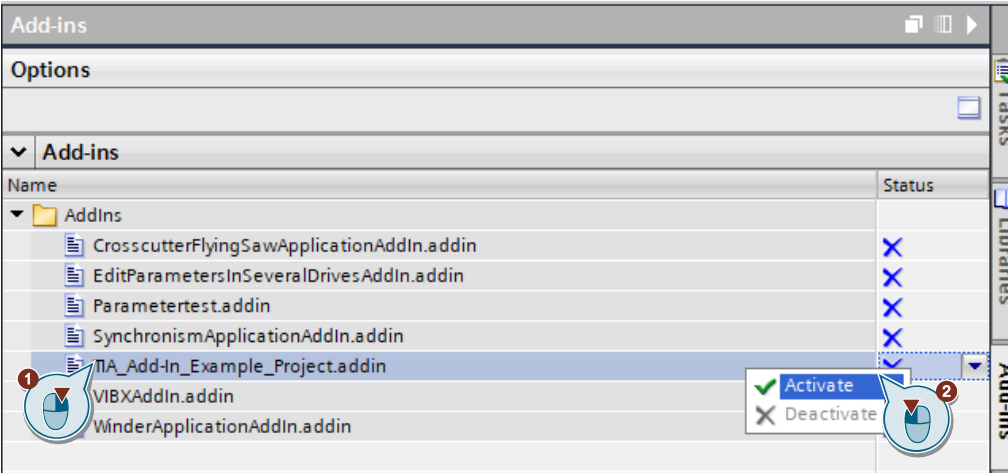
3 Creating the Visual Studio project

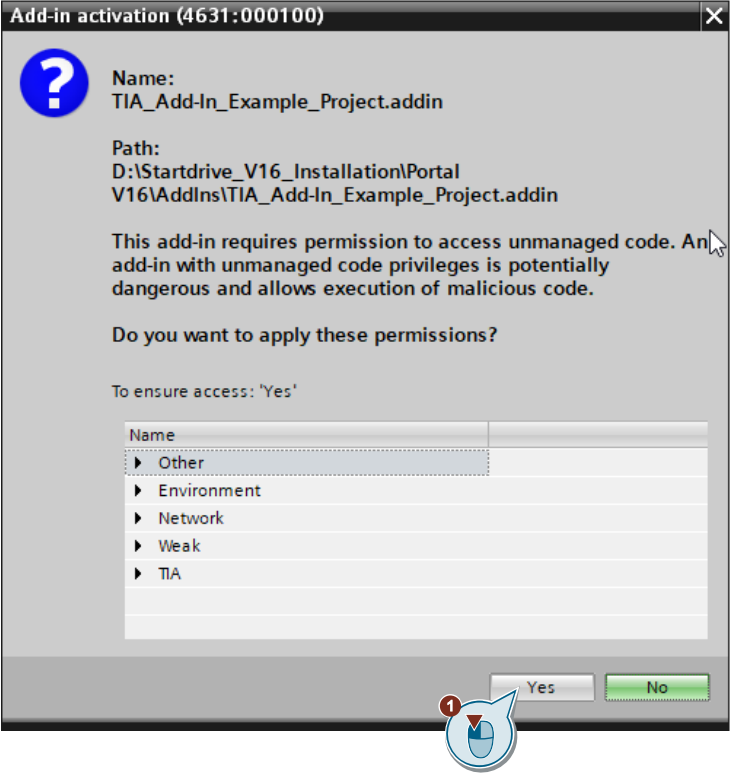
No.	Action
3.	<p>Click on "Add" (1) and enter your name or a user name of the person that is to use the TIA Add-In (2).</p> <p>Confirm your entry by clicking on "OK" (3).</p> 
4.	Restart your PC to complete the changes.

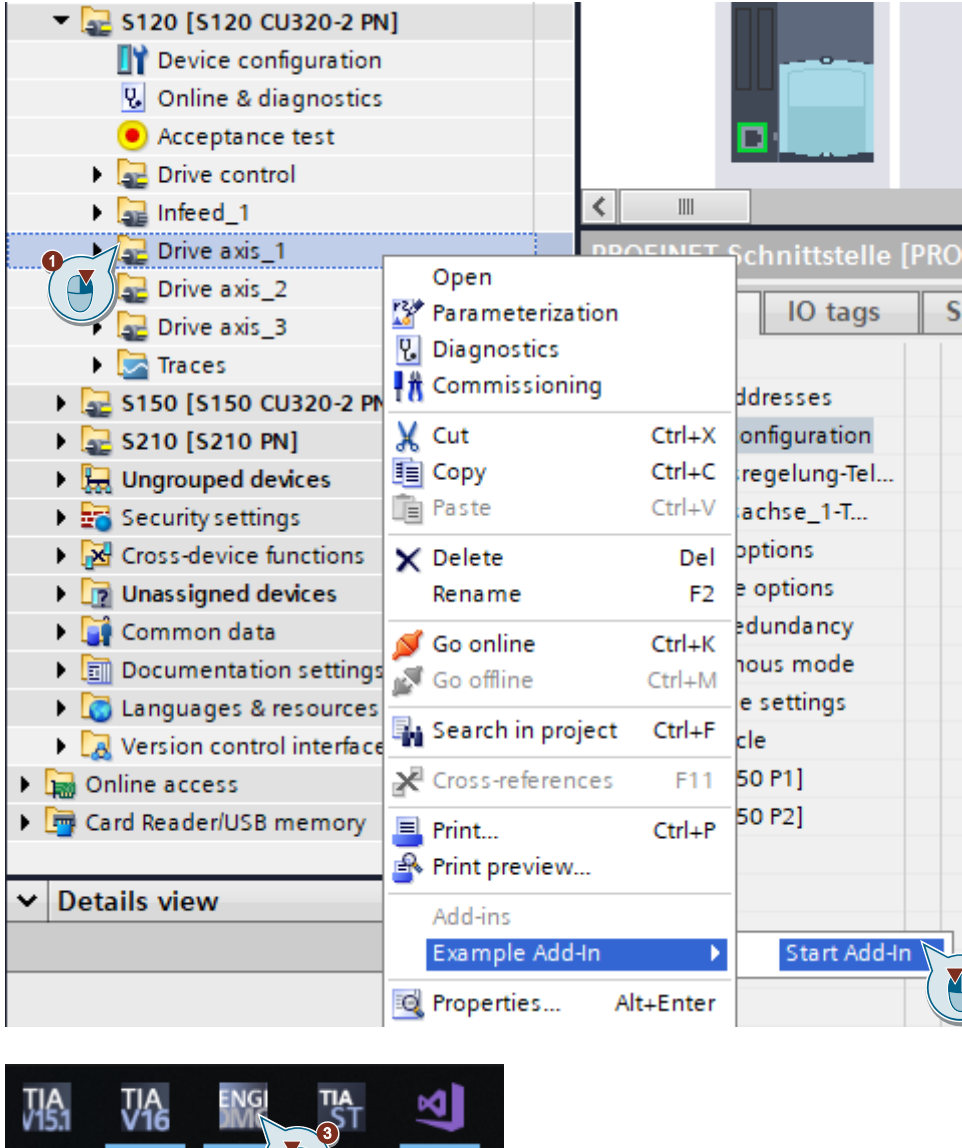
3.8 Activating and running the Add-In in the TIA Portal

Table 3-8 Instructions to activate the Add-In in the TIA Portal

No.	Action
1.	<p>On the right-hand side of the TIA Portal screen there is a tab called "Add-Ins"; press on this tab to open it.</p> 

No.	Action
2.	<p>In the next step, expand folder "AddIns".</p> 
3.	<p>Now right-click on "TIA_Add-In_Example_Project.addin", and select "Activate".</p> 

No.	Action
4.	<p>Click on "Yes" to activate the Add-In.</p> 

No.	Action
5.	<p>Right-click on a SINAMICS drive axis or a SIMATIC controller, for example (1), then, under entry "Add-Ins" → select "Start Add-In" (2) to display the message box (3) on the screen.</p> 

3.9 Note on the template of the current status

Note

The Visual Studio project is archived at this point, and saved under the name "01_TemplateAddIn_VisualStudioProject" and made available.

Name	Änderungsdatum	Typ	Größe
01_TemplateAddIn_VisualStudioProject.zip	17.11.2020 16:37	ZIP-komprimierte...	871 KB
02_Template_Drive_Parameter_Scripting.zip	17.11.2020 16:34	ZIP-komprimierte...	811 KB
03_Template_Drive_Parameter_Scripting_incl_LogFile.zip	17.11.2020 16:31	ZIP-komprimierte...	1.145 KB

4 Startdrive Openness – interconnecting drive parameters

Instructions for changing drive parameters of SINAMICS drives are described in the second document of the SIOS article.

For this purpose, a Visual Studio project is provided, which can be used as a template to directly access the drive parameters of an axis. All necessary preconditions, such as the structure of a TIA Add-In, are already part of the template. Also the possibility to create telegrams is described in the document.

All described methods are collected in a library, which can be integrated into other Visual Studio projects.

Elements of the second document and library:

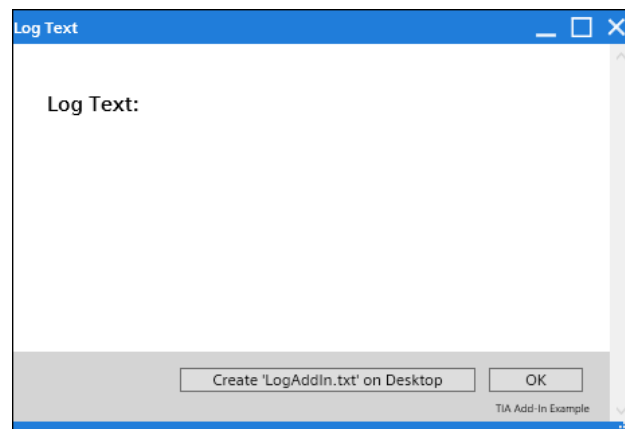
- Changing adjustable parameters
- Creating BICO interconnections
- Creating telegrams

5 Design template for a user interface

In the following chapter, a template is provided to display a log text in a display window. To do this, a Visual Studio project is provided, which can be used as template. The template contains the same code as in Chapter 4, which means that the drive parameters of an axis can be accessed. All the necessary preconditions, e.g. the structure of TIA Add-In are already part of the template. Changing the telegram is also part of the template.

This template has a new feature with the option of displaying a log file on the screen. The log text is also saved in a .txt file on the desktop.



Fig. 5-1 Log text: All parameters interconnected using the TIA Add-In



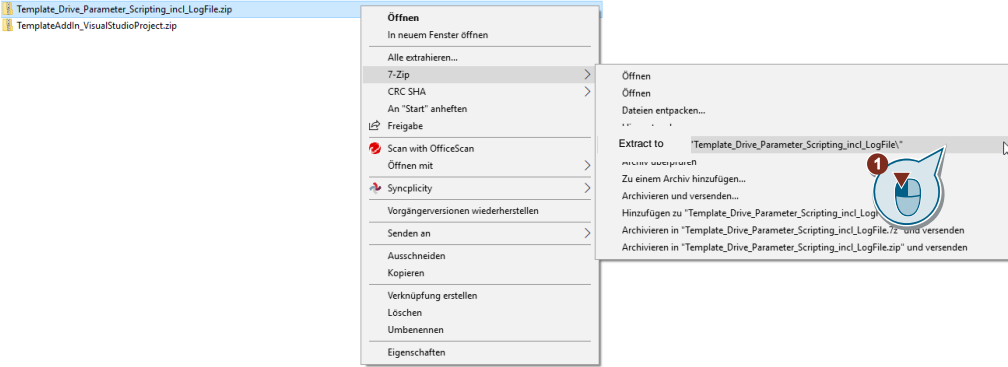
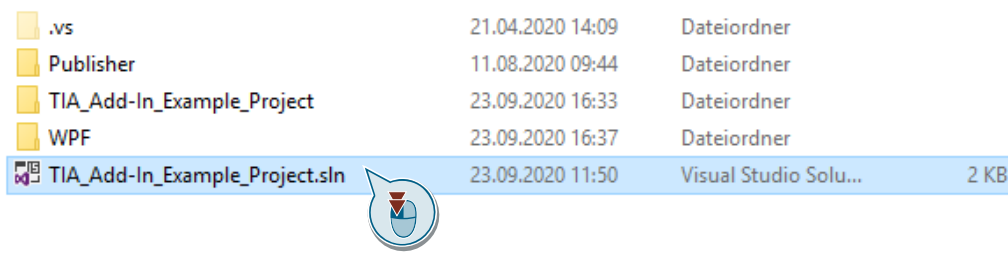
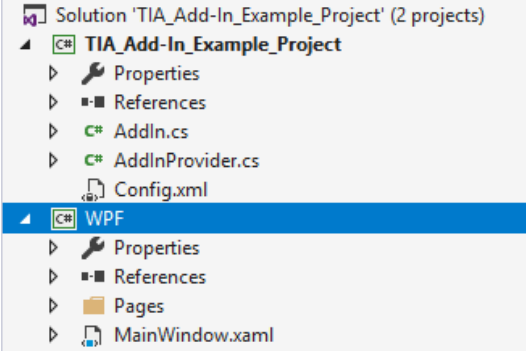
Further, the log text can be optionally saved in a .txt file called "LogAddIn.txt", where the user only has to select button "Create 'LogAddIn.txt' on Desktop".

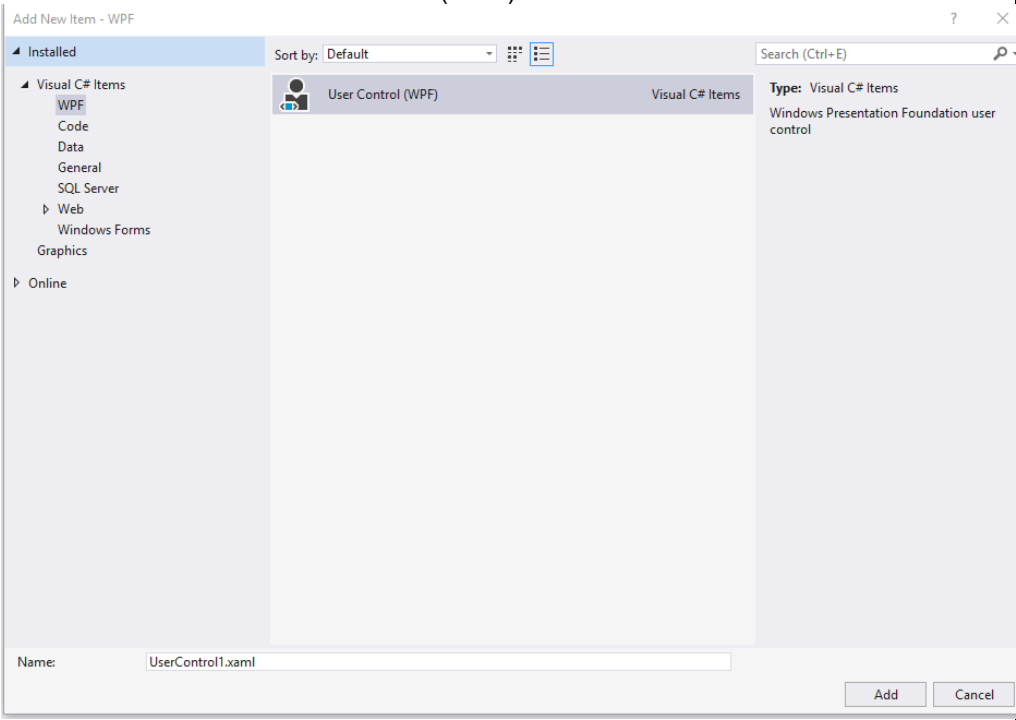
5.1 Template of a window

The starting point for this chapter is Visual Studio project "Template_Drive_Parameter_Scripting_incl_LogFile". To do this, the Visual Studio project provided must be unzipped.

No.	Action
1.	 Template_Drive_Parameter_Scripting_incl_LogFile.zip 23.09.2020 16:38 ZIP-komprimierte... 1.003 KB 

5 Design template for a user interface

No.	Action
2.	
3.	
4.	<p>In the following steps, a brief description is provided as to how the project from Chapter 4 was expanded.</p> <p>To display a new window in the TIA Portal, the Visual Studio Solution was expanded to include an additional project called "WPF".</p> 

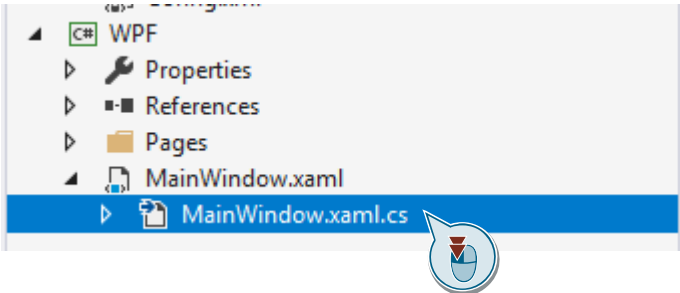
No.	Action
5.	<div><p>A Windows Presentation Foundation (WPF) user control was then inserted:</p></div>

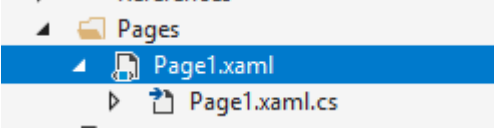
No.	Action
6.	<p>The WPF was called "MainWindow". It displays the window in the TIA Portal style. The XAML code of the window looks like this:</p> <pre> <Window xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" x:Class="WPF.MainWindow" Title="Template TIA Add-In" Topmost="True" WindowStyle="None" AllowsTransparency="True" Background="White" Width="{Binding SystemParameters.PrimaryScreenWidth}" Height="{Binding SystemParameters.PrimaryScreenHeight}" WindowStartupLocation="CenterScreen" BorderThickness="1,0" BorderBrush="Black" ResizeMode="CanResizeWithGrip" ScrollViewer.VerticalScrollBarVisibility="Visible" > <Grid Width="auto"> <Grid.RowDefinitions> <RowDefinition Height="25"></RowDefinition> <RowDefinition Height="502*"></RowDefinition> <RowDefinition Height="10"></RowDefinition> </Grid.RowDefinitions> <Grid.ColumnDefinitions> <ColumnDefinition Width="25*"></ColumnDefinition> <ColumnDefinition Width="26"></ColumnDefinition> <ColumnDefinition Width="26"></ColumnDefinition> <ColumnDefinition Width="26"></ColumnDefinition> </Grid.ColumnDefinitions> <DockPanel x:Name="TitleBar" DockPanel.Dock="Top" Height="25" VerticalAlignment="Top" Grid.ColumnSpan="4"> <DockPanel.Background> <LinearGradientBrush EndPoint="0.5,1" MappingMode="RelativeToBoundingBox" StartPoint="0.5,0"> <GradientStop Color="Black"/> <GradientStop Color="#FF207DDA"/> <GradientStop Color="#FF207DDA" Offset="0.222"/> </LinearGradientBrush> </DockPanel.Background> <TextBlock x:Name="title" HorizontalAlignment="Left" VerticalAlignment="Center" FontFamily="Siemens Sans" Text="Log Text" Foreground="White" FontWeight="Bold" DataContext="{Binding Mode=OneWay}" Width="auto" Height="auto" /> </DockPanel> <Button Name="btnMinimizeScreen" Grid.Column="1" Grid.Row="0" HorizontalAlignment="Right" VerticalAlignment="Top" BorderThickness="0" Style="{StaticResource {x:Static ToolBar.ButtonStyleKey}}" Height="30" Width="25" Click="btnMinimizeScreen_Click" Opacity="0.995" Background="{x:Null}" BorderBrush="{x:Null}" Foreground="{x:Null}" RenderTransformOrigin="0,0"> <StackPanel Height="25" Width="20"> <Rectangle Fill="White" Stroke="White" Height="2" Width="15" StrokeThickness="0.5" VerticalAlignment="Bottom" Grid.Column="1" Margin="2,15,2,0"/> </StackPanel> </Button> </pre>

No.	Action
7.	<pre> <Button Name="btnMaximizeScreen" Grid.Column="2" Grid.Row ="0" HorizontalAlignment="Right" VerticalAlignment="Top" BorderThickness="0" Style="{StaticResource {x:Static ToolBar.ButtonStyleKey}}" Height="30" Width="25" Click="btnMaximizeScreen_Click" Opacity="0.995" Background="{x:Null}" BorderBrush="{x:Null}" Foreground="{x:Null}" RenderTransformOrigin="0,0"> <StackPanel> <Grid> <Grid.ColumnDefinitions> <ColumnDefinition Width="10"></ColumnDefinition> <ColumnDefinition Width="10"></ColumnDefinition> </Grid.ColumnDefinitions> <Rectangle x:Name="MinimizeRectangle1" Fill="White" Stroke="Black" Width="13" Height="13" StrokeThickness="0.5" Grid.Column="1" Margin="-4,-6,-3,2" HorizontalAlignment="Left" Visibility="Hidden" /> <Rectangle x:Name="MinimizeRectangle2" Fill="White" Stroke="Black" Grid.Column="0" Width="13" Height="13" StrokeThickness="0.5" Margin="2,-2,0,0" Grid.ColumnSpan="2" HorizontalAlignment="Left" Visibility="Hidden" /> <Rectangle x:Name="MaximizeRectangle1" Fill="Transparent" Stroke="White" Width="15" Height="15" StrokeThickness="1.5" Grid.Column="1" Margin="-7,-5,0,0" HorizontalAlignment="Left" /> <Rectangle x:Name="MaximizeRectangle2" Fill="White" Stroke="White" Width="15" Height="1" StrokeThickness="1.0" Grid.Column="1" Margin="-7,-18,0,0" HorizontalAlignment="Left" /> </Grid> </StackPanel> </Button> <Button Name="btnCloseScreen" Grid.Column="3" Grid.Row ="0" HorizontalAlignment="Right" VerticalAlignment="Top" BorderThickness="0" Style="{StaticResource {x:Static ToolBar.ButtonStyleKey}}" Height="30" Width="25" Click="btnCloseScreen_Click" Opacity="0.995" Background="{x:Null}" BorderBrush="{x:Null}" Foreground="{x:Null}" RenderTransformOrigin="0,0"> <StackPanel> <Grid> <Grid.ColumnDefinitions> <ColumnDefinition Width="10"></ColumnDefinition> <ColumnDefinition Width="10"></ColumnDefinition> </Grid.ColumnDefinitions> <Rectangle Fill="White" Stroke="White" Height="2" StrokeThickness="0.5" VerticalAlignment="Bottom" RenderTransformOrigin="0.396,-2.25" HorizontalAlignment="Right" Width="16" Grid.ColumnSpan="2" Margin="0,-2,-3,2"> <Rectangle.RenderTransform> <RotateTransform CenterX="0" CenterY="0" Angle="45" /> </Rectangle.RenderTransform> </Rectangle> </pre>

No.	Action
8.	<pre> <Rectangle Fill="White" Stroke="White" Height="2" StrokeThickness="0.5" VerticalAlignment="Bottom" Grid.Column="0" RenderTransformOrigin="0.396,-2.25" HorizontalAlignment="Left" Width="16" Grid.ColumnSpan="2" Margin="-1,0,0,0"> <Rectangle.RenderTransform> <TransformGroup> <ScaleTransform/> <SkewTransform AngleY="-0.04"/> <RotateTransform Angle="-45"/> <TranslateTransform X="-0.004" Y="-0.004"/> </TransformGroup> </Rectangle.RenderTransform> </Rectangle> </Grid> </StackPanel> </Button> <ScrollViewer Name="myScrollviewer" Grid.Row="1" Grid.ColumnSpan="4"> <Frame x:Name="frame" Content="" ScrollViewer.CanContentScroll="True" ScrollViewer.VerticalScrollBarVisibility="Auto" HorizontalAlignment="Stretch" VerticalAlignment="Stretch" NavigationUIVisibility="Hidden" Grid.Row="1" Grid.ColumnSpan="4" /> </ScrollViewer> <DockPanel x:Name="LowerBar" Grid.ColumnSpan="4" Grid.Row="2" VerticalAlignment="Bottom" HorizontalAlignment="Stretch" DockPanel.Dock="Bottom" Height="10"> <DockPanel.Background> <LinearGradientBrush EndPoint="0.5,1" MappingMode="RelativeToBoundingBox" StartPoint="0.5,0"> <GradientStop Color="#FF207DDA"/> <GradientStop Color="#FF207DDA"/> <GradientStop Color="#FF207DDA" Offset="0.613"/> <GradientStop Color="#FF207DDA" Offset="0.136"/> </LinearGradientBrush> </DockPanel.Background> </DockPanel> </Grid> </Window> </pre> <div data-bbox="411 1429 464 1451">Note</div> <div data-bbox="531 1429 1169 1491">For the usage of the template, the SecurityPermission „Unmanaged Code“ is required, see chapter 3.4.</div>

No.	Action
9.	<p>To be able to use the template or the new project, the security permission "Unmanaged Code" is required in the Config.xml (2) and also the tag "AdditionalAssemblies" is needed to include the second project named "WPF" in the .addin file (1), see also chapter 3.4. The Config.xml is then supplemented as follows:</p> <pre> <?xml version="1.0" encoding="utf-8" ?> <PackageConfiguration xmlns= "http://www.siemens.com/automation/Openness/AddIn/Publisher/V.."> <Author>Your Name</Author> <Description>The Add-In short description.</Description> <AddInVersion>V0.1</AddInVersion> <Product> <Name>Add-In Name</Name> <Id>123456789</Id> <Version>0.1</Version> </Product> <FeatureAssembly> <AssemblyInfo> <Assembly>bin\Debug\TIA_Add-In_Example_Project.dll</Assembly> <Pdb>bin\Debug\TIA_Add-In_Example_Project.pdb</Pdb> </AssemblyInfo> </FeatureAssembly> <AdditionalAssemblies> <AssemblyInfo> <Assembly>bin\Debug\WPF.dll</Assembly> <Pdb>bin\Debug\WPF.pdb</Pdb> </AssemblyInfo> </AdditionalAssemblies> <RequiredPermissions> <TIAPermissions> <TIA.ReadWrite /> </TIAPermissions> <SecurityPermissions> <System.Security.Permissions.SecurityPermission.UnmanagedCode/> <System.Security.Permissions.FileDialogPermission/> <System.Security.Permissions.UIPermission/> <System.Security.Permissions.MediaPermission/> <Siemens.Engineering.AddIn.Permissions.ProcessStartPermission/> <System.Net.WebPermission/> <System.Security.Permissions.KeyContainerPermission/> <System.Net.SocketPermission/> <System.Data.Odbc.OdbcPermission/> <System.Security.Permissions.WebBrowserPermission/> <System.Net.NetworkInformation.NetworkInformationPermission/> <System.Security.Permissions.StorePermission/> <System.Configuration.ConfigurationPermission/> <System.Security.Permissions.FileIOPermission/> <System.Diagnostics.EventLogPermission/> <System.Drawing.Printing.PrintingPermission/> <System.Security.Permissions.IsolatedStorageFilePermission/> <System.Data.OleDb.OleDbPermission/> <System.Security.Permissions.RegistryPermission/> <System.Data.SqlClient.SqlClientPermission/> <System.Security.Permissions.EnvironmentPermission/> <System.Net.Mail.SmtpPermission/> </SecurityPermissions> </RequiredPermissions> </PackageConfiguration> </pre> <p>The XML code is annotated with two red boxes and numbers:</p> <ul style="list-style-type: none"> Box 1: Encloses the <code><AdditionalAssemblies></code> section, which includes the <code><AssemblyInfo></code> for the WPF project. Box 2: Encloses the <code><System.Security.Permissions.SecurityPermission.UnmanagedCode/></code> tag within the <code><SecurityPermissions></code> section.

No.	Action
10.	<p>The associated MainWindow.xaml.cs code looks like this:</p> 
11.	<pre> using System; using System.Windows; using System.Collections.Generic; namespace WPF { /// <summary> /// Interaction logic for MainWindow.xaml /// </summary> public partial class MainWindow : Window { public MainWindow(String logText) { InitializeComponent(); //load a new page to the Mainwindow frame.Content = new Page1(this, logText); } /// <summary> /// Minimize the Window /// </summary> private void btnMinimizeScreen_Click(object sender, RoutedEventArgs e) { this.frame.Focus(); this.WindowState = WindowState.Minimized; } /// <summary> /// Maximize the Window /// </summary> private void btnMaximizeScreen_Click(object sender, RoutedEventArgs e) { if (this.WindowState == WindowState.Normal) { this.WindowState = WindowState.Maximized; this.MinimizeRectangle1.Visibility = Visibility.Visible; this.MinimizeRectangle2.Visibility = Visibility.Visible; this.MaximizeRectangle1.Visibility = Visibility.Hidden; this.MaximizeRectangle2.Visibility = Visibility.Hidden; } else { this.WindowState = WindowState.Normal; this.MinimizeRectangle1.Visibility = Visibility.Hidden; this.MinimizeRectangle2.Visibility = Visibility.Hidden; this.MaximizeRectangle1.Visibility = Visibility.Visible; this.MaximizeRectangle2.Visibility = Visibility.Visible; } this.frame.Focus(); } } </pre>

No.	Action
12.	<pre> /// <summary> /// Reset the Scrollbar /// </summary> public void SetMyScrollViewerToTop() { myScrollviewer.ScrollToHorizontalOffset(0); myScrollviewer.ScrollToVerticalOffset(0); } /// <summary> /// Close the Window /// </summary> private void btnCloseScreen_Click(object sender, RoutedEventArgs e) { this.frame.Focus(); closeWindow(); } public void closeWindow() { Close(); } /// <summary> /// Set the title of the Window /// </summary> public void setTitle(String newTitle) { this.Title= newTitle; } } </pre>
13.	<p>This means that there is a template for the TIA Portal window. Pages can then be loaded one after the other in this TIA window. To do this, an additional WPF user control was inserted in the template:</p> 

No.	Action
14.	<p>The XAML code looks like this:</p> <pre> <Page xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:d="http://schemas.microsoft.com/expression/blend/2008" x:Class="WPF.Page1" mc:Ignorable="d" Title="Page1" Width="auto" Height="auto" > <Grid Width="auto" Background="White"> <Grid.RowDefinitions> <RowDefinition Height="10*" /> <RowDefinition Height="75*" /> <RowDefinition Height="30*" /> <RowDefinition Height="22*" /> </Grid.RowDefinitions> <Grid.ColumnDefinitions> <ColumnDefinition Width="65*" /> <ColumnDefinition Width="230" /> <ColumnDefinition Width="80" /> <ColumnDefinition Width="10" /> </Grid.ColumnDefinitions> <Label x:Name="label" Grid.Row="0" HorizontalAlignment="Left" Margin="20,30,0,0" VerticalAlignment="Top" FontSize="16" FontWeight="Bold"> <TextBlock x:Name="textBlock" TextWrapping="Wrap" Text="Log Text:" FontFamily="Siemens Sans" FontWeight="Bold" Width="Auto" MinWidth="500" /> </Label> <Label x:Name="label2" Grid.Row="0" HorizontalAlignment="Left" Margin="20,60,0,0" VerticalAlignment="Top" FontSize="16" FontWeight="Bold"> <TextBlock x:Name="textBlockInterconnections" TextWrapping="Wrap" Text="" FontFamily="Siemens Sans" FontWeight="Normal" Width="Auto" MinWidth="500" /> </Label> <Rectangle x:Name="BottomRectangle" Grid.Row="2" Grid.RowSpan="2" Grid.ColumnSpan="4" Fill="#FFD4D4" /> <Button x:Name="button_ok" Content="OK" Grid.Row="2" Grid.Column="2" Margin="0,0,10,0" FontFamily="Siemens Sans" VerticalAlignment="Bottom" Click="Button_ok_Click" /> <Button x:Name="button_createLog" Content="Create 'LogAddIn.txt' on Desktop" Grid.Row="2" Grid.Column="1" Margin="0,0,10,0" FontFamily="Siemens Sans" VerticalAlignment="Bottom" Click="Button_createLog_Click" Background="#FFDDDD" /> <Label Content="TIA Add-In Example" HorizontalAlignment="Left" Grid.Row="3" Grid.Column="2" VerticalAlignment="Bottom" FontSize="8" /> </Grid> </Page> </pre>
15.	<p>The associated Page1.xaml.cs code looks like this:</p> 

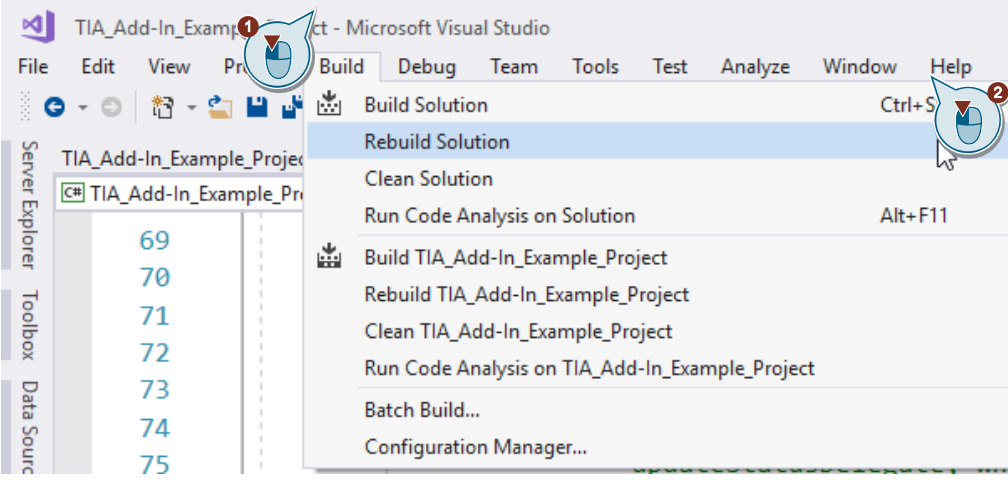
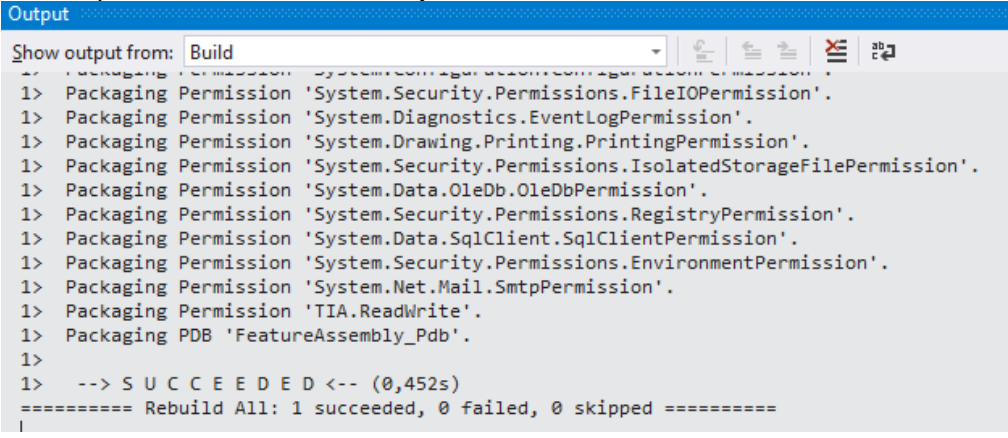
No.	Action
16.	<pre> using System; using System.IO; using System.Windows; using System.Windows.Controls; using System.Windows.Media; namespace WPF { /// <summary> /// Interaction logic for Page1.xaml /// </summary> public partial class Page1 : Page { MainWindow _window; string _text; public Page1(MainWindow window, String logText) { InitializeComponent(); button_createLog.Background = new SolidColorBrush(Color.FromRgb(221,221,221)); _window = window; _window.setTitle("MessageBox"); textBlockInterconnections.Text = logText; _text = logText; } /// <summary> /// Close the Window /// </summary> /// <param name="sender"></param> /// <param name="e"></param> private void Button_ok_Click(object sender, RoutedEventArgs e) { _window.closeWindow(); } /// <summary> /// Create a new log file on the Desktop named LogAddIn.txt /// </summary> private void Button_createLog_Click(object sender, RoutedEventArgs e) { // Set a variable to the Documents path. string docPath = Environment.GetFolderPath(Environment.SpecialFolder.Desktop); // Write the log text to a new file named "LogAddIn.txt". File.WriteAllText(System.IO.Path.Combine(docPath, "LogAddIn.txt"), _text); //Change the buttons background color to green button_createLog.Background = new SolidColorBrush(Color.FromRgb(0, 255, 0)); } } } </pre>
17.	<p>In class "AddIn", the main window is called as follows:</p> <pre> MainWindow myWindow = new MainWindow(_logText); myWindow.ShowDialog(); </pre> <p>Parameter _logText is a string variable that includes the required log text.</p>

5.2 Using the template

No.	Action
1.	<div><div><div><div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><</div></div></div></div>

No.	Action
4.	<p>The template can be used as described in Chapter 4. The chapter includes all the information as to how the template must be used to be able to change SINAMICS drive parameters. However, the difference is the option of be able to display a log text on the screen. To do this, string variable <code>_logText</code> must be supplied with data. Examples are saved in the code as comments. For example, the following must be done if you want to set parameter <code>p2000 = 1234</code>, and include this interconnection in the log text.</p> <pre>#region Interconnections on selected drive axis #region Setting-Parameters selectedDriveParameters.Find(2000, -1).Value = "1234"; _logText = _logText + nameof(selectedAxis) + ":p2000 = 1234" + Environment.NewLine; #endregion #endregion</pre> <p>This window is automatically called at the end of "OnDoSomething":</p> <pre>1 reference private void OnDoSomething(MenuSelectionProvider<DeviceItem> menuSelectionProvider) { DriveObject myDriveObject = null; //Get the actual selected DeviceItem from TIA Portal IEnumerable<DeviceItem> selection = menuSelectionProvider.GetSelection<DeviceItem>(); //change parameters for each selected drive in TIA Portal foreach (DeviceItem actDeviceItem in selection) ... MainWindow myWindow = new MainWindow(_logText); myWindow.ShowDialog(); }</pre> <p>To change parameters or to create telegrams, please follow the information provided in Chapter Fehler! Verweisquelle konnte nicht gefunden werden., from step 4.</p>

5.3 Creating the .addin file for TIA Portal

1.	<p>To be able to use a changed template in the TIA Portal, the Visual Studio Solution must be rebuilt:</p> 
2.	<p>The Output of the Visual Studio Project should look like this.</p> 
3.	<p>Then chapters 3.6, 3.7 and 3.8 must be followed.</p>

6 Additional reference recommendation

6.1 TIA Portal Openness Manual

An overview of all TIA Portal Openness functions is provided in manual "SIMATIC TIA Portal Openness: Automating creation of projects - system manual".

Chapter 5.20 includes Startdrive-specific functions.

<https://support.industry.siemens.com/cs/ww/en/view/109798533>

Chapter 5.20 includes Startdrive-specific functions. These functions are also described here. It would make sense to compare both documents.

<https://support.industry.siemens.com/cs/ww/en/view/109763491>

6.2 Programming support TIA Portal Openness Explorer

"TIA Portal Openness Explorer" helps you obtain an overview of the TIA Portal Openness API.

The "TIA Portal Openness Explorer" tool provides a special view of the TIA Portal Openness object model, and provides significant programming support.

<https://support.industry.siemens.com/cs/ww/en/view/109760816>

6.3 Example of a more comprehensive TIA Add-In

The subsequent example includes a Visual Studio project of a comprehensive TIA Add-In called "EditParametersInSeveralDrives".

The TIA Add-In "EditParametersInSeveralDrives" facilitates the commissioning of similar drives by copying an existing configuration from one drive object to other drive objects. The Visual Studio project is part of what has been published to extend your know-how

<https://support.industry.siemens.com/cs/ww/en/view/109777633>

7 FAQ

7.1 Why can I not access a certain SINAMICS parameter?

Not all drive parameters can be accessed via TIA Portal Openness. A whitelist contains all parameters that can be accessed via TIA Portal Openness. You can obtain the whitelist from your Siemens contact person. The whitelist can be found in the following entry:

<https://support.industry.siemens.com/cs/ww/en/view/109763491>

8 Appendix

8.1 Application support

Siemens AG
Digital Industries
Factory Automation
Production Machines
DI FA PMA APC
Frauenauracher Str. 80
D-91056 Erlangen, Germany
mailto: tech.team.motioncontrol@siemens.com

8.2 Links and references

Table 8-1

No.	Topic Title
\1\	Siemens Industry Online Support https://support.industry.siemens.com
\2\	Link to the entry page of the application example https://support.industry.siemens.com/cs/ww/en/view/109779415
\3\	

8.3 History

Changes to the application

Table 8-2

Version	Date	Change
1.0	09/2020	First edition
1.1	11/2020	Changed the application examples name to "TIA Add-Ins Getting Started", corrected false product names, Changed the style of the display window from chapter 5, Inserted a note, to set the property "Copy local" to false for the .dll file, changed the text of the TIA context menu (from "Script" to "Add-In Example")
1.2	11/2021	<ul style="list-style-type: none"> • Separation of a large part of chap. 4 into a separate document • Separation of the helper functions of the template into a separate class library • Minor revision in other places • Addition of information related to TIA Portal Openness V17
1.3	06/2022	The Publisher folder has been replaced in the post-build event code by a fixed path in the TIA Portal installation.