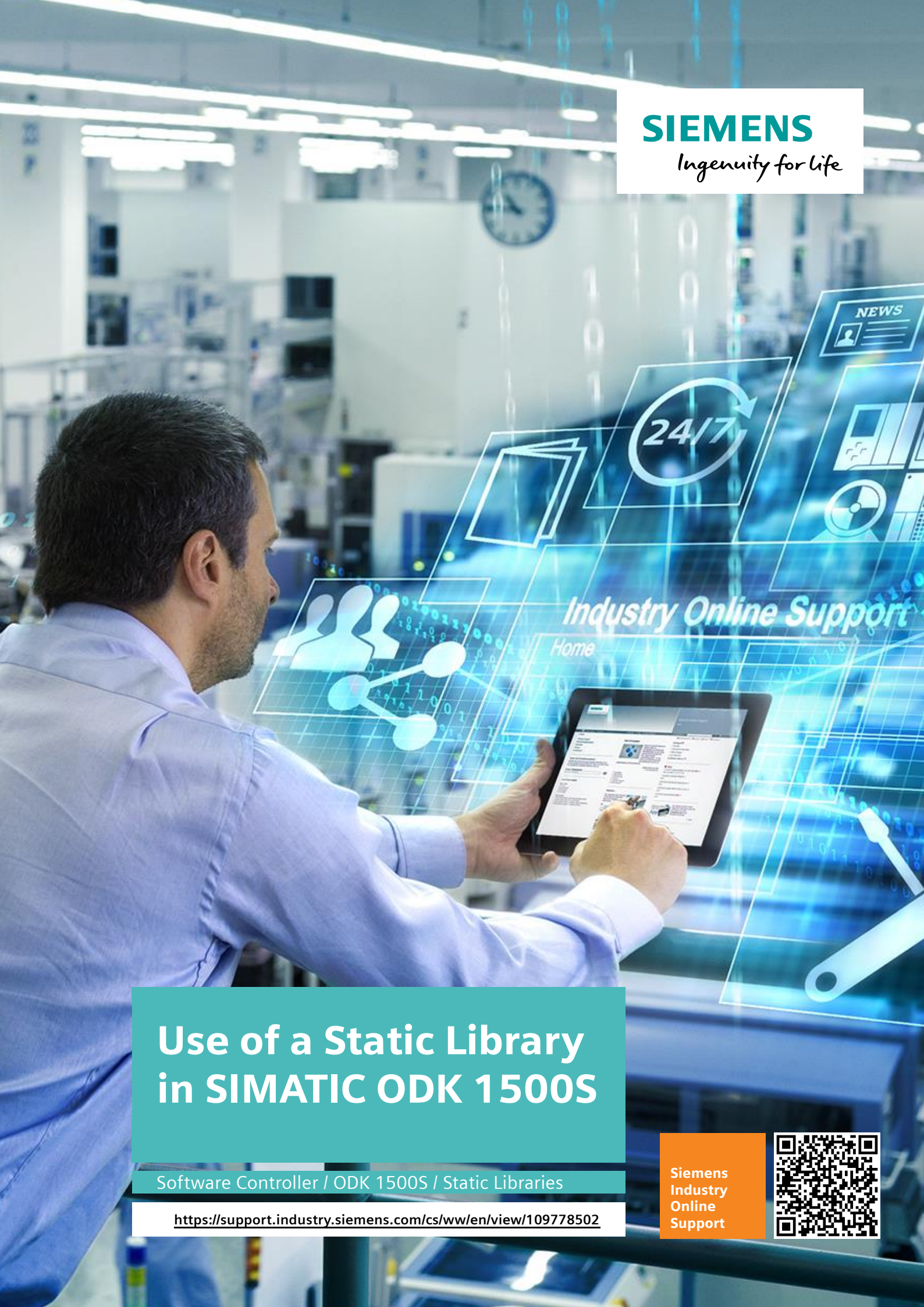


Use of a Static Library in SIMATIC ODK 1500S

Software Controller / ODK 1500S / Static Libraries

https://support.industry.siemens.com/cs/ww/en/view/109778502

Siemens Industry Online Support

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 sand 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.

© Siemens AG 2020 All rights reserved

Table of Contents

Legal information **2**

1 Introduction **4**

1.1 Overview 4
1.2 Description 5
1.3 Important SIMATIC ODK 1500S Information 5

2 Solution **6**

2.1 Applicability 6
2.2 Hardware Components Used 6
2.3 Software Components Used 6
2.4 Example Files and Projects 7
2.5 Hardware Setup 7
2.6 Requirements 7

3 Engineering **8**

3.1 Overview 8
3.2 Creating a Static Library with Eclipse 8
3.3 Adding Headers and Source Files 13
3.4 Integrating a Static Library into a SIMATIC ODK 1500S Project 15
3.5 Using a Static Library in a SIMATIC ODK 1500S Project 20
3.6 SIMATIC ODK 1500S Application in the TIA Portal 23

4 Appendix **26**

4.1 Service and support 26
4.2 Links and Literature 27
4.3 Change Documentation 27

© Siemens AG 2020 All rights reserved

1 Introduction

1.1 Overview

The application example describes the configuration steps for using Static Libraries in a SIMATIC ODK 1500S Real Time Application. You can find an overview of the hardware and software components used and the hardware setup of this application example in Chapter

2 Solution.

So that Static Libraries can be used in SIMATIC ODK 1500S Real Time Applications, it is important that the settings of the compiler exactly match the default compiler settings of the SIMATIC ODK 1500S project.

Which settings are necessary and how to create a Static Library with Eclipse is described in Chapters

3.2 Creating a Static Library with Eclipse and

3.3 Adding Headers and Source Files.

The integration and use of the Static Library in a SIMATIC ODK 1500S project is described in Chapters

3.4 Integrating a Static Library into a SIMATIC ODK 1500S Project and

3.5 Using a Static Library in a SIMATIC ODK 1500S Project.

Finally, Chapter

3.6 SIMATIC ODK 1500S Application in the TIA

shows how to integrate and operate the SIMATIC ODK 1500S application from the TIA Portal application example.

© Siemens AG 2020 All rights reserved

The following functions are defined in the Static Library of this application example:

- Faculty calculation for values between 0 and 32 ($0 \leq x \leq 32$)
- Sine function for user-defined frequency and cycle time
- Triangle function for user-defined frequency and cycle time

Of course, you can use the Static Library in your projects, as well as adapt or extend it for your own needs.

1.2 Description

A Static Library is a Programming Library with functions that are pre-packaged in compiled form. This includes, for example, user-defined modules or sub-programs that are used frequently and repeatedly in various applications.

By integrating Static Libraries into the SIMATIC ODK 1500S environment, you can, for example, incorporate existing functionality from a library and run it in real time on a SIMATIC controller.

Re-engineering your functions in PLC code is not necessary.

1.3 Important SIMATIC ODK 1500S Information

Executability

Not all SIMATIC Controllers support high-level language integration using SIMATIC ODK 1500S. The following list provides an overview of the SIMATIC Controllers that support high-level language integration.

SIMATIC ODK 1500S applications can be run on the following SIMATIC Controller:

- SIMATIC ET 200SP Open Controller PC2
- SIMATIC S7-1500 Software Controller
- SIMATIC S7-1500 CPU 1518 MFP

Compatibility

Table 1-1 shows the compatibility between SIMATIC ODK 1500S versions and SIMATIC S7-1500 CPU firmware versions.

Table 1-1

SIMATIC ODK 1500S Version	SIMATIC S7-1500 CPU firmware[1]	Compatibility
V2.0	< V2.0	✗
V2.0	≥ V2.0	✓
V2.5	< V2.5	✗
V2.5	≥ V2.5	✓

Note The SIMATIC ODK 1500S version used requires **at minimum** the same CPU firmware version. Backwards compatibility is not possible! \4\ \5\

[1] Valid for SIMATIC S7-1500 software and hardware Controllers.

© Siemens AG 2020 All rights reserved

2 Solution

2.1 Applicability

The application example is valid for the components and versions listed in Table 2-1.

Table 2-1

Components	Version
SIMATIC ODK 1500S	as of V2.5
SIAMTIC ET 200SP Open Controller PC2	as of CPU FW V2.5
SIMATIC S7-1500 Software Controller	as of CPU FW V2.5
SIMATIC S7-1500 CPU 1518 MFP	as of CPU FW V2.5
SIMATIC STEP 7 TIA Portal	as of V15

2.2 Hardware Components Used

The application example was created with the hardware components listed in Table 2-2.

Table 2-2

Components	Article number	Note
SIMATIC IPC 427E	6AG4141-.....-....	**Engineering Station** with • SIMATIC ODK 1500S • SIMATIC STEP 7 TIA Portal
SIMATIC ET 200SP Open Controller PC2 CPU 1515SP PC2	6ES7677-2DB42-0GB0	**Runtime Station**[2] with • CPU 1505SP

2.3 Software Components Used

The application example was created with the software components listed in Table 2-3.

Table 2-3

Components	Article number	Note
SIMATIC STEP 7 Professional V16	6ES7822-1..06-..	TIA Portal V16
SIMATIC ODK 1500S	6ES7806-2CD03-0YA0	SIMATIC ODK 1500S V2.5.1
SIMATIC S7-1500 Software Controller	6ES7672-5DC11-0YA0	CPU 1505SP V2.6

Note Siemens Industry Online Support offers you SIMATIC ODK 1500S V2.5 as a 21-day trial version. \3\

Download link: https://support.industry.siemens.com/cs/ww/en/view/109752598

[2] Alternatives: See Chapter 1.3

© Siemens AG 2020 All rights reserved

2.4 Example Files and Projects

Table 2-4 lists the names of the example data for this application example.

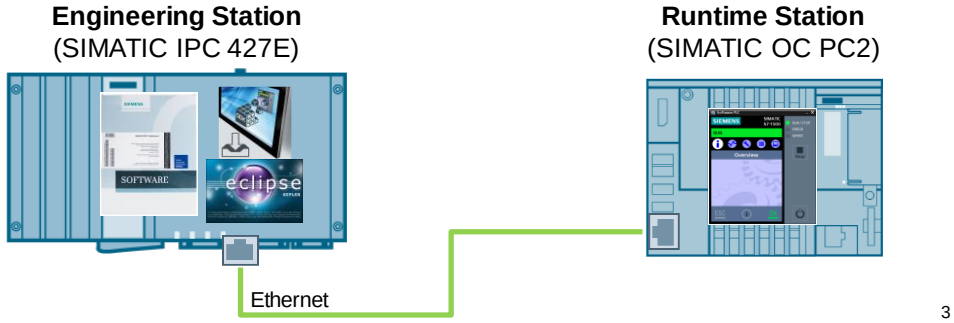
Table 2-4

Components	Note
109778502_StaticLibraries_ODK1500S_Projects_V10.zip	Contains the Eclipse Projects • Static Library • SIMATIC ODK 1500S Project
109778502_StaticLibraries_ODK1500S_TIAPV16_PROJ.zip	TIA Portal Project archive
109778502_StaticLibraries_ODK1500S_DOC_V10_EN.pdf	This document

2.5 Hardware Setup

Figure 2-1 displays the general hardware setup of the application example. At the **Engineering Station** (left), the Static Library is created with Eclipse and integrated into a SIMATIC ODK 1500S project. Then the TIA Portal Project is programmed with the generated code files from SIMATIC ODK 1500S before the project is loaded and run on the **Runtime Station** (right).

Figure 2-1



[3]

2.6 Requirements

In order to be able to implement the application example, you need experience in using:

- SIMATIC industrial PCs
- Windows operating system
- SIMATIC STEP 7 TIA Portal
- SIMATIC S7-1500 Software Controller
- SIMATIC ODK 1500S
- Eclipse IDE

You can find further information on the internet at:

- www.siemens.com/pc-based
- https://support.industry.siemens.com/cs/ww/en/view/75852684

[3] SIMATIC OC PC2 = SIMATIC ET 200SP Open Controller PC2

© Siemens AG 2020 All rights reserved

3 Engineering

3.1 Overview

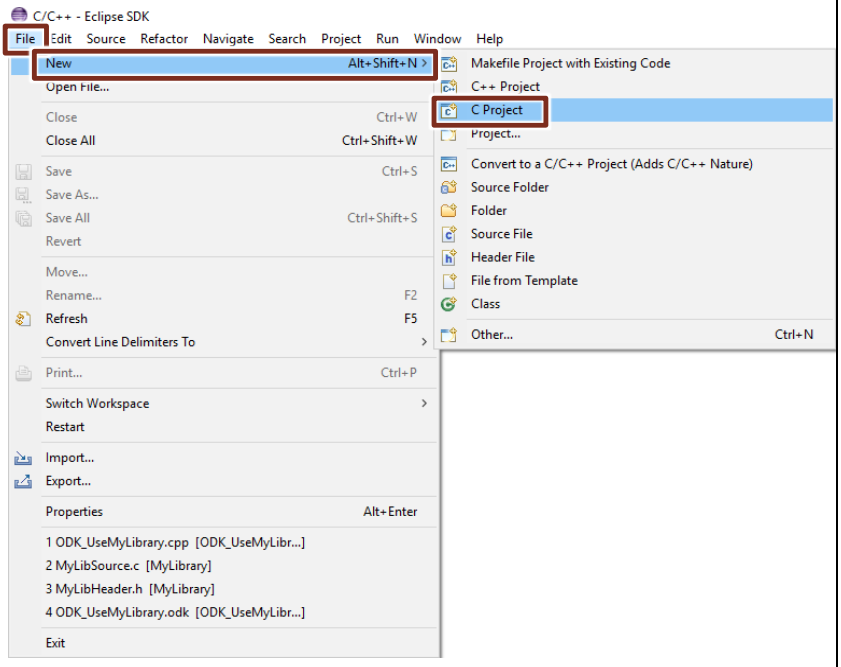
This chapter shows you the general workflow for creating a Static Library of source files with Eclipse and using them in a SIMATIC ODK 1500S Real Time Project.

Note The steps and screenshots shown refer to Eclipse Kepler, which is included in the SIMATIC ODK 1500S scope of delivery.

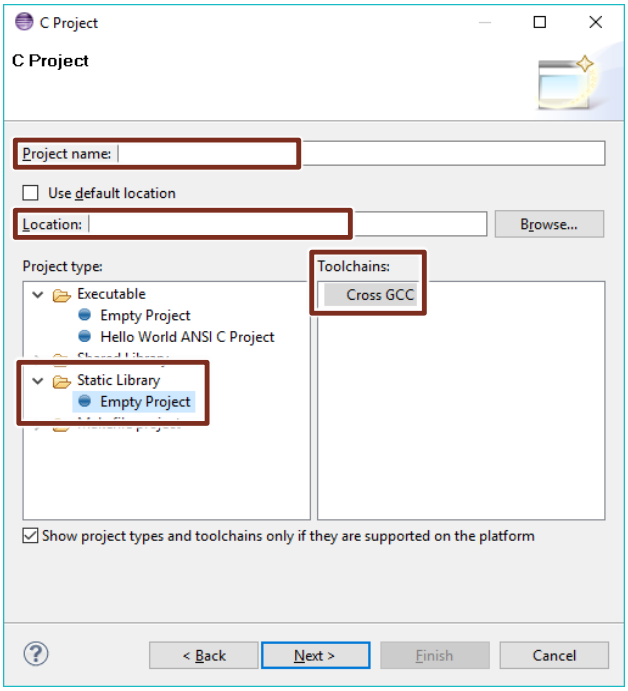
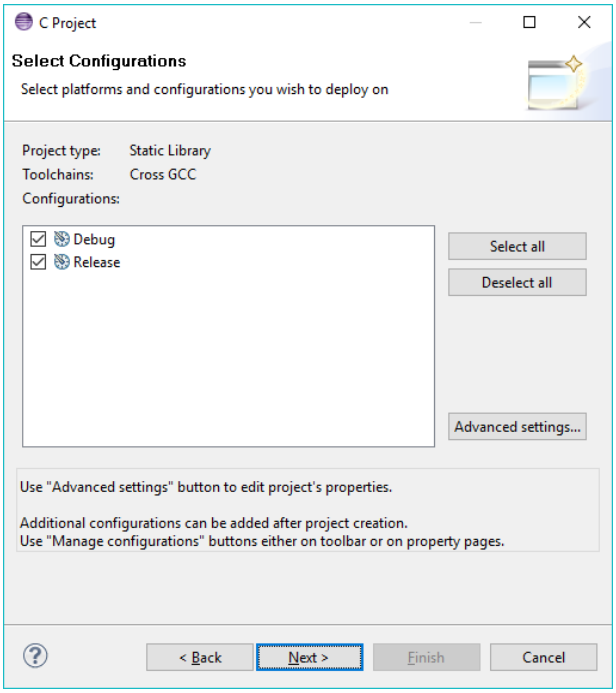
3.2 Creating a Static Library with Eclipse

Table 3-1 describes how to create a new empty Static Library in Eclipse and which settings to implement.

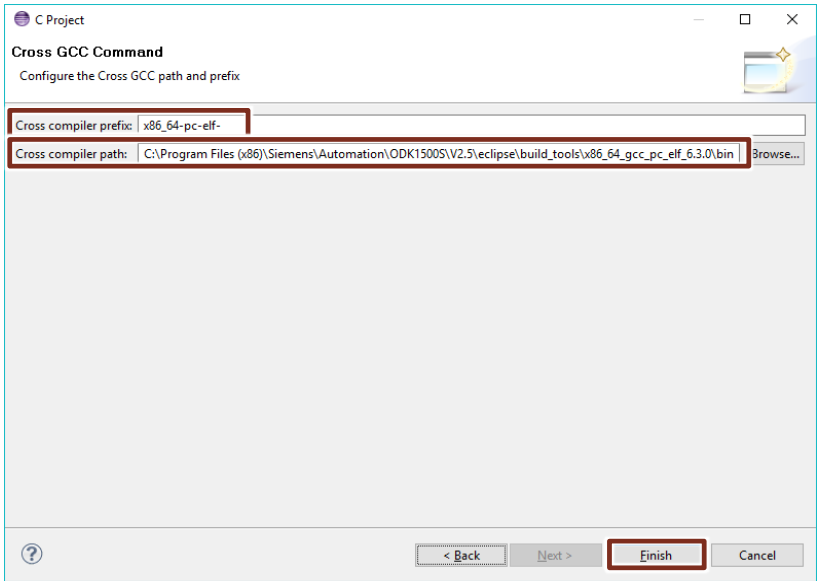
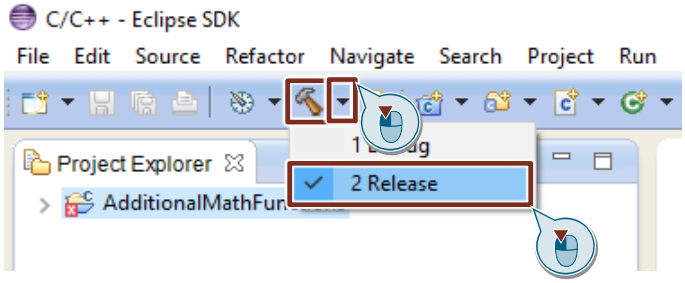
Table 3-1

Step	Action
1.	Start **Eclipse for ODK 1500S V2.5** by using the desktop icon. 
2.	Create a new C project under **File > New > C Project**. 

© Siemens AG 2020 All rights reserved

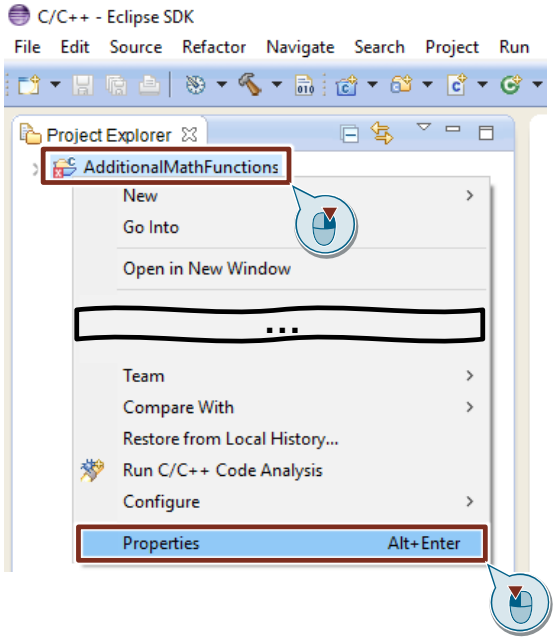
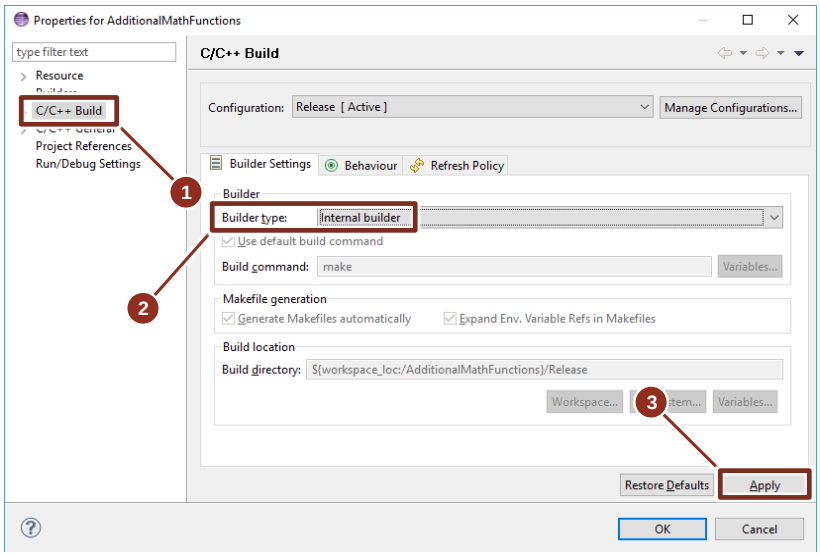
Step	Action
3.	General settings for the C project. • **Project name** > name of the Static Library • **Location** > storage location of the Library • Select **Project type > Static Library > Empty Project** • Select **Toolchains > Cross GCC** 
4.	Configuration settings for the C project. • Do not change the default settings in this window 

© Siemens AG 2020 All rights reserved

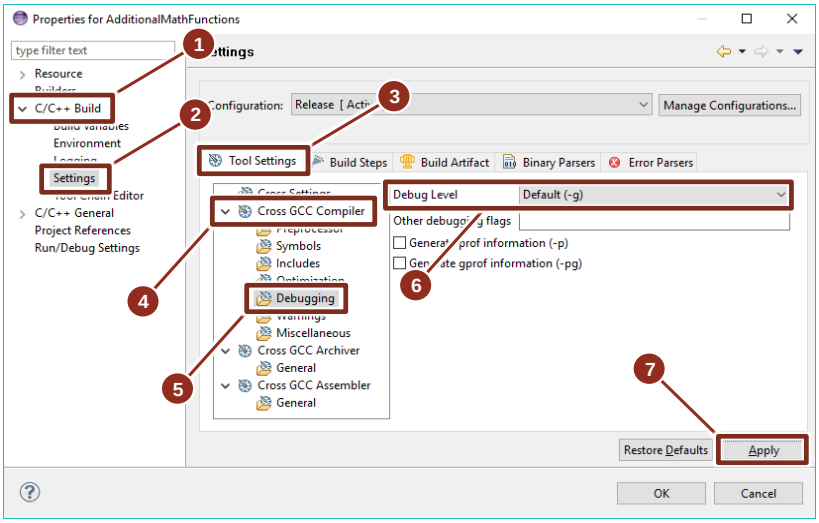
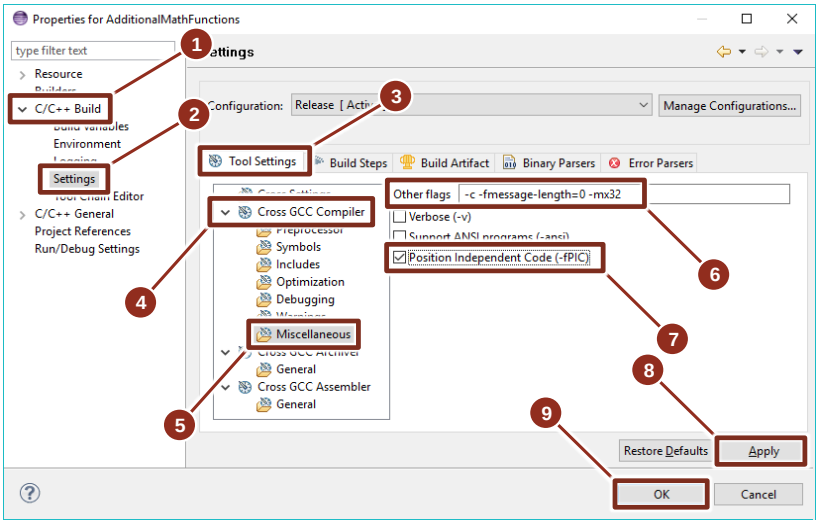
Step	Action
5.	Compiler settings for the C project. • **Cross compiler prefix > x86_64-pc-elf-** • **Cross compiler path > …\ODK1500S\V2.5\eclipse\build_tools\x86_64_gcc_pc_elf_6.3.0\bin**[4] • Finally, click **Finish** 
6.	Changing the build setting for the C project. • **Left-click** the **arrow** next to the Hammer symbol to open the **Release Menu** • **Left-click** to enable the **Release** setting 

[4] The Cross Compiler Path depends on the system's installation directory and may differ accordingly.

© Siemens AG 2020 All rights reserved

Step	Action
7.	Open the Project Properties of the C project. • **Right-click** the project in the Project Explorer • **Left-click** “Properties” in the context menu 
8.	Builder settings for the C project. • Open the **C/C++ Build** Properties • Select **Builder type > Internal builder** • Finally, click **Apply** 

© Siemens AG 2020 All rights reserved

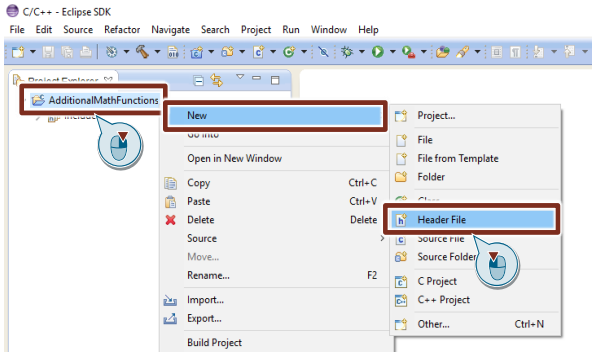
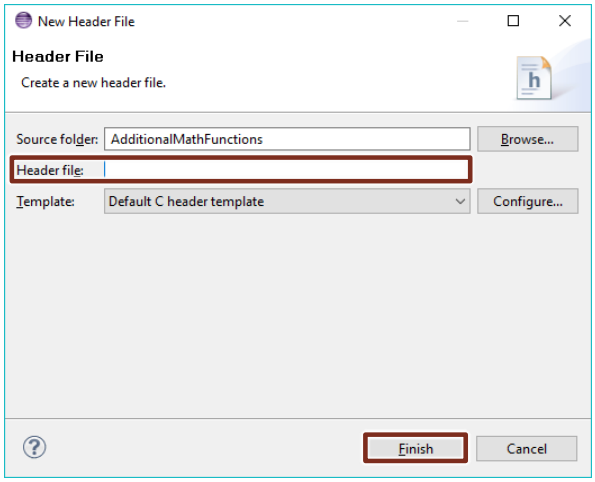
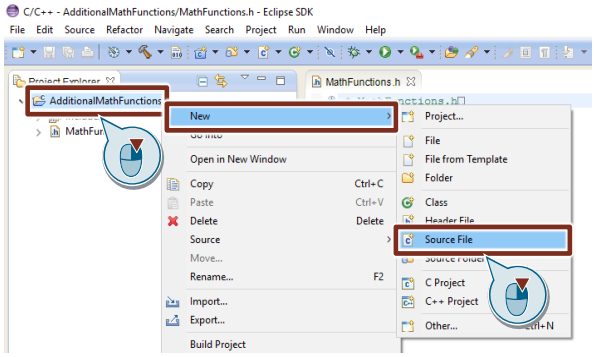
Step	Action
9.	Debugging settings for the C project. • Open the **C/C++ Build** menu • Open the **Settings** • Open the **Tool Settings** tab • Open **Cross GCC Compiler > Debugging** • **Debug Level > Default (-g)** • Click **Apply** to apply the settings 
10.	Miscellaneous settings for the C project. • Open the **C/C++ Build** menu • Open the **Settings** • Open the **Tool Settings** tab • Open **Cross GCC Compiler > Miscellaneous** • **Other flags > -c -fmessage-length=0 -mx32** • Check the **Position Independent Code (-fPIC)** check box • Click **Apply** to apply the settings • Finally, click **OK** 

© Siemens AG 2020 All rights reserved

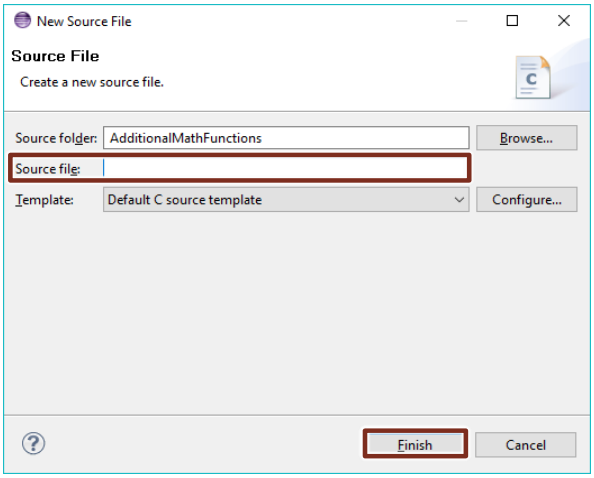
3.3 Adding Headers and Source Files

Table 3-2 describes how to add header and source files to a Static Library when using Eclipse.

Table 3-2

Step	Action
1.	Adding a header file. • **Right-click** the project in the Project Explorer • **Left-click Header File** in the context menu 
2.	Header file settings. • Enter a name for the header file including the file extension (*.h) • Finally, click **Finish** 
3.	Adding a source file • **Right-click** the project in the Project Explorer • **Left-click Source File** in the context menu 

© Siemens AG 2020 All rights reserved

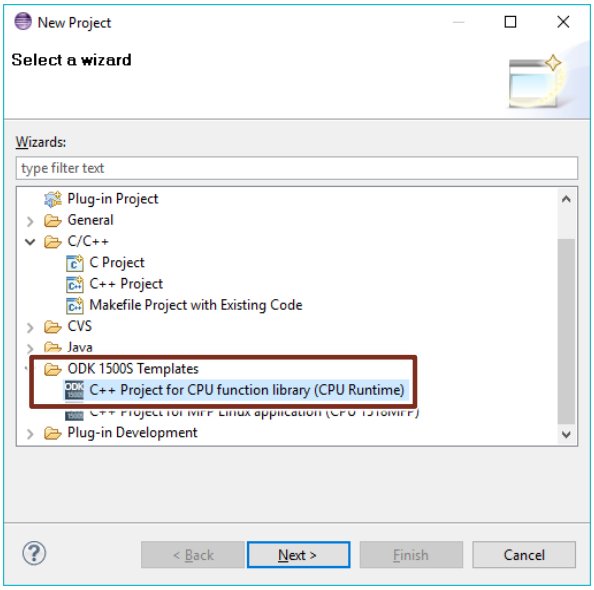
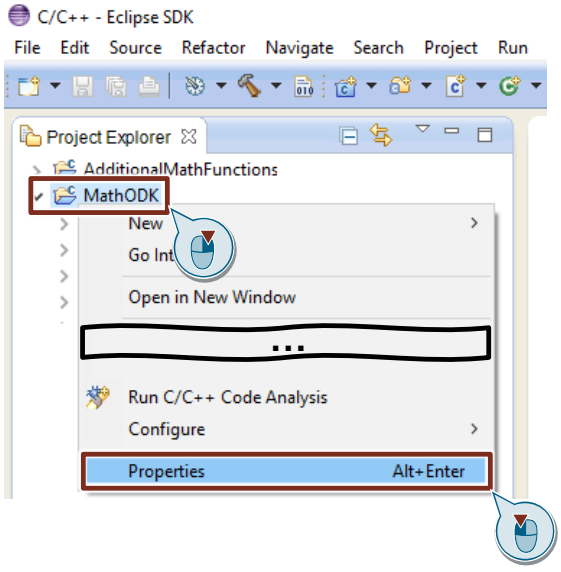
Step	Action
4.	Source file settings. • Enter a name for the source file including the file extension (*.c) • Finally, click **Finish** 
5.	Define your customer-specific functions in the header file.
6.	Implement your customer-specific functions in the source file.
7.	Finally, you must compile the project. A successful compilation generates an *.a file (Library file) in the Release folder of the project directory.

© Siemens AG 2020 All rights reserved

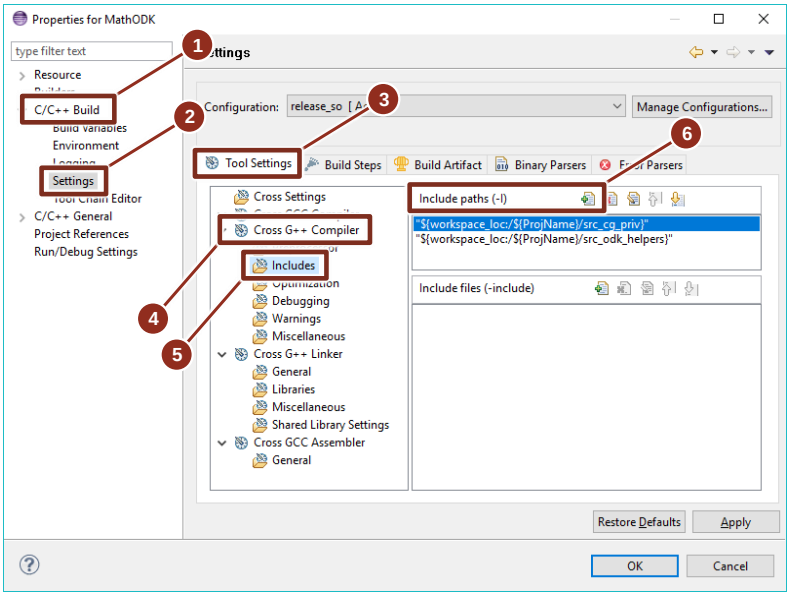
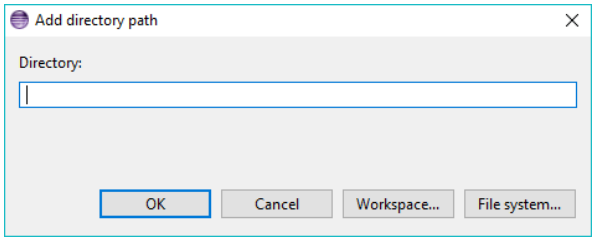
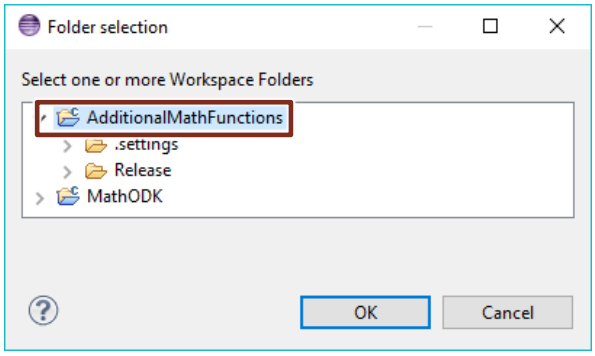
3.4 Integrating a Static Library into a SIMATIC ODK 1500S Project

Table 3-3 shows the integration of a Static Library into a SIMATIC ODK 1500S Real Time Application.

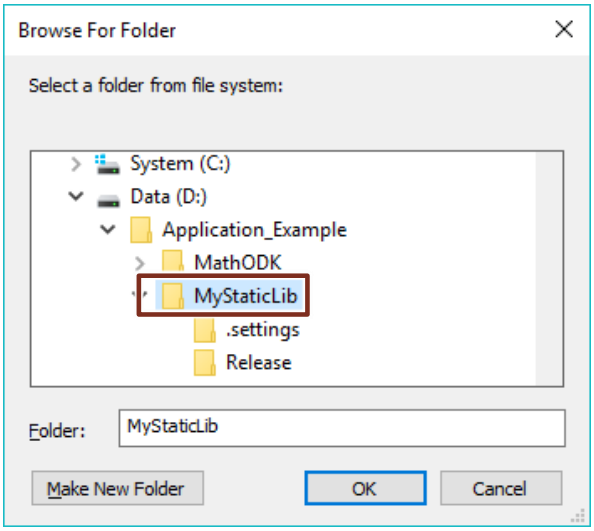
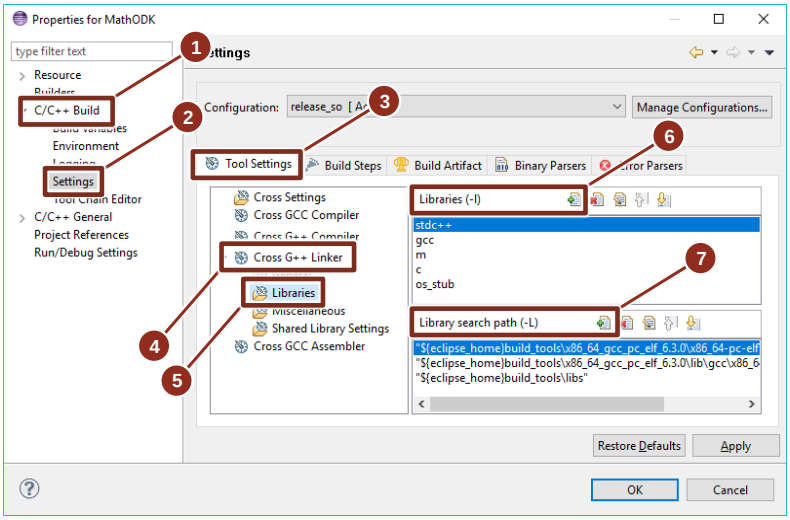
Table 3-3

Step	Action
1.	Open **Eclipse for ODK 1500S V2.5** and create a new project for a SIMATIC ODK 1500S Real Time Application. The Wizard assists you in the project creation. 
2.	Open the Project Properties of the ODK project. • **Right-click** the project in the Project Explorer • **Left-click** “Properties” in the context menu 

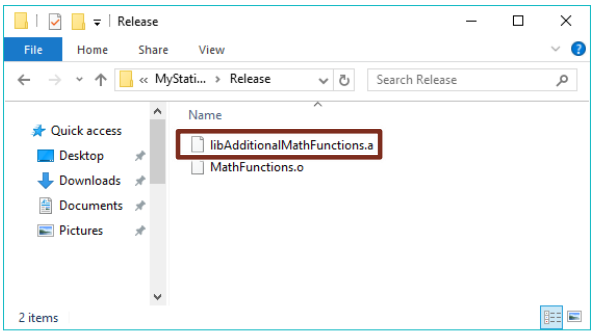
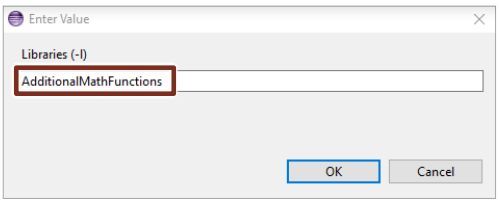
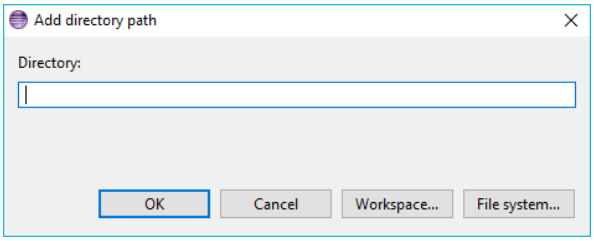
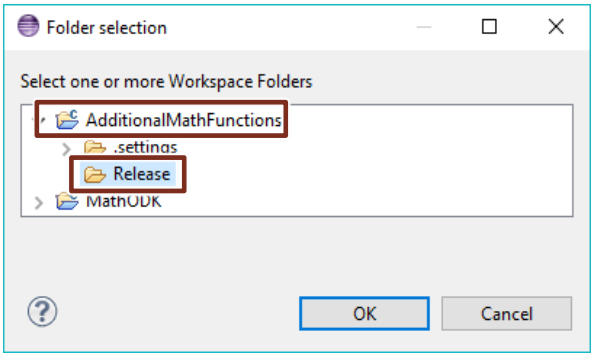
© Siemens AG 2020 All rights reserved

Step	Action
3.	Include settings for the ODK project. • Open the **C/C++ Build** menu • Open the **Settings** • Open the **Tool Settings** tab • Open **Cross G++ Compiler > Includes** • Click the **Add** button under **Include paths (-I)** 
4.	The storage location of the header file must be specified.  Choose one of two possibilities: • Storage location in the Eclipse **Workspace** (step 5) • Storage location in the **File system** (step 6)
5.	Click **Workspace** to open the Eclipse Workspace overview. Select the project folder of the Static Library. 

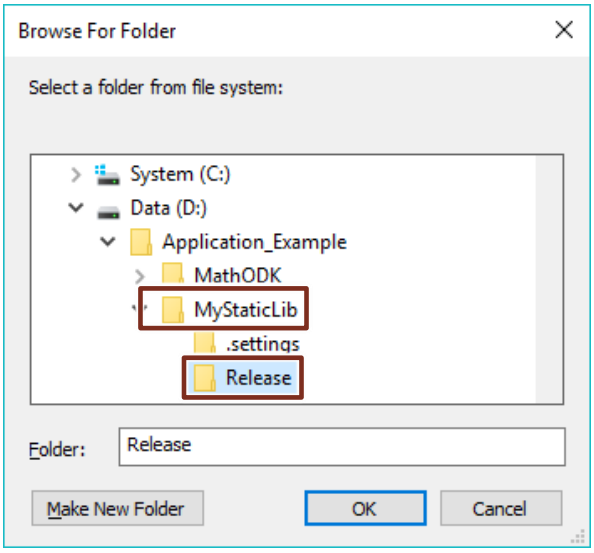
© Siemens AG 2020 All rights reserved

Step	Action
6.	Click **File system** to open the Windows Explorer. Select the project folder of the Static Library. 
7.	After adding the storage location of the header file, click **Apply** in the Settings menu (Figure from step 3).
8.	Library settings for the ODK project. • Open the **C/C++ Build** menu • Open the **Settings** • Open the **Tool Settings** tab • Open **Cross G++ Linker > Libraries** • Click the **Add** button under **Libraries (-l)** (step 9) • Click the **Add** button under **Library search path (-L)** (step 10) 

© Siemens AG 2020 All rights reserved

Step	Action
9.	The name of the Static Library must be given without the **lib** suffix and with the ***.a** file extension.  
10.	The storage location of the Library must be specified.  Choose **one** of two possibilities: • Storage location in the Eclipse **Workspace** (step 11) • Storage location in the **File system** (step 12)
11.	Click **Workspace** to open the Eclipse Workspace overview. Select the Release folder of the Static Library. 

© Siemens AG 2020 All rights reserved

Step	Action
12.	Click **File system** to open the Windows Explorer. Select the Release folder of the Static Library. 
13.	After adding the Static Library storage location, click **Apply** in the Settings menu (Figure from step 8).

© Siemens AG 2020 All rights reserved

3.5 Using a Static Library in a SIMATIC ODK 1500S Project

The requirement for using a user-defined Static Library in a SIMATIC ODK 1500S project is successful integration. This procedure is described in Chapter 3.4.

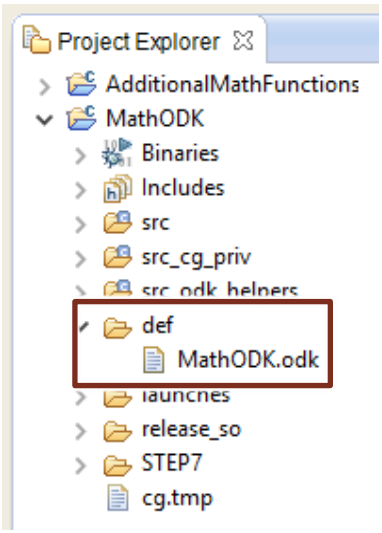
Note

This chapter does not cover the programming of a SIMATIC ODK 1500S project.

You can find information on operation and programming with SIMATIC ODK 1500S in the SIMATIC ODK 1500S manual. \4\ \5\

First, you must specify the interface definition in the SIMATIC ODK 1500S project for the three different functions. The ***.odk file** in the **def folder** is available for the interface definition.

Figure 3-1



Three functions have been created for the application example, which are also available in the TIA Portal as three different FBs.

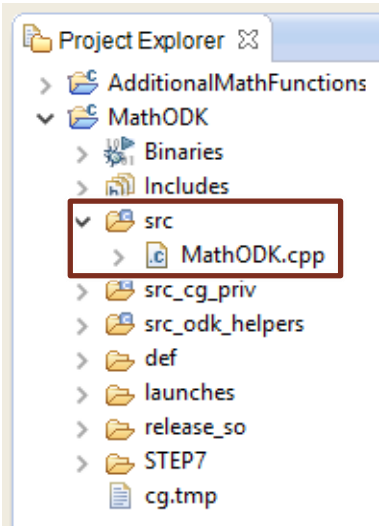
Figure 3-2

```
// Basic function in order to show

ODK_RESULT StaticLibFacultyFunction(    [IN]    ODK_INT16           FacultyInput
                                   ,    [OUT]   ODK_UINT64          FacultyResult
                                   ,    [OUT]   ODK_S7STRING[15]    Info);

ODK_RESULT StaticLibSinusFunction(      [IN]    ODK_FLOAT       OBCycle
                                  ,     [IN]    ODK_FLOAT       freq
                                  ,     [IN]    ODK_FLOAT       offset
                                  ,     [IN]    ODK_FLOAT       amplitude
                                  ,     [OUT]   ODK_FLOAT       SinusValue);

ODK_RESULT StaticLibTriangleFunction(   [IN]    ODK_FLOAT       OBCycle
                                    ,   [IN]    ODK_FLOAT       freq
                                    ,   [IN]    ODK_FLOAT       offset
                                    ,   [IN]    ODK_FLOAT       amplitude
                                    ,   [IN]    ODK_FLOAT       ratio
                                    ,   [OUT]   ODK_FLOAT       TriangleValue);
```

Open the ***.cpp file**, which is located in the **src folder**.

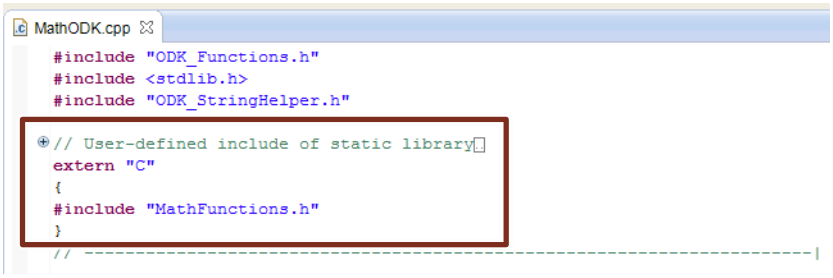
© Siemens AG 2020 All rights reserved

Figure 3-3



The header of the Static Library must be added to the ***.cpp file**. Otherwise, the functions from the Static Library are unrecognized.

Figure 3-4



Note

In this application example, the Static Library header files (*.h) are named **MathFunctions.h**.

The name of the header file can differ depending on your Library.

© Siemens AG 2020 All rights reserved

Before you can compile the SIMATIC ODK 1500S project, you must program the logic of the respective functions in the ***.cpp file**.

To understand how to use the Static Library, the Function Calls of the three functions from the created Static Library in the application example are underlined in red in Figure 3-5.

Figure 3-5

```
// Call of three ODK functions with internal call of functions from static library
ODK_RESULT StaticLibFacultyFunction(const ODK_INT16& FacultyInput, ODK_UINT64& FacultyResult
        ,ODK_S7STRING Info[15])
{
    // Local declaration of buffer for info of FB in PLC
    char BufferOK[15] = "Valid Value!";
    char BufferNOTOK[15] = "Invalid Value!";

    // IF-LOOP to check range of FacultyInput
    if((FacultyInput >= 0) && (FacultyInput <= 32))
    {
        // Call Faculty function from static library
        FacultyResult = Faculty(FacultyInput);
        Convert_SZSTR_to_S7STRING(BufferOK,Info);
    }
    else
    {
        FacultyResult = Faculty(FacultyInput);
        Convert_SZSTR_to_S7STRING(BufferNOTOK,Info);
    }

    // Copy to output
    return ODK_SUCCESS;
}
```

```
ODK_RESULT StaticLibSinusFunction(const ODK_FLOAT& OBCycle, const ODK_FLOAT& freq,
        const ODK_FLOAT& offset, const ODK_FLOAT& amplitude, ODK_FLOAT& SinusValue)
{
    // Call Sinus function from static library
    SinusValue = Sinus(OBCycle, freq, offset, amplitude);

    // Copy to output
    return ODK_SUCCESS;
}
```

```
ODK_RESULT StaticLibTriangleFunction(const ODK_FLOAT& OBCycle, const ODK_FLOAT& freq,
        const ODK_FLOAT& offset, const ODK_FLOAT& amplitude, const ODK_FLOAT& ratio,
        ODK_FLOAT& TriangleValue)
{
    // Call TriangleSinusValue from static library
    TriangleValue = Triangle(OBCycle, freq, offset, amplitude, ratio);

    // Copy to output
    return ODK_SUCCESS;
}
```

If you have completed these four steps in the SIMATIC ODK 1500S project, you can compile the entire SIMATIC ODK 1500S project.

© Siemens AG 2020 All rights reserved

3.6 SIMATIC ODK 1500S Application in the TIA Portal

The compilation delivers several files, two of which must be processed further.

Firstly, you receive a ***.scl file** that provides the interface information for the Function Blocks in the TIA Portal. Therefore, it must be imported into the TIA Portal project. You can then right-click the file to generate the Function Blocks in the TIA Portal. You must then call the FBs in the Organization Blocks of the TIA Portal project.

The second file is the ***.so file**. It contains the “logic” of the SIMATIC ODK 1500S application. This file must be located in the **File Browser directory** of the SIMATIC Controller's PLC Web Server; otherwise, the SIMATIC ODK 1500S application will not be functional.

In the enclosed TIA Portal project, the blocks are already generated from the ***.scl file** and called in the OBs. The TIA Portal project contains four different groups in the **Program Blocks** folder (Table 3-4).

Table 3-4

Group	Contents
Blocks_Global	Global blocks for the entire TIA Portal project
ODK_Faculty	Blocks for the Faculty Function
ODK_Sine	Blocks for the Sine Function
ODK_Triangle	Blocks for the Triangle function

Note

To unload and load the SIMATIC ODK 1500S Real Time Application ***.so file**, the corresponding Unload and Load FBs must be accessed in the Startup OB of the PLC program.

All global blocks used throughout the entire TIA Portal project are stored in the **Blocks_Global** group.

© Siemens AG 2020 All rights reserved

ODK_Faculty group

Faculty Function description

The function returns the factorial of an integer between 0 and 32.

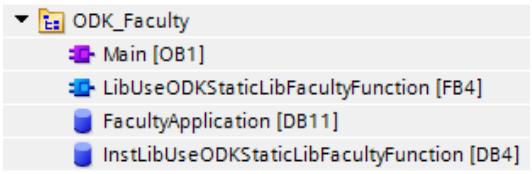
- INPUTS
  - Integer for the calculation
- OUTPUTS
  - Result
  - Result is 99 if invalid value was entered

In the **Main** OB, the FB **LibUseODKStaticLibFacultyFunction** is called and is linked to the data of the DB **FacultyApplication**.

With the online observation in the DB, you can change the values of the factorial function input variable and display the calculation result.

Note: If an integer less than 0 and greater than 32 is entered, the error value 99 is returned with the message “Invalid Value!”.

Figure 3-6



ODK_Sine group

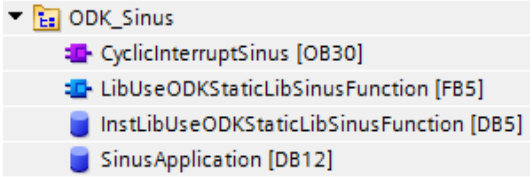
Sine Function description

This function returns a sine.

- INPUTS
  - OBCycle : Value of the time interval of the OB of the PLC in seconds [s]
  - freq : Frequency in Hertz [Hz]
  - offset : Offset
  - amplitude : Amplitude
- OUTPUTS
  - Sine wave

The FB **LibUseODKStaticLibSineFunction** is called in the OB **CyclicInterruptSine** at a time interval of 2000µsec. In the DB **SineApplication**, you can change the input values of the function, such as frequency, time interval, and amplitude during online observation, as well as view the sine values. Comments explain the different variables in the DB.

Figure 3-7



© Siemens AG 2020 All rights reserved

ODK_Triangle group

Triangle Function description

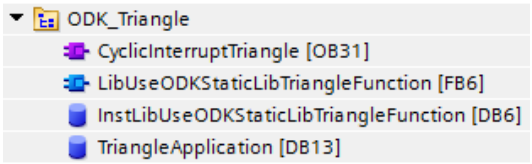
This function returns a triangle.

- INPUTS
  - OBCycle : Value of the time interval of the OB of the PLC in seconds [s]
  - freq : Frequency in Hertz [Hz]
  - offset : Offset
  - amplitude : Amplitude
  - ratio : Value of the ratio in percent [%]
- OUTPUTS
  - Triangle wave
  - Result is 99 if OBCycle, freq or ratio is zero

Here, the FB **LibUseODKStaticLibTriangleFunction**—analogous to the **ODK_Sine** group—is called in the OB **CyclicInterruptTriangle** at a time interval of 1000µsec. In the DB **TriangleApplication**, the input values of the function, such as frequency, time interval, amplitude, and comments are again available and can be changed accordingly.

Note: If less than or equal to 0 is entered for OBCycle, Frequency, or Ratio, the error value 99 is returned.

Figure 3-8



Additional traces in the project

Two additional traces are available in the TIA Portal project so that you can check the course of the Sine and Triangle Functions. Both trace functions record the course of the Sine or Triangle function over a period of 20 seconds. **TraceSine** records the signal of the Sine Function and **TraceTriangle** records the signal of the Triangle Function.

Figure 3-9



© Siemens AG 2020 All rights reserved

4 Appendix

4.1 Service and support

Industry Online Support

Do you have any questions or need assistance?

Siemens Industry Online Support offers round the clock access to our entire service and support know-how and portfolio.

The Industry Online Support is the central address for information about our products, solutions and services.

Product information, manuals, downloads, FAQs, application examples and videos – all information is accessible with just a few mouse clicks:
https://support.industry.siemens.com

Technical Support

The Technical Support of Siemens Industry provides you fast and competent support regarding all technical queries with numerous tailor-made offers – ranging from basic support to individual support contracts. Please send queries to Technical Support via Web form:
www.siemens.com/industry/supportrequest

SITRAIN – Training for Industry

We support you with our globally available training courses for industry with practical experience, innovative learning methods and a concept that's tailored to the customer's specific needs.

For more information on our offered trainings and courses, as well as their locations and dates, refer to our web page:
www.siemens.com/sitrain

Service offer

Our range of services includes the following:

- Plant data services
- Spare parts services
- Repair services
- On-site and maintenance services
- Retrofitting and modernization services
- Service programs and contracts

You can find detailed information on our range of services in the service catalog web page:
https://support.industry.siemens.com/cs/sc

Industry Online Support app

You will receive optimum support wherever you are with the "Siemens Industry Online Support" app. The app is available for iOS and Android:
https://support.industry.siemens.com/cs/ww/en/sc/2067

© Siemens AG 2020 All rights reserved

4.2 Links and Literature

Table 4-1

No.	Subject
\1\	Siemens Industry Online Support https://support.industry.siemens.com
\2\	Link to the article page of the Application Example https://support.industry.siemens.com/cs/ww/en/view/109778502
\3\	SIMATIC ODK 1500S V2.5 Trial Download https://support.industry.siemens.com/cs/ww/en/view/109752598
\4\	SIMATIC ODK 1500S manual https://support.industry.siemens.com/cs/ww/en/view/109752683
\5\	SIMATIC ODK 1500S manual supplement https://support.industry.siemens.com/cs/ww/en/view/109752687
\6\	SIMATIC S7-1500 CPU 1518 MFP https://support.industry.siemens.com/cs/ww/en/view/109749061
\7\	SIMATIC S7-1500 Software Controller manual https://support.industry.siemens.com/cs/ww/en/view/109762855
\8\	SIMATIC ET 200SP Open Controller PC 2 manual https://support.industry.siemens.com/cs/ww/en/view/109758551

4.3 Change Documentation

Table 4-2

Version	Date	Change
V1.0	08/2020	First version

© Siemens AG 2020 All rights reserved