PROCESS AUTOMATION

Freelance 2019

Engineering Manual
System Configuration

PROCESS AUTOMATION

Freelance 2019

Engineering Manual
System Configuration

Document Number: 3BDD012503-111
Revision: A
Release: January 2019

Notice

This document contains information about one or more ABB products and may include a description of or a reference to one or more standards that may be generally relevant to the ABB products. The presence of any such description of a standard or reference to a standard is not a representation that all of the ABB products referenced in this document support all of the features of the described or referenced standard. In order to determine the specific features supported by a particular ABB product, the reader should consult the product specifications for the particular ABB product.

ABB may have one or more patents or pending patent applications protecting the intellectual property in the ABB products described in this document.

The information in this document is subject to change without notice and should not be construed as a commitment by ABB. ABB assumes no responsibility for any errors that may appear in this document.

Products described or referenced in this document are designed to be connected, and to communicate information and data via a secure network. It is the sole responsibility of the system/product owner to provide and continuously ensure a secure connection between the product and the system network and/or any other networks that may be connected.

The system/product owners must establish and maintain appropriate measures, including, but not limited to, the installation of firewalls, application of authentication measures, encryption of data, installation of antivirus programs, and so on, to protect the system, its products and networks, against security breaches, unauthorized access, interference, intrusion, leakage, and/or theft of data or information.

ABB verifies the function of released products and updates. However system/product owners are ultimately responsible to ensure that any system update (including but not limited to code changes, configuration file changes, third-party software updates or patches, hardware change out, and so on) is compatible with the security measures implemented. The system/product owners must verify that the system and associated products function as expected in the environment they are deployed.

In no event shall ABB be liable for direct, indirect, special, incidental or consequential damages of any nature or kind arising from the use of this document, nor shall ABB be liable for incidental or consequential damages arising from use of any software or hardware described in this document.

This document and parts thereof must not be reproduced or copied without written permission from ABB, and the contents thereof must not be imparted to a third party nor used for any unauthorized purpose.

The software or hardware described in this document is furnished under a license and may be used, copied, or disclosed only in accordance with the terms of such license. This product meets the requirements specified in EMC Directive 2014/30/EU and in Low Voltage Directive 2014/35/EU.

Trademarks

All rights to copyrights, registered trademarks, and trademarks reside with their respective owners.

Table of contents

3 - Project tree

4 - Hardware structure

5 - Commissioning

6 - Documentation

About this book

Use of warning, caution, information, and tip icons

This publication includes **Warning**, **Caution**, and **Information** where appropriate to point out safety related or other important information. It also includes **Tip** to point out useful hints to the reader. The corresponding symbols should be interpreted as follows:

Electrical warning icon indicates the presence of a hazard which could result in *electrical shock.*

Warning icon indicates the presence of a hazard which could result in *personal injury.*

Caution icon indicates important information or warning related to the concept discussed in the text. It might indicate the presence of a hazard which could result in *corruption of software or damage to equipment/property.*

Information icon alerts the reader to pertinent facts and conditions.

Tip icon indicates advice on, for example, how to design your project or how to use a certain function

Although **Warning** hazards are related to personal injury, and **Caution** hazards are associated with equipment or property damage, it should be understood that operation of damaged equipment could, under certain operational conditions, result in degraded process performance leading to personal injury or death. Therefore, comply fully with all **Warning** and **Caution** notices.

Terminology

The Glossary contains terms and abbreviations that are unique to ABB or have a usage or definition that is different from standard industry usage. Please make yourself familiar to that.

You will find the glossary at the end of this manual.

Document conventions

The following conventions are used for the presentation of material:

- The words in names of screen elements (for example, the title in the title bar of a window, the label for a field of a dialog box) are initially capitalized.
- Capital letters are used for the name of a keyboard key if it is labeled on the keyboard. For example, press the ENTER key.
- Lowercase letters are used for the name of a keyboard key that is not labeled on the keyboard. For example, the **space bar**, **comma key**, and so on.
- Press CTRL+C indicates that you must hold down the CTRL key while pressing the C key (to copy a selected object in this case).
- Press **ESC, E, C** indicates that you press and release each key in sequence (to copy a selected object in this case).
- The names of push and toggle buttons are boldfaced. For example, click **OK**.
- The names of menus and menu items are boldfaced. For example, the **File** menu.
  - The following convention is used for menu operations: MenuName > MenuItem > CascadedMenuItem. For example: select **File > New > Type**.
  - The **Start** menu name always refers to the **Start** menu on the Windows Task Bar.

- System prompts/messages are shown in the Courier font, and user responses/input are in the boldfaced Courier font. For example, if you enter a value out of range, the following message is displayed:

  ```
  Entered value is not valid. The value must be 0 to 30.
  ```

  You may be told to enter the string TIC132 in a field. The string is shown as follows in the procedure:

  TIC132

  Variables are shown using lowercase letters.

  sequence name

1 Overview

1.1 Freelance system

The Freelance system contains automation functions, graphics, the operator interface, and fieldbus lines (PROFIBUS, FOUNDATION Fieldbus, HART and so on). The integrated environment facilitates engineering, commissioning, maintenance and fieldbus management. The direct operator interface enables easy operation and diagnostics of the entire system.

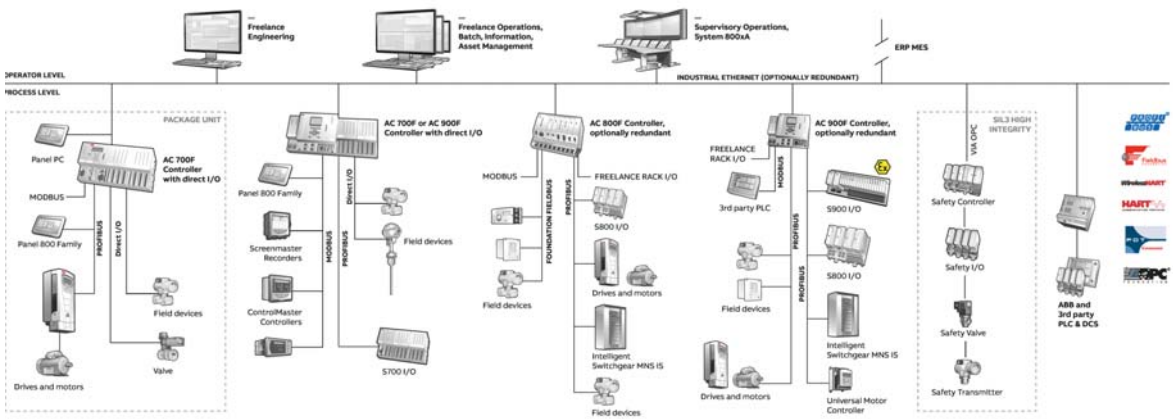
1.1.1 General description - Freelance system

A Freelance system provides an operator level and process level. The operator level contains the functions for operation and observation, archives and logs, trends and alarms. The tool used for Freelance operator stations is Freelance Operations (FO). Also an OPC gateway (Freelance OPC server) is available, which allows OPC clients to access data and alarms from the Freelance controllers.

The process level contains the open-loop and closed-loop control functions. The control functions are processed in the controllers which communicate with actors and sensors in the field. Freelance supports a various number of controllers like the AC 900F, AC 800F and AC 700F.

The tool used to perform the whole configuration and commissioning of such a Freelance system is Freelance Engineering (FE). This document describes the engineering tool Freelance Engineering (FE).

DWORD). The data type for analog modules is permanently set to UINT. In addition, the following data types are for field devices: INT, DINT, UDINT and REAL.



Freelance system.png

1.1.2 Engineering tool - Freelance Engineering

Freelance Engineering (FE) is a software application for **configuration**, **commissioning**, **documentation** and **maintenance** of the user programs and displays in the Freelance system.

Configuration using Freelance Engineering is based on the IEC 61131-3 standard. You can use Freelance Engineering to configure the entire process control level as well as create programs. This is facilitated by the fact that all components access a common database.

The system configuration is saved as a project file and is located in a preset project directory. To process a project, you must first open a project file. You then have unrestricted access to all project data within the project file.

The project name and file name of the project file can be selected independently from one another.

1.2 Starting the project

The following sections describe how to start a project in the Freelance Engineering application.

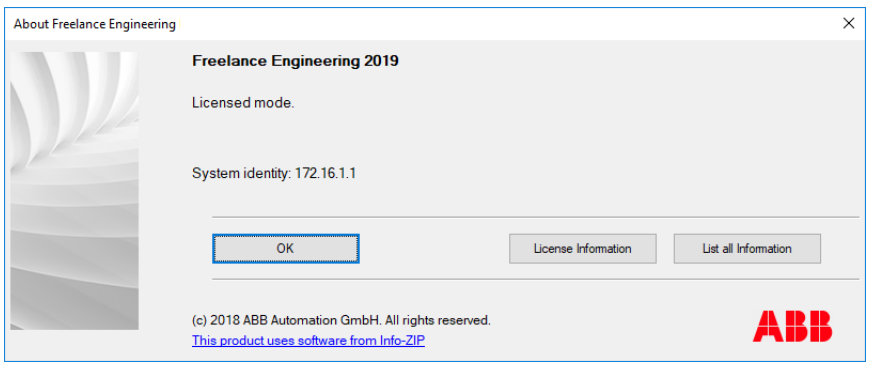
1.2.1 Starting Freelance Engineering

The Freelance Engineering is launched in Windows from the corresponding program group.

Windows 7:
Start > All Programs > ABB > Freelance 2019> Freelance Engineering

Windows 10:
Start > ABB > Freelance Engineering

Once the program is opened, the Freelance about box appears. This dialog box can also be opened at any time through the menu **Help > About**.



Aboutbox_us.png

A hard key and an authorization file is required to work with the software. These enable the functions of Freelance Engineering and/or Freelance Operations as per the order.

License Information
The currently installed licenses are displayed in list form.

List all Information
The installation on the Freelance Engineering PC is checked and the result is displayed in list form. For more information, refer to ***Freelance Getting Started Manual***.

Demo mode

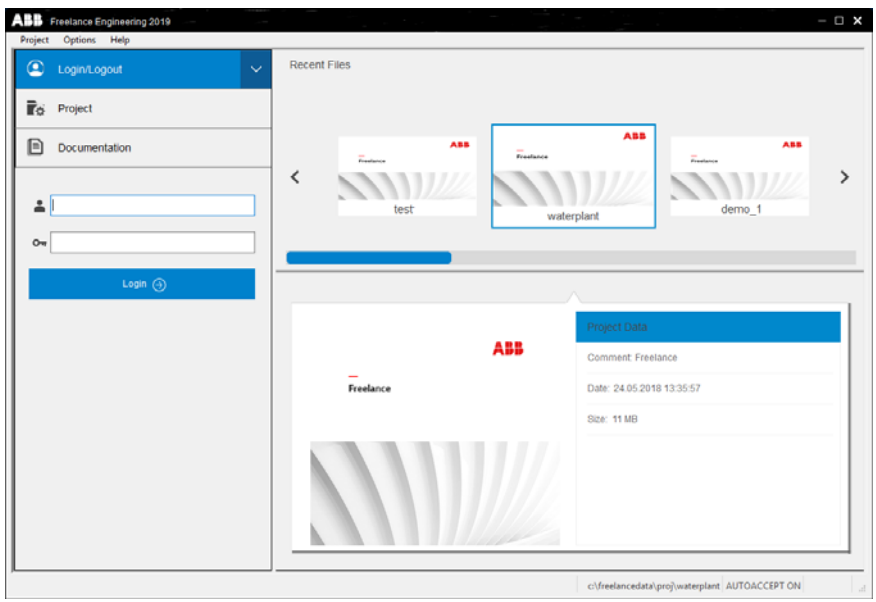
Freelance Engineering passes automatically to demo mode if no hard key has been found. Demo mode runs for 100 days and may be used for evaluation and

presentation purposes only. During this time, all kinds of operations such as save, export, and so on are allowed. At the end of this period the software stops functioning, and if required it can be reinstalled.

Without hard key the usage of this software is not allowed for professional purposes.

1.2.2 Welcome screen

Once you click **OK** button in the About Freelance Engineering Demo dialog box, the Welcome screen appears. The Welcome screen provides all functionality required for creating new project or accessing existing project.



Startpage_us.png

The Welcome screen contains the following groups:

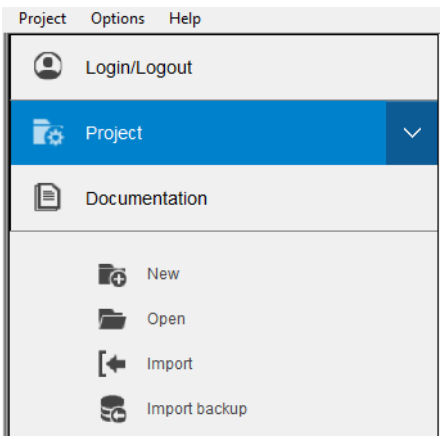
- User login
- Project
- Documentation

- Recent Files
- Information

User login

The User login group is used to access the Freelance Engineering for configuration and commissioning. It is only enabled if the User Management is enabled. User Management is used for user access management, it includes Security Lock and Extended User Management, user can select the access control method according to the actual situation. For more information, refer to the Engineering Manual User Management.

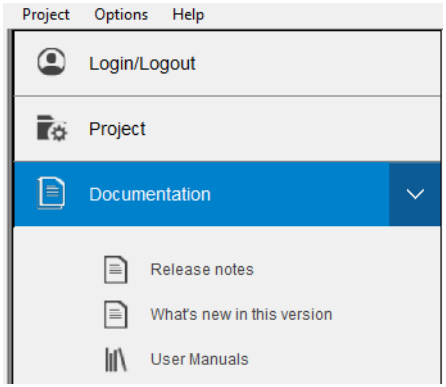
Project



Startpage_project_us.png

The project group is used to create new project, open existing project, import and import backup projects. For more information, refer to Menu structure of the Project Manager on page 37.

Documentation



Startpage_docu_us.png

The Documentation group contains the links to the following documents.

Document	Description
Release Notes	Release Notes of the installed Freelance version
What's new in this Version	Document that describe new features of Freelance version
Documentation	A link to configure the User documentation or other document.

Recent Files

The Recent Files group contains the last opened projects. The Recent Files list displays the file name of the project. The whole path of the corresponding file is available in the tooltip. A recent file can be pinned to the Recent Files list. Also it directly indicates if the project is password protected.

The Recent Files list is machine specific and will be available to all the users.

Pinning/Unpinning a project in the Recent files

To pin/unpin a project in the Recent Files, proceed as follows:

Point on the Recent Files project, Pin icon appears. Click Pin button

Or

Right-click on the project, select **Pin to list**

Nine out of ten projects in the recent project files list can be pinned. If nine projects are pinned, the Pin option for the tenth project will be unavailable.

Removing project in Recent Files

To remove a project from the Recent Files, proceed as follows.

Right-click on the project, select **Remove from list**

The selected project will be removed from the Recent Files.

Information

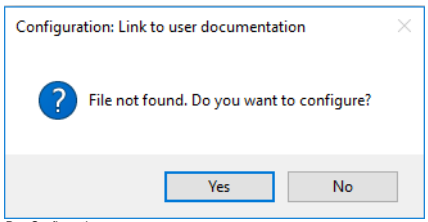
The Information group provides information to a project, which is listed in the Recent Files list. The information is displayed when you move the mouse over the listed project. The following information are displayed.

Project data	Description
Comment	Project comment as edited in the project header.
Date	Date of the project creation
Project size	Size of the project file in kilobytes (KB)

Next to the project information a preview image can be associated. This can be configured in Project Manager. For more information, refer to Associate preview on page 37.

Configuring link to Documentation

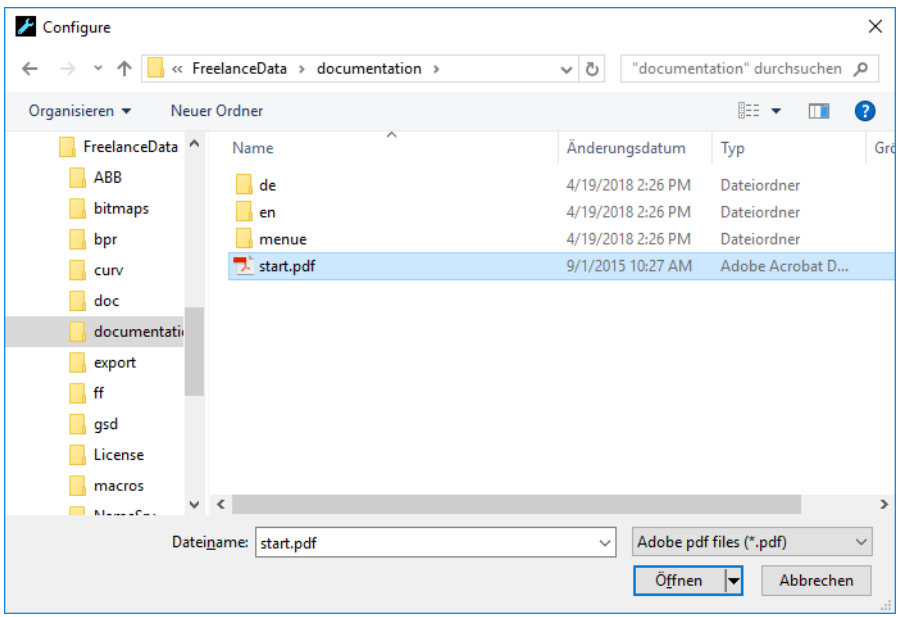
The documentation is not copied to the drive during installation. If you press the **Documentation** button for the first time, the *Configuration: Link to user documentation* dialog appears with the following message.



Doc Configuration.png

Yes	Click **Yes**, if you want to configure a link to the documentation.
No	Click **No**, if you do not want to configure a link to the documentation.

If you select Yes, a Configure dialog appears.



Configure dialog box_us.png

Select the path of the document file and click **Open** to update the path of a configured document.

Before configuring the link, it is recommended to copy the user documentation to the Freelance installation folder as shown in the above dialog.

To open the .pdf documents, the Adobe Acrobat Reader is required.

The configuration of the documentation link is machine specific and will be available to all the users.

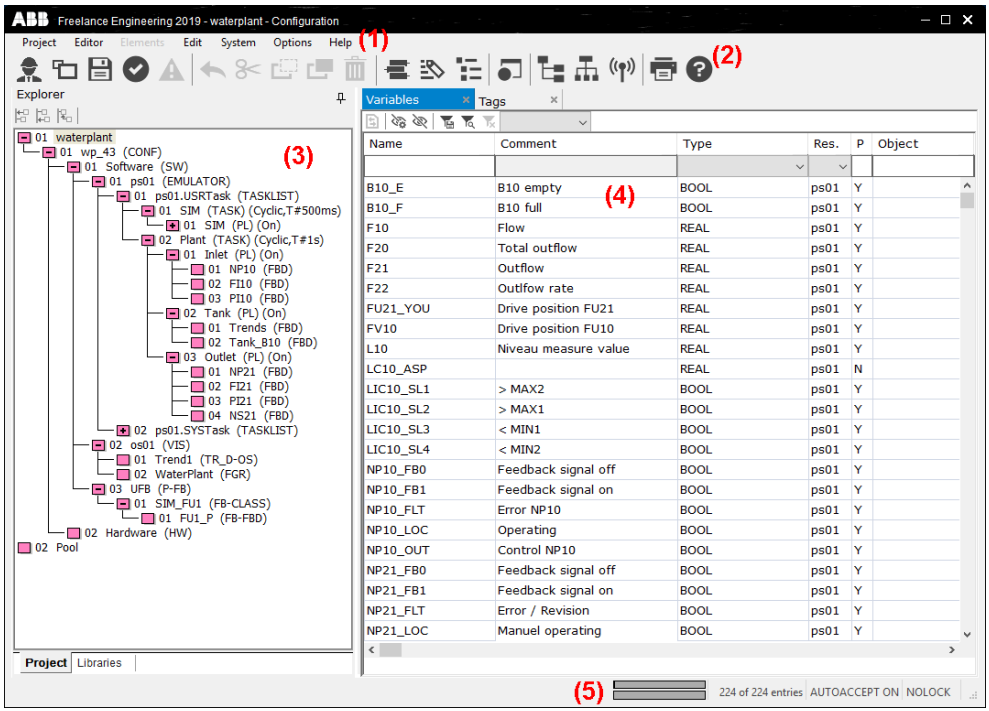
If already linked user document is moved to other location or deleted, the link must be reconfigured to the user document as described above.

Menu structure of the Welcome screen

Project	New...
	Open...
	Import...
	Import backup...
	Exit
Options	Login...
	Logout
	Change password...
	Run Security Lock
Help	Contents
	Overview
	About...

1.2.3 Configuration interface

Once a project is opened, it appears in configuration mode as shown in the following figure.



Project tree interface_us.png

The user interface is divided into:

- Menu bar (1): Contains menus which are used for editing the project.
- Common toolbar (2): Contains buttons which are used for editing the project.
- Explorer pane (3): Contains two tabs. The project tab shows the project tree, which provides an overview of the project objects in a project. The Library tab provides an overview of the available libraries by editing a user program.
- Workspace pane (4): Shows opened project objects from the project tree like user programs or graphic displays. Next to the project objects also general lists (variable list or tag list) are represented. Each opened object or list is shown in an own tab inside the Workspace pane. Switching between the tabs can be performed by selecting another tab.
- Status bar (5): Always visible and provides general information to user.

Status bar

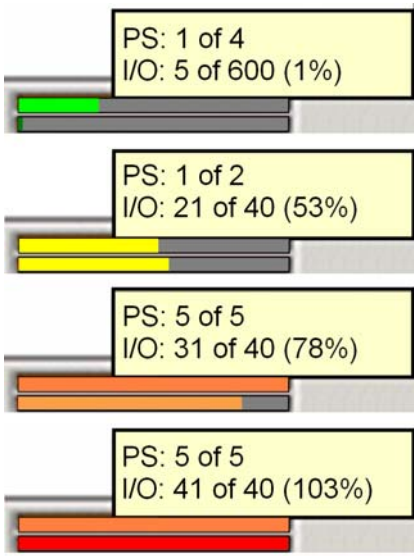
Status bar.png

Depending on the user interface, the **Status bar** shows the following information:

- Operator hints
- Path to the selected object in short or long form; in the hardware structure, the selected object is displayed without path information.
- User or responsible operator logged into Freelance Engineering. If User Management is not enabled, the display `NOLOCK` appears.
- Software license version display
- AUTOACCEPT ON/OFF. If Auto accept option is enabled, the display `AUTOACCEPT ON` appears.

Software license version display

All process stations and I/O components used in a project are counted and displayed in the Status bar using two horizontal bargraphs: The upper bargraph shows the ratio between the number of licensed I/O points and the number of process stations configured in the project. The lower bargraph shows the ratio between the number of licensed I/O points and the number of I/O points configured in the project. In addition to the bargraph size, the resource consumption is indicated by means of colors. The color changes from dark green (0%) to yellow (50%) and to orange (100%). Any percentages above 100% are indicated in red. The exact figures are shown by means of tooltips.



IOCount01.bmp

Explorer pane

Explorer pane contains the following two tabs which can be changed at the bottom:

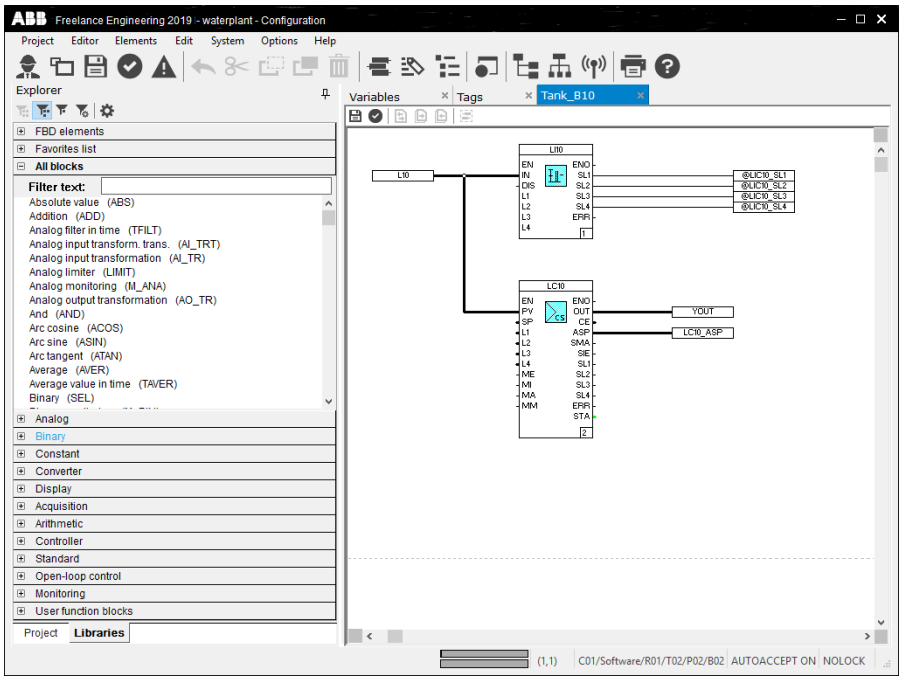
- Project
- Libraries

Project

The project tab shows the project tree to view the entire project configuration in a tree structure. Nodes from the project tree can be opened in an own tab in Workspace pane. For more information, refer to Section 3, Project tree.

Libraries

The Libraries tab provides all the available libraries (Function blocks, User function blocks, and Editor specific elements) of different editors as shortcut bars. The Libraries tab is used to select elements and drop to the editors.

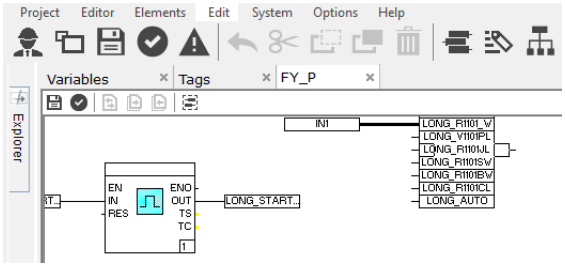


Libraries.png

Based on the active editor on the Workspace pane, corresponding libraries of that particular editor appears on the Libraries tab. For more information, refer to ***Engineering Manual IEC 61131-3 Programming***.

Hide/Unhide Explorer pane

An **Hide/Unhide** button is provided on the right top corner of the Explorer pane. Hide/Unhide function will hide the Explorer pane to provide maximum workspace area. To hide the Explorer pane, click **Hide/Unhide** button (unpin). The Explorer pane is hidden and available as Explorer tab on the left frame as shown in the figure.



Hide Explorer.png

When the mouse is pointed on the Explorer tree tab on the left frame, the Explorer pane appears. To unhide the Explorer pane again click **Hide/Autohide** button (pin).

Workspace pane

Workspace pane is used to view and edit project nodes and lists. Each of them are opened in an editor in an own tab. The following editors are available:

- Variable list
- Tag list
- Editors (FBD, IL, ST, LD and SFC)
- Graphics/Faceplate/Macro editors
- Structured data types
- Documentation editor
- User function block class

Newly opened tab will be in active and edit mode. The other opened tabs which are inactive will be visible on top of the Workspace pane.

Only one tab can be activated at a time. By activating a new tab, the previous one will go to inactive mode which requires a save for the changes. By default this is performed automatically as long as the Auto accept option is enabled. For more information, refer to Auto Accept on page 143.

By selecting such a tab it will get activated.

Variable and Tag list tabs are always opened at left most of the tabs. All other tabs are shifted. The recently opened tab will be added at the left most of the tabs after the Variables and Tags tabs (if opened) and will be in active mode.

When many tabs are opened in the Workspace pane, the tabs will move out of visible screen. Use the horizontal scroll option to scroll in between the opened tabs and select the required tab.

Context menu for tabs in the workspace pane

The context menu for tabs in the Workspace pane contains the following options:

- Locate in project: Find the corresponding node location in the project tree.
- Opened tabs: Switch between the opened tabs.
- Close tab: Closes the tab and corresponding editor.
- Close all but this: Closes all tabs except the one selected.

Closing Editor tab

The Editor tabs can be closed individually through:

From Editor menu, click **Close**

Or

Click **X** on the editor tab

Menu structure of the Configuration interface

The menu bar of the Configuration interface is always visible. Depending on the selected Editor type or the selected pane, the menus will change and provides only the corresponding and required functions.

The first menu is **Project** which is a common entry and is always visible. The function of the **Project** menu is same for all Editor types and panes. This menu provides all basic functions to manage the project. For more information, refer to the following table for project tree menu structure and ***Engineering Manual IEC 61131-3 Programming***,

All descriptions below refer to the **configuration mode**.

Project	Save project
	Save tab
	Documentation
	Check
	Check all
	Show error list
	Header...
	Comment
	Network...
	Commissioning
	Project Manager
	Close project
	Exit Engineering Tool
Edit	Undo
	Program
	Insert above, Insert below, Insert next level
	Expand, Full expand, Compress, Cut, Copy, Paste, Delete
	Export block..., Import block..., Import block as redundant...
	Search...
	Access rights (only on Security Lock), User groups (only on Security Lock)
	Display target stations
System	Variable list
	Tag list
	Structured data types
	Global message processing
	Local message processing
	Hardware structure
	Display access
	Communication configuration...
	Area definition...
	Show global variables...
	Show exported variables...
	Show all objects...
	Show selected objects...
	Breakpoint list
	OPC item List

Options	Print
	Long state line
	Color setting...
	Lock/Unlock UFB-Class
	Lock/Unlock Tag Type
	No initial variable/tag-filter
	Auto accept
Help	Contents
	Overview
	About...

Toolbar buttons of the Configuration interface

Common toolbar


Change to configuration mode


Change to commissioning mode


Edit the general information (header) of the selected object


Save current project or project part being edited or modified


Check selected object with all accompanying sub objects


List all error messages of selected object and the accompanying sub objects


Undo your last action


Cut the selected object and the accompanying sub objects


Copy the selected object and the accompanying sub objects


Paste the object from the clipboard which is captured through cut or copy actions


Delete the selected object and the accompanying sub objects


Open variable list


Open tag list


Open hardware structure


Open structured data types


Show list of break points


Open network configuration editor


Open communication configuration editor


Print the project content


Open online help for a selected object

Project tree toolbar


Insert a new object above the selected object


Insert a new object below the selected object


Insert a new object into the new hierarchial level

Libraries explorer toolbar


All block libraries


General block libraries


Communication block libraries

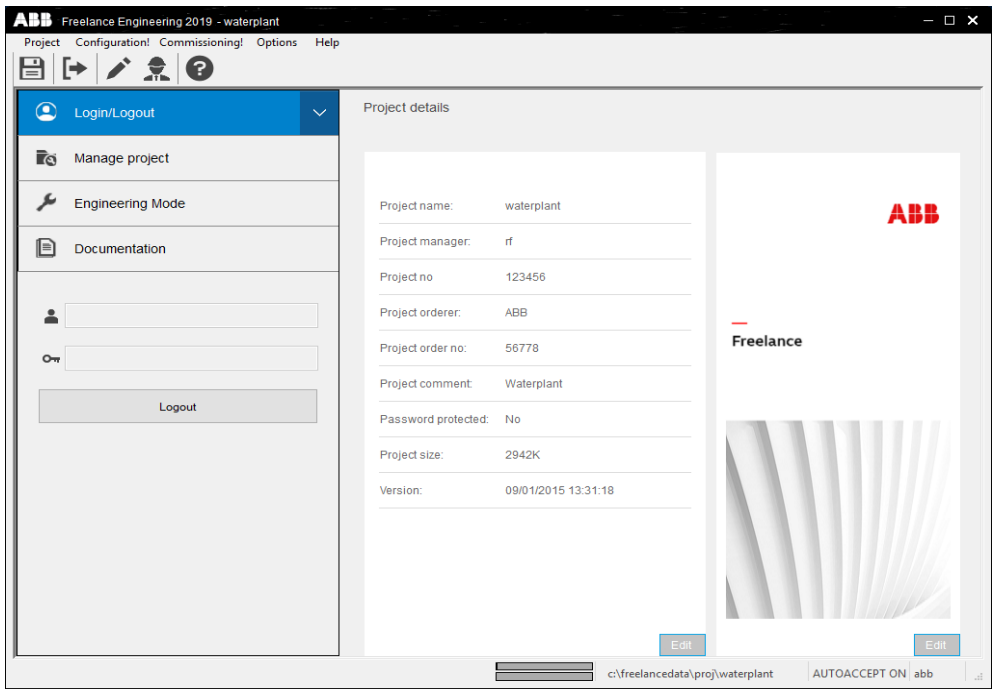

Own library list


Specify own library list

1.2.4 Project Manager

The Project Manager is used for general settings or modifications on an existing project. The settings or modifications like renaming, deleting or exporting of the project can be performed through the Project Manager. The Project Manager can be started only from opened project through.

> **Project > Project Manager**



Project manager_us.png

The Project Manager contains the following groups:

- User login
- Manage project
- Engineering Mode
- Documentation

- Project details

User login

The User login is used to access the Freelance Engineering for configuration and commissioning. A user can log out from the opened project and another user can log in to the project. For more information, refer to Run Security Lock on page 61.

This functionality is only enabled if the User Management is enabled.

Engineering Mode

The Engineering Mode provides access to the configuration interface in Configuration or Commissioning mode. For more information to the different modes, refer to Menu structure of the Project Manager on page 37.

Manage project

The Manage project group contains basic functionality for saving, exporting, closing or deleting the project. For more information, refer to Menu structure of the Project Manager on page 37.

Documentation

The Documentation group contains the links to the user documents. For more information, refer to Documentation on page 22.

Project details

The project details group describes the header details of the opened project. Next to these information it is also possible to edit the details.

Edit project details

This is used to edit the project header details. For more information, refer to Editing the project header on page 59.

Associate preview

For each project an own preview image can be selected. This preview image is also visible on the Welcome screen. For more information, refer to Information on page 23.

The preview image file will be stored in the **Images** folder below **<FreelanceData>\proj**.

If no image is associated with the project, then the default Freelance preview image will appear.

To associate a preview image to the project, click **Associate Preview** and select a new file.

Select the image file to associate to the project and click **Open**.

The maximum size of the image file is 5 MB. The image file format must be from the type bitmap and the preferred image aspect ratio is 16:10.

Menu structure of the Project Manager

Project	Save project
	Save as...
	Export...
	Export backup...
	Delete...
	Header...
	Comment
	Close project
	Exit Engineering Tool
Configuration!	
Commissioning!	
Options	Login...
	Logout
	Change password...
	Run security lock
	Tag names...
	PLC-Open export...

	Enable Freelance Operations write access on version error
	High resolution
	Start diagnostic server*
	Stop diagnostic server*
Help	Contents
	Overview
	About...

*- To be used only by authorized L3 support personnel. Do not use these options unless asked to do so.

Toolbar buttons of the Project Manager


Change to configuration mode


Change to commissioning mode


Save all changes in the current project


Export of the current project


Open online help for a selected object

1.2.5 Operation

Freelance Engineering provides all the functions required for creating, editing and commissioning a project. The menu allows the operator to select the various project processing options.

All operator actions can be performed by means of mouse or keyboard. For mouse operation the most important and frequently required functions are also made available through toolbars and shortcut menus. The buttons on the toolbar as well as the contents of the shortcut menus are dependent on the current state of project processing.

The toolbar buttons are used to initiate general, that is, object-independent functions, while the functions on the shortcut menu relate to the object currently being processed. The shortcut menu is executed with a right-click of mouse button.

Multiple ways of operations are possible in the configuration of Freelance Engineering. For a better overview, this document describes only the way through the Menu bar.

2 Project management

2.1 Editing projects

Once Freelance Engineering is opened, it starts with the Welcome screen. From this view new projects can be created or existing projects can be opened. For more information, refer to Welcome screen on page 20.

After a project is opened, the Freelance Engineering can be run in two different operating modes. The first mode is **Configuration** in which the project can be structured, configured and documented.

Configuration can be processed offline. In offline mode, no control system needs to be connected, to configure all the project objects. The hardware can be ordered later or can be already installed while the user program is still being configured. The project objects are assigned to the hardware structure as part of system configuration, and can then be downloaded to the stations when the connection is later online.

The second mode is **Commissioning**. When **Commissioning** is selected, a connection is automatically set up to the process and operator stations through the Control Net. For more information about setting the automatic connection, refer to Section 3, Project tree and Section 4, Hardware structure, and ***Engineering Reference Manual Communication and Fieldbusses***.

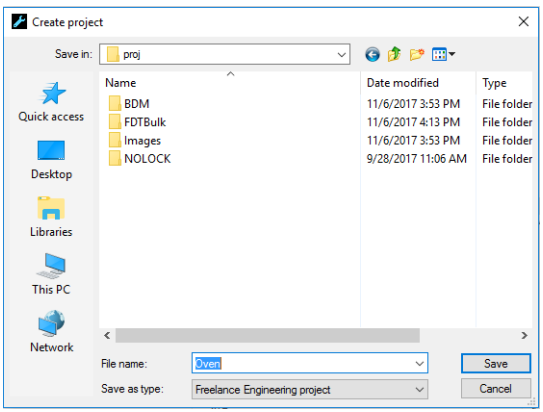
The Project Manager is available for general settings on an existing project. This Project Manager is opened only from an opened project. For more information, refer to Project Manager on page 35.

2.1.1 Creating a new project

From Welcome screen, select **Project > New...**

A Create project dialog box appears with the previously selected directory. A file name (maximum 256 characters) should be specified in the Create project dialog box. The system adds the extension **.pro** to the file name.

The default directory for the projects on the hard disk is **<FreelanceData>\proj**. The previously selected directory is opened.



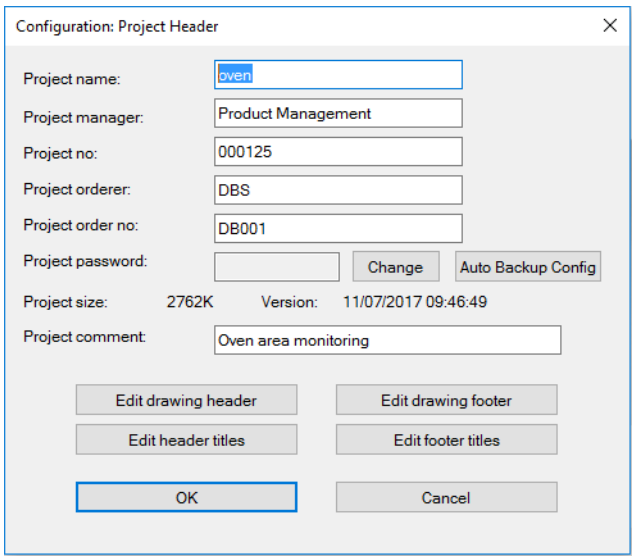
Create Project.png

Click **Save**, the project file will be created

A project header dialog box appears to edit general information of the project.

2.1.2 General information of the project

Edit the project details to be created in the header of the project.



Configure Project Header.png

Type the following information which will apply to the entire project:

Project name Name of the project. The maximum characters allowed are 12, The project name must not be the same as the name of the project file (.pro)

Project Manager

Name of the Project Manager. The maximum characters allowed are 27.

Project no The project number. The maximum characters allowed are six.

Project orderer

Name of the vendor. The maximum characters allowed are 27.

Project order no

The project order number. The maximum characters allowed are 12.

Project password

Password to protect the project. The password must contain at least four and not more than eight characters. The password can be modified or deleted using the **Change** button. To reopen the password-protected project, type the correct password. If a

password is set for the project, all the data (including .zip file, .csv file, .dmf file, .hwm file, .prt file, .eam, .msr file, .st file, .lad file, .fbd file, .as file, and .awl file, etc.) will be encrypted.

Please do remember the password well if a password was set for the project, because there is no way to restore the password if user forgets it.

Auto Backup Config

Opens the configuration dialog for Automatic backup of project. The automatic backup configuration can be performed through Project Manager. For more information on automatic backup configuration, refer to Configuring automatic backup on page 45.

Project size The current size of the project file (*.pro) is displayed in kilo bytes.

Version The project creation date is displayed.

Project comment

The project related comments can be typed in this field. This field can have a maximum of 34 characters.

Edit drawing header/footer

Assign specific system variables (F2) to the boxes in the drawing header or footer. For more information on drawing header or footer, refer to Section 6, Documentation.

Edit header/footer titles

This option is used to change the titles in the drawing headers or footers of each window. For more information on editing header or footer titles, refer to Section 6, Documentation.

OK Saves all entries in the database. The dialog box closes.

Cancel Closes the dialog box without saving any entries.

Changing the project password

To change the project password, proceed as follows.

> **Edit Project Details** > click **Change**

Enter password dialog box appears as shown in the following figure.

Change Password.png

Type a valid password in the *New Password* field. A valid password should contain at least 4 characters and no more than 8 characters.

Confirm the password by typing the same password again in the *Please retype* field.

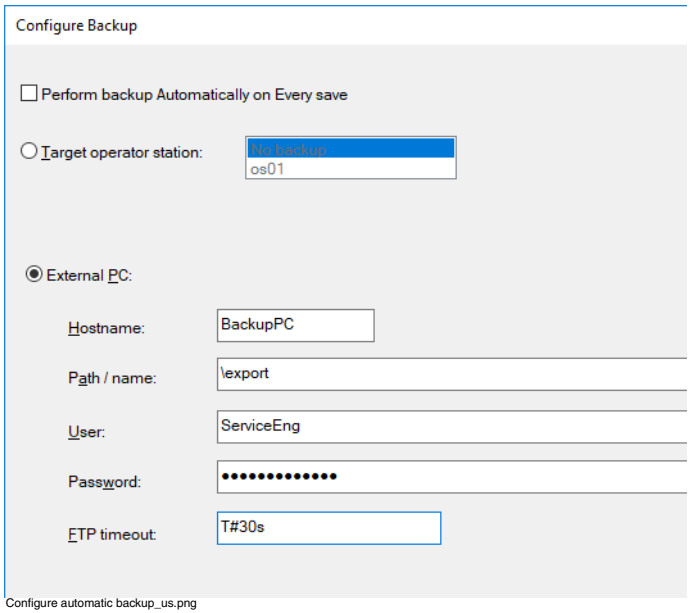
Click **OK**.

Please do remember the password well if a password was set for the project, because there is no way to restore the password if user forgets it.

Configuring automatic backup

To configure automatic backup, proceed as follows.

> Edit Project Details > Auto Backup Config



Configure automatic backup_us.png

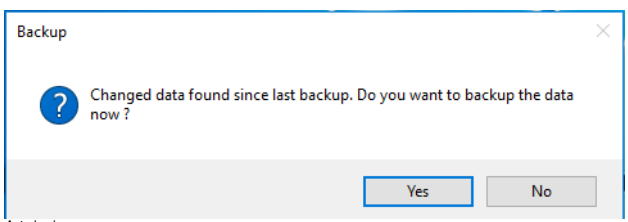
In the dialog box that opens, select one of the operator stations configured in the project or enter the data of an external PC where the backup of the project file should be located using FTP. This requires an FTP installation on the PCs.

If no operator station is configured in the project, add an Operator station resource in the project tree.

Perform backup automatically on every save

☑ Backup is performed automatically as per Configured backup on every save

☐ The system shows the following Message box on every save.



Auto-backup-message_us.png

Yes	Backup of the project will be saved
No	No Backup will be saved.

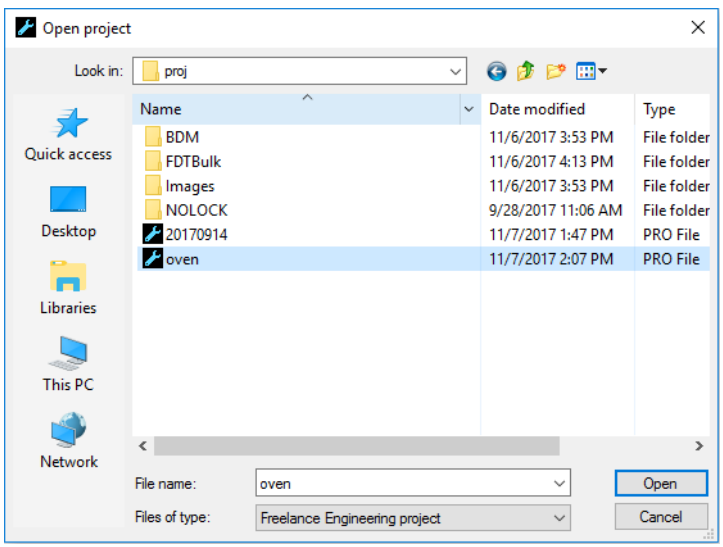
Configure Backup > **OK**

If the project is password protected, the saved Auto backup file in the destination folder is encrypted, and user has to input the correct password to import backup files.

2.1.3 Opening a project

From Welcome screen, select **Project > Open...**

The open project dialog box appears.

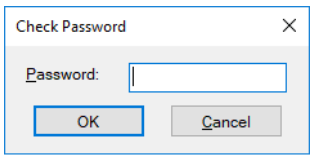


Open project window_us.png

Select a project file from the Open project dialog box > click **Open**

As for cyber security and for some projects require higher data security, user can set a password for the project. Consequently, user needs to input a valid password to

access the password-protected project. The file type for the password-protected project appears as *.PRS after the project is encrypted. Without special notes, the project file in the manual refers to the *.PRO file.



CheckPassword_us.png

After the project be encrypted, all the related data will be encrypted as well, and some file names change afterwards. E.g. *.CSV file changes to *.CSVS file, *.DMF file changes to *.DMFS file, *.HWM file changes to *.HWMS file, and *.PRT file changes to *.PRTS file. The backup file .ZIP remains the same after encryption.

In some exceptional situations, project name *.pro and project name *.prs might exist at the same time. The system will prompt a message asking if user would like to accept a recovery. If Yes, the *.prs file will be deleted and the *.pro file can be accessed; if No instead, neither of the file can be opened until user deletes one of them.

Access to the directory is defaulted during installation. The previously selected directory is opened (the default directory for projects on the hard disk is **<FreelanceData>\proj**.

When a project is opened, the project file (**.PRO**), a file with the extension **.LOG**, and a file with the extension **.BAK** are created.

The file with the extension **.LOG** is a temporary file, in which the steps of a project work session are logged. Its purpose is to prevent the loss of changes made to the project during a session of Windows crash. If this happens, the file can be restored in the database after a restart of Freelance Engineering and reproduce the state at the time of the crash.

The temporary project file with the extension **.LOG** increases incrementally with every change in the project and is only reset to zero kilobytes when the project is saved.

The file with the extension **.BAK** is a compressed backup file of the saved project file. When the Freelance Engineering is closed and select **No** from the **Save** menu

item, the file is expanded and the extension **.BAK** is replaced by the extension **.PRO**.

When the project is opened the block versions in the project are compared with the versions of installed blocks. If any discrepancies are found, then a Function block update dialog is opened. This dialog contains a list of all the block types with different version numbers.

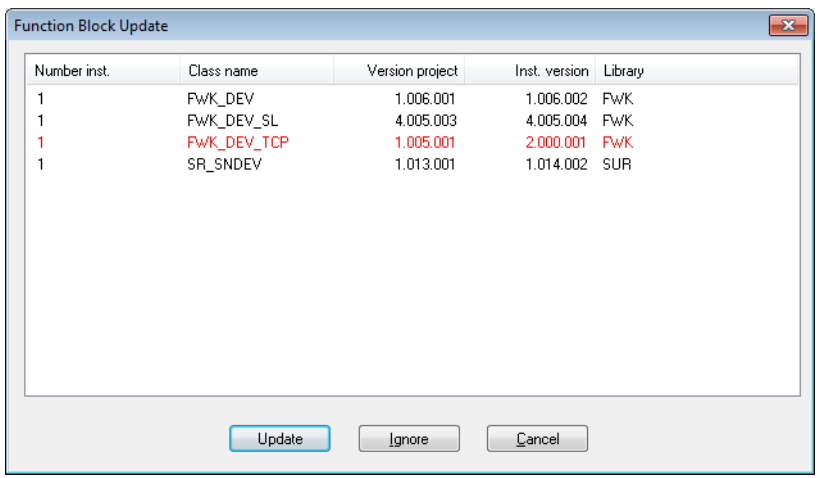
A block type version number is composed of three levels. The individual version numbers are separated with a period.

Major_version.minor_version.code_version

Major version The major version number identifies the external form of a block (for example, its representation in the FBD program).

Minor version The minor version number identifies the parameters defined for a block (for example, the layout of the parameters dialog).

Code version The code version number identifies the stage reached in the internal processing of a block.



tc001_us.png

The following parameters are shown for each block type:

Number inst. Number of instances. Shows how many times the block type has been used in the project.

Class name Name of the block type.

Version project Version number of block type in the project.

Inst. version Installed version number of the block type (FE).

Library Name of library to which the block type belongs.

The differences between versions can be updated or rejected.

Update All the selected programs are set to implausible. Differences in **major version** cannot be updated. Any blocks affected remain incompatible.

In the case of differing **minor versions** an update function is executed, which updates the selected blocks.

Block types with differing **code versions** are flagged for reloading.

After the plausibility check and loading of changed objects it is possible to continue working with the project.

Ignore All the selected programs are set to implausible. All the selected blocks remain unchanged in the project.

If you select **Ignore** you should not load any object in the process station.

Cancel The project is left unchanged and is closed without being saved.

Opening a project from recent files

The project can also be opened from the recent files. To open the project from the recent files, proceed as follows:

Click the project

Or

Right-click on the project, select **Open project**

2.1.4 Hard disk memory requirements

During a session on an open project, there are three files open: the project file (.PRO), a temporary project file (.LOG) and a backup file (.BAK).

The project file size is dependent on the size of the project and quickly reaches several Mbytes of space requirements on your hard disk.

We recommend at least 5 times the file size of a plausible project for the temporary requirements of free hard disk space. A project which has been subjected to a plausibility check is about 3 times larger than the one which has not.

2.1.5 Exporting a project

From Project Manager, select **Project** > **Export...**

This function exports the entire content of the current project to a file. The file formats available are **Unicode export file (*.csv)**, **PLC-Open-file (*.plc)** or **extended PLC-Open-file (*.ple)**. A **csv file** can be reloaded using **Import**. A **PLC file** is used for transferring the project data to other systems, such as Maestro UX and System 800xA.

An Export project dialog appears. Provide a file name for the export file in the explorer that appears.

If the project is password-protected, the exported *.csv will be encrypted and the file type turns to *.csv.

Select the file name and click **OK**.

Click **Save** after selecting the **Save as type**.

The default directory for export files is the **<FreelanceData> \export**.

If possible, only plausible projects must be exported. Only then it is ensured that all data are correct and consistent.

It is important with a PLC and PLE export that the names of tags and variables that are to be transferred to another system must conform to the naming conventions of the target system.

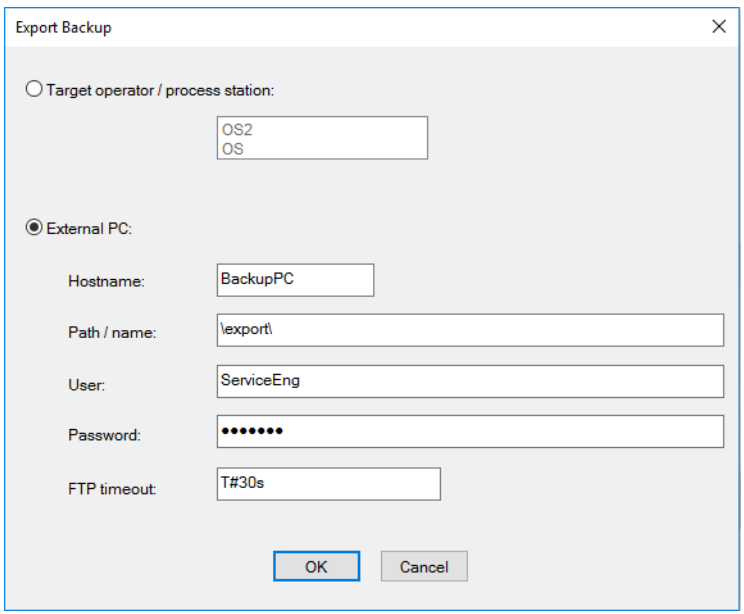
2.1.6 Exporting a backup

The project file can be exported to an operator station or to an SD card inserted in an AC 700F or AC 900F controller, or through FTP export to have a backup. The logged in user must have privileges to backup the project or no User Management is enabled. If required, maintenance personnel can access this backup.

To backup the project, proceed as follows:

From Project Manager, select **Project > Export backup...**

An Export backup dialog appears.



Export backup_us.png

Select one of the operator stations or process stations configured in the project or enter the IP address of an external PC where the backup of a project file must be located.

If no operator station is configured yet in the project, add a D-OS resource in the project tree and assign it to an IP address in the hardware structure.

Exporting a backup to process station is actually exporting the backup to the SD card in the controller. An existing backup file will be overwritten by the new backup file.

Regarding redundant case, only the primary controller will store the file on its SD card. If users want to have the same copy on both controllers, they need to toggle the controllers in between because file download is not synchronized between the redundant controllers currently.

Type valid data for the following fields:

Hostname	Type the host IP address.
Path / name	Provide the path to export the backup of project.
User	Type valid user name
Password	Type valid password
OK	Saves all entries in the database. The dialog will be closed.
Cancel	Closes the dialog without saving any entries.

Click **OK**.

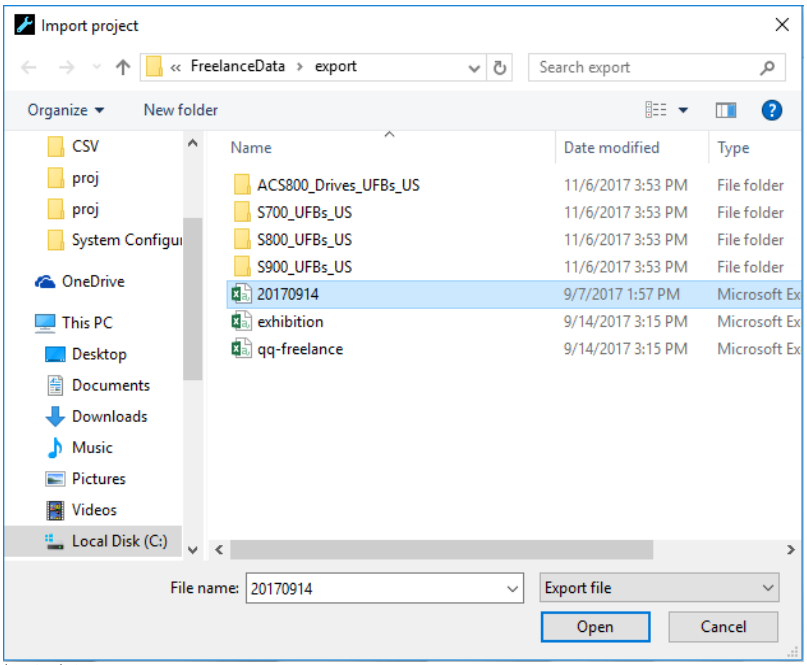
An online connection is automatically set up to transfer the project over the Control Net to the operator station. The project file is then saved in compressed form on the operator station in the **<FreelanceData>\proj** under the file name **Backup.zip**, or SD card inside of the controller if exporting to process station is selected, or on the FTP Server (Microsoft Peer Web Services), corresponding to the home directory and path.

2.1.7 Importing a project

To import a project, proceed as follows:

From Welcome screen, select **Project > Import...**

An Import project dialog appears.



Import project_us.png

Select an Export file from the **Import project** dialog > click **Open**

This function imports the content of a project from a **csv file** previously generated with **Export**. The new project name is first displayed in the dialog. After the import process is complete, save the project under this project name. Confirm the file name of your choice, another dialog opens. Select the required csv file from a directory of your choice.

When the **csv** file of a project previously generated with **Export** is imported, the preview image associated with the project is lost and shows the default Freelance preview image. For more information on associating the preview image to the project, refer to Project details on page 36.

The default directory for import files on the hard disk is **<FreelanceData>\export**.

Any offline changes in the csv file are not recognized by the system and could possibly destroy the project database. If a manipulated project file is loaded into the process station, this leads to a system crash.

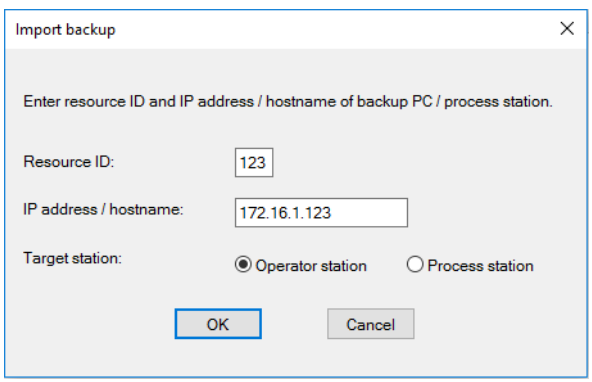
The export and subsequent import of **csv files** is only guaranteed within one main version and between any existing sub versions or between two sequential main versions. if there is a jump of more than one main version, the project has to be exported and imported over the intermediate versions.

2.1.8 Importing a backup

If no updated project file on the engineering station, the option of importing a backup of the project file can be used. Project can be imported from a backup system. The backup system can be remote or local system.

From Welcome screen, select **Project > Import backup...**

One condition for an import is that a backup was already exported to an Operator station / Process station or that a backup file with the name **Backup.zip** exported through FTP-Export is located in the **<FreelanceData>\export**. No projects are allowed to be open. Any open projects must be closed before the import commences.



Import a Backup_us.png

In the Import backup dialog, type the values for Resource ID and Hostname (IP address) fields of the Operator station / Process station where the project file backup is located.

Click **OK**

An on-line connection is automatically established and the project is transferred from the corresponding station through the Control Net. The project file is saved in compressed form under the file name **Backup.zip** in the **<FreelanceData>\proj**, or in the SD card used for the controller.

The IP address is part of the TCP/IP installation and can be changed under **Windows**, (**Control Panel** > **Network and Internet** > **Network and Sharing Center** > **Change adapter settings** > Click **Network Connection**).

The resource ID is assigned during the Freelance setup and may be changed through the program group **ABB > Freelance <version>** > **Settings**.

For more information, refer to Section 4, Hardware structure.

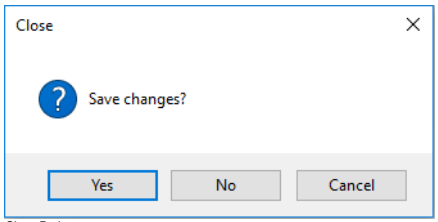
2.1.9 Closing the project

A opened project can be closed from the Configuration interface or the Project Manager:

> **Project** > **Close Project**

If you make any changes to the project, you will be asked to save these changes. If a backup is not configured, then the system displays a message asking the user if a backup should be created. For more information on configuring backup, refer to Exporting a backup on page 52.

A Save dialog appears.



Close Project.png

Yes Saves and closes an opened project. The project file (.pro) is updated, and the temporary file (.log) and the backup file (.bak) are closed. The project is closed and the Welcome screen appears.

No All changes made during the entire session are rejected. The backup file (.bak) is expanded and overwrites the project file (.pro).

Cancel Returns to the Project Manager.

2.1.10 Saving a project

A project can be saved from the Configuration interface or the Project Manager:

> **Project** > **Save Project**

When **Save Project** command is selected without giving any further instructions, all the changes made after opening the project with **Open** are saved. This includes the plausibility checks on project objects, loading the objects to a station or changing a configuration. The default directory for projects is the **<FreelanceData>\proj**. The project remains open and can be edited.

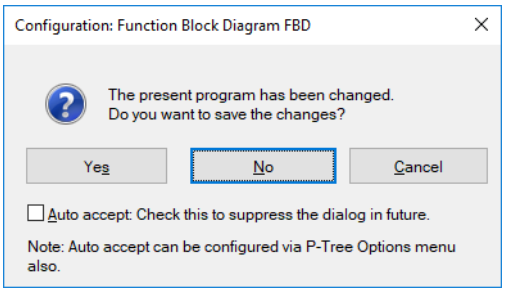
The save function resets the temporary file (.LOG) and deallocates storage.

2.1.11 Saving a tab

From Configuration interface, select **Project > Save Tab**

Switching between tabs

If modification in a tab are performed, the changes has to be saved. by leaving the active tab the following dialog appears.



Saving tab data_us.png

Yes The changes will be saved.

No The changes will not be saved.

Cancel The performed action is cancelled.

Auto Accept ☑ If this option is selected, the save action will be performed automatically and the Save dialog will not appear.

☐ If this option is not selected, the Save dialog will appear by leaving a tab in which modifications were performed.

For more information on configuring the Auto Accept option, refer to Auto Accept on page 143.

2.1.12 Save project as

If you want to change the project name, use the **Save As** command.

It is possible to change the file name of a project. The project name continues to exist and can be changed in the project header.

From Project Manager, select **Project > Save As...**

A project save as dialog appears. The default directory for projects is the **<FreelanceData>\proj**.

2.1.13 Deleting a project

To delete the project, proceed as follows:

From Project Manager, select **Project > Delete...**

When **OK** is selected, all the files associated with the project are deleted. The only exceptions are the files created using the **Export** or **Export block** commands.

2.1.14 Editing the project header

The project header contains general information on the project which can be printed through the project documentation. To edit the project details, proceed as follows:

From the Project Manager, select **Project > Header...**

For more information on editing the project header, refer to General information of the project on page 42.

2.1.15 Editing a project comment

From Project Manager, select **Project > Comment**

Use the Configure comment dialog box to create or modify a free text for the project, to import this text or to print this text. Additionally, it is possible to import and export Unicode TXT files.

To Export or Import comments for the project.

> Configure comment > Edit > Export/Import

2.1.16 Exiting Freelance Engineering

From Configuration interface or Project Manager, select **Project** > Exit Engineering Tool

A Close dialog box appears to confirm the save changes.

When you confirm by selecting **"Yes"** command, the open project is saved and closed. The project file is saved and Freelance Engineering is terminated, and the system returns to the Windows user interface.

2.2 Project Manager options

Project > Project Manager > Options

From Project Manager using the menu item **Options** you can open the following commands:

- Login...
- Logout
- Change password...
- Run security lock
- Tag names...
- PLC-Open export...
- Enable Freelance Operations write access on version error
- High resolution
- Start diagnostic server
- Stop diagnostic server

2.2.1 Run Security Lock

Security Lock is part of the User Management and installed by default during the Freelance installation. It assigns the user rights and defines the user groups. Group specific rights can be assigned to the user groups for:

- Configuration
- Commissioning
- Operator actions

Every user is identified by a **password**, which can be changed. If Security Lock is enabled on an engineering station, the user must enter the password before starting to edit or modify a project.

With Security Lock enabled, you must log in before you can work using Freelance Engineering or Freelance Operations.

As long as no user has logged in, the rights assigned to the guest group are activated.

The standard users are:

NOLOCK	No license for Security Lock
Guest	No one has logged in, (for example, on starting Freelance Operations)
System	Operator action by system (may appear in log)

The **Login**, **Logout**, **Execute Security Lock** and **Change password** actions can be recorded in the logbook file under Freelance Engineering. The **Login**, **Logout**, and **Change password** menu options are executed only if Security Lock is enabled in the system.

From the Welcome screen or Project Manager, login with a valid User name and Password (with privileges to configuration access rights). Open a project by selecting **Open** button or from the Recent Files list. For more information on user access, refer to ***Engineering Manual User Management***.

As an alternative to Security Lock, Extended User Management can be activated. The Extended User Management enables user management via the local Windows user administration or via a Windows domain.

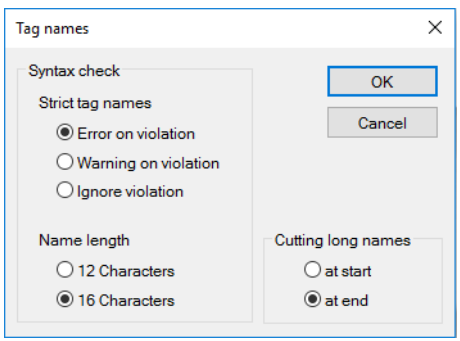
2.2.2 Checking tag names

From Project Manager, select **Options > Tag names...**

The Syntax check in a project does not permit a tag name to start with a number (according to IEC 61131-3). In certain process sectors (for example, in the power generation sector), this regulation must be bypassed. It is also possible to choose between 16 character and 12 character tag names. The appropriate selections should be made in the Project Manager through **Options > Tag names**.

Special characters (! § $ % & / ( ) = ? ß { } [ ] \ ' # + * ~ - _ . : , ; |) and umlauts can be used in tag names only if check for IEC 61131-3 conform names is deactivated through **Options > Tag names...**.

Checking tag names is based on the rules defined in IEC 61131-3 standard. These rules allow leading and embedded “ ” characters in tag names but not leading numbers. Sequences (more than one in a row) of or trailing “_” are not allowed.



Tag Names.png

Strict tag names

Error on violation

◉ All tag names must conform to the IEC naming conventions (defaults).

Warning on violation

◉ Tag names that do not conform to IEC conventions are allowed, but a warning is issued.

Ignore violation

◉ Tag names are not checked. The selected setting is stored both in the project database and the CSV export file. This ensures that the same tag name checking procedure is used each time a project is either opened or imported (for example, to a different Freelance Engineering PC).

Name length

12/16 Characters

The length of tag names is limited to either 12 or 16 characters. Without this adjustment the names have to be changed manually to get the project checked.

Cutting long names

at start/at end If there is not enough space for displaying the whole tag name, the name is cut in the display. The complete name can be shown by using the ToolTip.

2.2.3 Automatic execution of a command with PLC-Open export

From Project Manager, select **Options > PLC-Open export...**

A Windows command can be automatically executed immediately after the creation of a *.PLC file. Select menu item **> Options > PLC-Open export...** a dialog appears to enable a command and associated parameters to be entered.

Select **Execute command with PLC export**. The specified command will be executed directly after the creation of a PLC file (through Project menu **> Export > PLCOpen file**).

The example below will automatically create a backup of the PLC file:

Command: **xcopy**

Parameter: **c:\BACKUP\save.plc**

Every time a PLC file is created it will automatically be copied to directory **c:\BACKUP** with the name **save.plc**.

2.2.4 Freelance Operations write access on version error

If only the process stations are loaded upon project re-configuration, a version conflict occurs between the operator stations and the process stations. Disable the write access from the operator stations if not useful. In special cases and if the configuration changes made are known, then the write access is possible from the operator stations.

From Project Manager, select **Options > Enable Freelance Operations write access on version error**

Use this menu option in the commissioning phase. Otherwise, ignoring the version error may cause a system crash on the process station.

2.2.5 High resolution

This option serves as default of a high resolution when creating a new operator station in the project or configuring group displays. If the presentation of a standard group display is changed, in the configuration dialog for the group display, select whether the group display for a Freelance Operations station is to be created with high or low resolution. Depending on the selection, 6 x 1 or 5 x 2 standard faceplates can be put together in a group display.

If at this point a tick is set in the actual configuration menu the high resolution is already set by default. Otherwise select when configuring the group display.

From Project Manager, select **Options > High resolution**

2.2.6 Start and stop diagnostic server

These options are to be used only by L3 support personnel.

From Project Manager, select **Options > Start diagnostic server** or **Stop diagnostic server**

2.3 Miscellaneous

2.3.1 Freelance Engineering version

Using the menu option **Help > About**, the dialog described in Starting Freelance Engineering on page 19 can be opened.

2.3.2 Event log

The Event log is a log which can record operator or editor actions performed in Freelance Engineering during Commissioning. To perform Windows service, start the eventvwr.exe file in the directory: **<Windows-directory>\system32**. Select **Event Log (local) > Application**, Freelance events will be displayed. For more information double-click on an event.

The following events are recorded in a log. For example, loading project objects in resources with:

- Date and time (h:min:s) of loading operation
- Loading state
- Object number
- Object type of loaded project object
- Name of loaded project
- Name of loaded project objects (task, program list, program)
- Name of loaded tag names (=function blocks) in this program

Commissioning actions such as:

- D-PS resource start, Stop, Init/Reset and Boot.
- Task: Start, Stop, Reset and “Once” (single execution)

2.3.3 License level display

With the I/O counting tool all the I/O components used within a project can be counted and displayed. For an open project, all the I/Os that have already been used can be displayed in the hardware manager. This information is always displayed by means of two horizontal bar graphs in the Status bar of Freelance Engineering:

- The upper bar graph represents the ratio between the number of licensed process stations and the number of process stations configured in the project.
- The lower bar graph represents the ratio between the number of licensed I/O components and the number of I/O components configured in the project.

Apart from the size of the bar graph also its color has a meaning: it represents the resource consumption. The color changes from dark green (0%) through yellow (50%) to orange (100%). All percentages exceeding 100% are displayed in red. The exact figures are displayed in a tooltip.

2.3.4 Monitoring of project database

The Freelance Engineering database is constantly monitored. Once a problem is detected, immediately a dialog appears to close the project. Except for a few changes, the database can be restored.

To ensure a secure project work for very large Freelance projects, the database compression can be enabled. This should be carried out starting from a project file size (*.PRO) of 1.6 GB. If this is required for your project, please contact your local ABB Service.

2.3.5 Importing a Freelance project of earlier versions

When a freelance project is imported from an earlier freelance version, a message whether the group displays for Freelance Operations shall be imported with high resolution or not appears. The arrangement of the faceplates within the group display is adapted to the new grid by default adjustment. For example, if six blocks contained in a group display, to which faceplates in standard size exist, previously these six faceplates were displayed next to one another. With the option **“High resolution”** enabled, the sixth faceplate is displaced from the sixth position in the first line to the first position in the second line.

When this option is disabled the arrangement of the faceplates remains unchanged. Due to the new formats of the faceplates, only five tags are visible next to one another in Freelance Operations, the sixth tag becomes visible only by scrolling. If the operator station is operated with the classical resolution (vertical resolution less than or equal to 1024 pixels), answer the question with “No”. The width of groups with six faceplates remains unchanged.

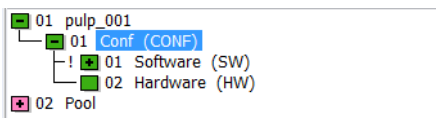
3 Project tree

3.1 General description - Project tree

The project tree provides an overview of the project objects in a project. The individual elements or objects are structured in accordance with IEC 61131-3.

The uppermost element in a project is the **Configuration CONF**, which is the sum of all project objects in a Freelance system.

The first structural level below configuration is Software and Hardware nodes as shown in the following figure.



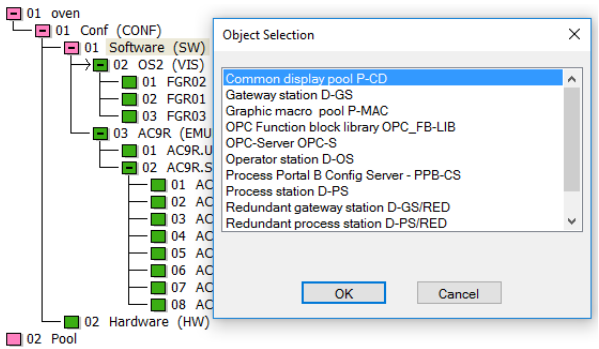
Software hardware node_us.png

The software node contains the resources, which represent the various different stations in a project. The hardware node provides an easy option to access the Hardware structure (also can be accessed through Menu or Toolbar). For more information on Hardware structure, refer to Section 4, Hardware structure on page 147.

For the processing of the process there are the **D-PS (Process station) resources**, for operation and observation of the process there are the **D-OS (Operator station) resources**, and **800XA-AS (System 800xA Aspect server)**, for interfacing to external systems there are **D-GS (Gateway station) resources**, for integrating data from external systems there is the **OPC server (OPC-S) resource**.

Data exchange between the process stations is performed by means of lateral communication. Each process station is equipped with 10 connections for data exchange with operator stations and gateway stations. If more stations are configured, then the effective communication links of all the subscribers in a Freelance system are specified in a separate parameter definition dialog.

There are also additional structural elements such as the **User function block pool P-FB**, **the Graphic macro pool P-MAC**, and the **Common display pool P-CD** as shown in the following figure:



Software node items.png

According to the different tasks of these resources, additional project objects are available on the next levels for configuration:

Process stations D-PS

Process stations are added with **resource types D-PS** and **D-PS/Red** (non-redundant and redundant stations respectively). Correlation to the physical stations is carried out in the hardware manager. Here, Process stations (PS) can be selected for rack systems, AC 900F controllers (AC 900F P, AC 900F, AC 900F L), AC 800F controllers, AC 700F controllers and FieldControllers (FC) can be selected for connecting field bus modules.

The execution of user programs within a process station is controlled by **tasks**. Within a task the sequence of user programs are defined either by **program lists** or by a structure - the sequence flow - that is configured using the **Sequential Function Chart (SFC)**. While the programs in a program list are run cyclically, SFC programs are run only for specified periods of time according to their structure.

The user programs within a process station, both under a program list and under a sequence flow, are created using the IEC 61131-3 programming languages: **Function Block Diagram Language (FBD)**, **Ladder Diagram (LD)**, **Instruction List (IL)** or **Structured Text (ST)**.

Operator stations D-OS

For operation and observation of the process, the **D-OS** resources can be selected for the Operator stations using the Freelance Operations software, while the **800XA-AS stations** can be selected for the Workplace Operator stations.

Standard operating facilities (for example, **faceplates**), are provided for all known tags and variables on the Operator stations. In addition, **graphic displays**, **trend displays**, **SFC displays** and **logs** can be configured and structured for the Freelance Operations Operator stations.

Gateway stations D-GS

The gateway stations are added with resource types **D-GS** and **D-GS/Red** (non-redundant and redundant stations respectively). The gateway stations are used to make data from the Freelance system available to other systems. In principle, all the data from the system can be read and written through a gateway station. In addition to each gateway station in a Freelance system, the appropriate server software from the add-on packages OPC server must also be installed in the network.

For example, a gateway station of OPC gateway type is configured in a Freelance system, then the OPC server software must be installed on a PC that is linked with the Freelance process stations on the network.

For each gateway station it is configured which tags and variables are available to the other system for reading and/or writing through the gateway.

OPC server stations OPC-S

Configuration of an **OPC server resource** in the project tree enables data to be integrated into a Freelance system.

According to the configuration of the gateway stations, parameters are defined for interfacing to an OPC server that determine which data from the external system is to be integrated into the Freelance project.

OPC function block library OPC_FB-LIB

The **OPC function block library OPC_FB-LIB** node in the project tree is located below the software (SOFTWARE) node. The Tag types are arranged in the **OPC function block library** node in the project tree. Import the standard library that is delivered with Freelance system.

User function block pool P-FB

New classes of block are defined below this project object; these can then be used in the configuration of the user programs like the function blocks contained in the firmware.

Common display pool P-CD

Displays and logs set up under this project object must be available on all Freelance Operations Operator stations. The objects from the display pool are loaded only on those stations that have access to the process data required in these displays and logs.

Graphic macro pool P-MAC

Below this element all graphic macros are defined that can be used in the graphic display.

Project pool

Alongside the Freelance project there is a project object **Pool**. This project pool can be used to temporarily store any project components as required. Any unchecked or unneeded configurations can be stored here and then completed or re-integrated into the project at a later point in time.

All the project objects are represented in a tree structure. A node is shown in front of each object. The color of the nodes is used to represent their processing state, and branches can also be recognized from the symbols. Sections of the project tree can be compressed as required, and this allows the overall structure to remain clear even in sizeable projects.

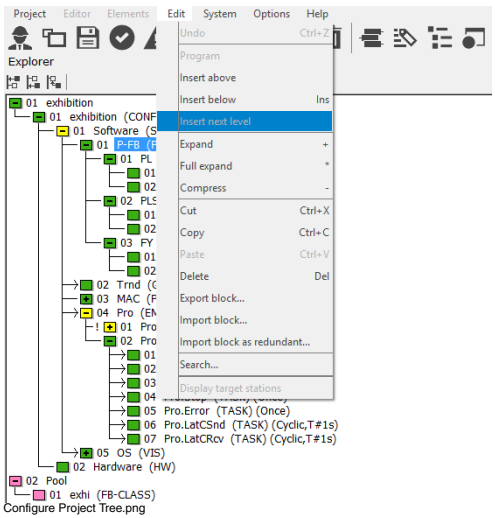
The Freelance Engineering program contains two project processing states, **Configuration** and **Commissioning**. During configuration there must be no link in existence to the stations in the project. When switching to commissioning, a network link is established to all configured stations. The configured and plausibility-checked user programs can be downloaded from the project tree to all connected process stations, Operator stations and Gateway stations.

The import and export functions that have been implemented can be used to exchange parts of the configuration with other projects. Checks are carried out when importing whole and part projects to ensure that all the labels within a project are unique.

User Management allows to set different **user rights** to various **user groups** to operate or configure the projects. User can select either Security Lock or Extended User Management to manage control access for both Freelance Engineering and Freelance Operations. For more information on User Management, refer to ***Freelance Engineering Manual User Management***.

3.2 Configuring the project tree

The project tree is used to configure the project objects in a hierarchial tree structure. The menus described here are only valid for the project tree. The project structure is generated by selectively inserting the available project objects.



Configure Project Tree.png

3.2.1 Project objects

The junction beside the node shows the processing sequence of the project objects on the corresponding level. In the case of Tasks featuring different interval times, the project objects are processed according to their interval time irrespective of their processing number.

General project objects

Name	Content	Description
	First line	The assigned project name appears here.
(CONF)	Configuration	The configuration level groups software and hardware nodes.
(SOFTWARE)	Software	The software level groups all resources and the associated resource objects.
(HARDWARE)	Hardware	The hardware object links to the hardware structure
(D-PS or PS, AC 900F P AC 900F, AC 900F L, AC 800F, AC 700F)	Process Station (PS)	A PS contains the CPU module which processes the programs configured under the resource. The type of a PS is defined in the hardware structure by allocating the resource to the hardware object. The short label **D-PS** indicates that a station has not yet been assigned in hardware manager to a physical station. **Following resource allocation**, the assigned station type is displayed: **PS** for a rack system, AC 900F P, **AC 900F** or AC 900F L for an AC 900F controller, **AC800F** for an AC 800F controller, **AC700F** for an AC 700F controller.
(D-PS/RED or PSR, AC 800FR, AC 900FR, AC 900FR L, AC 900FR P	Redundant process station	A redundant process station contains redundant CPU modules. The rack system uses two types of DCP 10 CPU modules. A redundant FieldController consists of two AC 800F. The two controllers are connected by a redundant link and appears the same as a PS to the user. When a PS is loaded, one of the two CPU modules becomes the primary CPU and the other becomes the secondary CPU. The short label **D-PS/RED** indicates that a station has not yet been assigned in hardware manager to a physical station. After it has been assigned, the station type **Following resource allocation** appears: **PSR** for a rack system and **AC 900FR /** AC 900FR L / AC 900FR P or **AC 800FR** for a redundant AC 900F or AC 800F controller.

Name	Content	Description
(D-OS or VIS)	Operator station	An operator station is a resource permitting operation and observation on a Personal Computer (PC) using the Freelance Operations program package. The code **D-OS** indicates that the station has not yet been assigned to a physical station. After it has been assigned, the station type **VIS** appears.
(800XA-AS)	800xA Aspect server	A console of the 800xA system can be used as operator station for a Freelance system. The data of the 800xA system is centrally stored in the database of an Aspect server. The new project tree element 800xA Aspect server (800XA-AS), representing the configuration database, must be configured to transfer data to the 800xA system.
(D-GS or GWY)	Gateway station (GS)	A gateway station enables data to be transferred to other systems through interfaces such as OPC. A gateway station is also set up to interface to a higher-level process control system such as Maestro. The code **D-GS** indicates that the station is not yet assigned to a physical station in the hardware manager. After it is assigned, the station type **GWY** is displayed.
(D-GS/RED or GWY)	Redundant Gateway-station	The redundant gateway is only available to interface with a Maestro UX system. It must be equipped with two type DCP 10 CPU modules; one of these becomes the primary gateway, the other becomes the secondary gateway. The code **D-GS/RED** indicates that the station is not yet assigned to a physical station in the hardware manager. After it is assigned, the station type **GWY** is displayed.
(OPC-S)	OPC server	An OPC server is used to import data from other systems to the Freelance project through an OPC interface.

Name	Content	Description
(P-CD)	Common display pool	Displays and logs set up under this project object must be available on all operator stations. The objects from the display pool are loaded only on those stations that have access to the process data needed in these displays and logs. Trend displays which are linked with an acquisition block must be assigned to one specific operator station.
(P-FB)	User function block pool	New classes of block are defined below this project object; these can be used in the configuration of the user programs like the function blocks contained in the firmware.
(P-MAC)	Graphic macro pool	Below this element all graphic macros are defined that can be used in the graphic display.
OPC_FB-LIB	Tag library	Library which hosts Tag types.
Pool		“Memory” for implausible project objects or project objects no longer required for processing, which can be returned to the process if required.

Project objects of a graphic macro pool (P-MAC)

Name	Content	Description
(MAC)	Graphic macro	Graphic display elements can be combined to form macros and used freely in graphic displays.
(STRUCT)	Structure node	Structural element of the operator station. To provide a clearer overview, the displays and logs from an operator station can be grouped together with this project element.

Project objects of a D-OS operator station

Name	Content	Description
(SFCD)	Sequential function chart display	Structured display presenting a program sequence configured in the Sequential Function Chart (SFC).
(OVW)	Overview display	Display for the rapid selection of displays and/or logs. Up to maximum of 4 x 24 displays or logs can be entered in an overview display.
(GRP)	Group display	The Group display is a group of several faceplates. It offers the user the possibility of displaying associated tags in a display.
(FGR)	Graphic display	Display of freely grouped static and dynamic display objects generated by the graphics editor.
(TR_D-OS)	Trend display	The Trend display is for the graphic display of values across a time axis. A total of maximum six trends can be displayed in one trend display.
(TS_D-OS)	Time scheduler display	The Time scheduler display shows the state of a Time scheduler and permits its operation. The display consists of a trend area for set point and actual value curves, a state field and the associated face plate.
(WEB)	WEB display	When calling up a WEB display in Freelance Operations, the local WEB browser is started and the link to the configured WEB page is activated.

Name	Content	Description
(OPL)	Operation log	Cyclical, manual, or event-related log containing maximum of 200 selected variables which are logged within a configurable text.
(SSL1)	Signal sequence log 1	Logging of system errors, fault messages, switching messages, operator instructions, and operator actions with continuous printout.
(SSLN)	Signal sequence log N	Logging of system errors, fault messages, switching messages, operator instructions, and operator actions in a log file. Printing is possible at the end of logging or manually.
(DCL_D-OS)	Disturbance course log	Disturbance course log belongs to the state logs. Its purpose is to log temporal sequences of selected analog and binary tags.
(REPORT)	Excel report	A report is configured in Freelance Engineering to acquire one or more samples of a set of variables.

Project objects of a process station D-PS or D-PS/RED

Name	Content	Description
(TASKLIST)	Task list	An object for separating the system tasks and the user tasks.
(Task)	Task	An object which controls the processing of the subordinate program lists and sequential controllers within the resource. A distinction is made between cyclical tasks and those which are processed only once for specific events. In addition, a default task can be configured for each resource. This task is always executed when none of the other tasks is being executed (cyclically or once only).
(TASK/RED)	Redundant task	All subsidiary program lists and SFCs within this task are executed redundantly. All tasks can be in redundant format. All the variables in a redundant task must be written through the process image mode.

Name	Content	Description
(SFC)	Sequential function chart program	This program is generated by using SFC. The assigned programs are executed using a structure.
(PL)	Program list	List of Function Block Diagram (FBD), Ladder Diagram (LD), Instruction List (IL), or Structured Text (ST) programs, which are executed according to their number in the project tree. The execution of the PL can be switched “Off” or “On”.
(IL)	IL program	Program which is generated using the IL language.
(FBD)	FBD program	Program which is generated using the Function Block Diagram language (FBD).
(LD)	LD program	Program which is generated using the Ladder Diagram (LD) language.
(ST)	ST program	Program which is generated using the Structured Text (ST) language.

Project objects of a OPC function block library

Name	Content	Description
OPC_FB-CLASS	Tag types from the Standard library and/or user defined Tag types	Every Tag type contains faceplates

3.2.2 Inserting project objects

Select the insertion position in the project tree

Edit > Insert above inserts a new object before the selected object on the same level in the hierarchy

Edit > Insert below inserts a new object under the selected object on the same level in the hierarchy

Edit > Insert next level inserts a new object one level lower in the hierarchy

Depending on the object selected in the project tree, the associated “Object selection” dialog box opens.

> Select the required object type and click **OK.**

3.2.3 Assigning project object names

Object names are assigned in the header of an object. Each object is assigned with a unique name. All uppercase and lowercase letters, figures and the special character “underscore” (_) are permitted. An object name can be up to maximum of 12 characters long, with the exception of the resources for which only four characters are allowed.

The object header can be edited by selecting:

Project > Header...

Use the comment editor to produce a free text of several pages long for each object. This comment can then be output with the project documentation.

Project > Comment

3.2.4 Display of project object states

The following states which are generated by subjecting them to a plausibility check can be detected by displaying the nodes of the individual project objects.

The path is closed; there are no more branches.

The path is open.

!

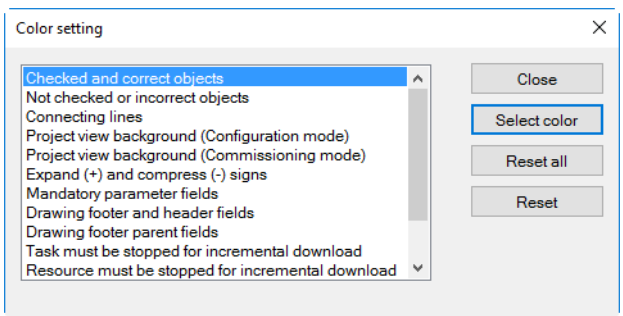
The path is closed; there are side effects below the displayed level. There are no more branches.

(Pink) Object has been changed; a plausibility check has not yet been performed or errors were found during the plausibility check. Path with arrow: During the plausibility check, objects modified compared to the previous configuration state were detected.
(Flooded green) Plausibility check completed correctly.

(Flooded red) Plausibility check completed correctly; modified objects with side effects on the resource were detected. To load these objects, you must load the resource.
(Flooded yellow) Plausibility check completed correctly; modified objects with side effects on the task were detected. To load these objects, you must load the related task.

Color settings of nodes

The colors of the nodes are preset and you can modify them in the project tree using: > **Options** > **Color setting...**



Color Settings.png

Close Accepts the modified color settings and exits the dialog.

Select color Opens a Color dialog box to change the color state. Select **Define custom colors >>** to compose Custom colors.

Reset all Resets all changed color settings to a default setting. The following colors can be set by default:

green	Checked and correct objects
magenta	Not checked or correct objects
black	Connecting lines
black	Expand + and compress - signs
red	Mandatory parameter fields
green	Drawing footer fields
pink	Drawing footer parent fields
yellow	Task must be stopped for incremental download
red	Resource must be stopped for incremental download
red	Resource/Task running partially
yellow	Current and configured value are different

Reset Resets the currently selected color state to the default setting.

3.2.5 Searching in the project tree

> **Edit** > **Search...** > Type a name or a part of a name in the Entry name field > select **Search**

The name to be searched for is entered either in full or in part in a Search dialog box. The names of all objects in the project tree are checked. The first object found is marked in the project tree. If the **Search** is pressed repeatedly, all objects whose name contains the character string entered will be found. The search is case-sensitive.

3.2.6 Expand, full expand, compress

To enhance the clarity of the project tree, individual project sections can be opened or closed. This is done by selecting the nodes ahead of the objects.

Expand

This is only possible if the node is selected with  . The node is displayed by one level

> Select node > **Edit** > **Expand**

Or

> Click on selected node

Full expand

This is only possible if the node is selected with  . The node is fully expanded.

> Select node > **Edit** > **Full expand**

Compress

Only possible if the node is selected with  . The node is minimized to an object.

> Select node > **Edit** > **Compress**

Or

> Click on selected node

3.2.7 Moving, deleting objects

Use menus in the project tree to cut, copy, insert or delete single or blocks of objects and specify paths or sub-paths. An exception to this rule is the project name and the system tasks. To move objects without the menus use the mouse.

Individual objects

> Click on the selected project object

Cut

Removes the object and saves it in the clipboard to insert later.

> Select project object > **Edit** > **Cut**

Copy

Saves the object in a clipboard to insert later at another position.

> Select project object > **Edit** > **Copy**

Paste

Copy or cut an object, before using the **Paste** command. If the insertion position is not permitted, the **Paste** command in the menu is disabled (highlighted in gray).

> Select insertion position > **Edit** > **Paste**

The Paste dialog opens to define the insertion position: **Above**, **Below** and, if required, **Level**.

Specify a unique name for each object that you copy or paste.

Delete

Deletes the selected objects from the project database. Any objects below the selected object are deleted along with it if they have also been selected; otherwise these objects are moved into the pool.

> Select project object > **Edit** > **Delete**

You are not asked if you really want to delete the objects if the node is displayed thus:

If the node is displayed thus  , the delete dialog opens and you are asked: "`Delete object? Programs are moved to the pool`?"

You can undo the delete operation with > **Edit** > **Undo**.

You cannot undo a delete operation after you have saved the project.

Cut and insert (Move)

> Click and hold the mouse button to select object a second time.

> Move mouse to insertion position.

An icon appears, indicating whether insertion is permitted  or not permitted ⃠.

Release the mouse button at the required position.

The Move dialog box opens to define the insertion position: **Above**, **Below**, and if required, **Level**.

Several objects (Block)

The selected objects are placed together in a frame and highlighted (in color) for further processing.

Blocks can be handled in the same way as individual objects (see above). However, there is a difference in mouse and keyboard operation.

> Select the object > click and hold the mouse button.

> Move the mouse to the next level (second next level, and so on).

> Release the mouse button at the required position.

3.2.8 Undo

> **Edit > Undo**

Undo only the last action executed.

3.2.9 Exporting and importing blocks

Reuse project sections in the existing project or in other projects by exporting and importing blocks.

Export block

> Select block in project tree > **Edit** > **Export block...**

Exports the entire content of the selected block to a PRT file which can reload by using the **Import block** command. Type the file name in the **Export partial project** dialog box. Access to the directory is by default. The previously active directory is called. The default directory for Freelance Engineering export files is **<FreelanceData>\export**.

When exporting a block, if the node in front of the selected object is compressed,

then while exporting the block all the objects below it are exported
or else
only the selected object is exported.

Import block

> **Edit** > **Import block...**

Imports the content of the block from a PRT file that previously generated by using **Import block** to the pool. From there, the entire block can be moved or individual objects to the required position in the project tree. Access to the directory is by default. The previously active directory is called. The default directory for Freelance Engineering import files is **<FreelanceData>\export**.

When importing a block, collisions may occur at tag and variable names. Automatic renaming is controlled through two entries in the Windows registry. Registry changes are only possible if you have administrator rights. Unqualified changes of the Windows registry may cause serious system stability problems.

Under HKEY_LOCAL_MACHINE\SOFTWARE\Hartmann & Braun\Freelance\DigiTool\SETUP you will find the entries “AutoRenameEAM” and “AutoRenameMSR” with the values 0 or 1. Value 0 means that names are usually not changed when importing. Variable names are always retained, tag names only if there are no collisions. If an imported tag name already exists in the project, the name of the imported module is extended by ....00. An additional import would result in ...01, and so on.

If one of the entries AutoRenameEAM or AutoRenameMSR has the value 1, a dialog box is opened during importing, asking you whether the names concerned must be changed.

If the answer is yes, the names are extended by ...00. An additional import would result in ...01, and so on.

If the answer is no, variable names are retained, tag names are removed if collisions occur in the existing project.

By retaining the names in imported project sections, it is easy to link several project sections.

If 16 character tag names are used in a project but only 12 character tag names are set in Freelance Engineering, the tag names will be cut at the beginning or end.

After an import, the imported variables are not allocated to a process station!

Import block as redundant

> Edit > Import block as redundant...

This function corresponds to the Import block menu item. In the import, all resources and tasks are converted into redundant resources or tasks. Access to all global variables in the imported programs can optionally be converted to an access

through the process image. This menu item is used to make existing projects “redundant”, in that the corresponding resources are exported as a block, deleted, and then imported again.

3.2.10 Pool for unneeded objects

The pool is a “memory” for incorrect project objects or for project objects no longer required in processing and which you may want to return to the process. The individual programs or entire structures can be saved. The processing options in the pool are identical to those in the higher-order process level.

If the objects are imported into the project, they are saved in the pool and must be moved them from there to the required resource or task.

3.2.11 Access rights and user groups

In the project tree, access rights for editing or modifying can be permitted to the individual or several project objects simultaneously in an Operator station. This determines whether a Freelance Operations user can only view or also operate the specified displays. The same applies to logs or trend displays.

In the same way as displays and logs, access rights can be permitted to tag names in the tag list.

If User Management is enabled, user can select to manage the user access via either Security Lock or Extended User Management. Security Lock is installed automatically during the Freelance installation, user can create the users and groups in the Security Lock interface. And for Extended User Management, it includes login Freelance via either Use Local Account or Use Domain Account.

For more information on User Management, refer to ***Freelance Engineering Manual User Management***.

Access rights to Freelance Engineering

The following access rights are available.

No access The user cannot configure or commission the project.

Configure The user can configure the project.

Commission The user can execute all commissioning functions such as load user program sections or modify parameters.

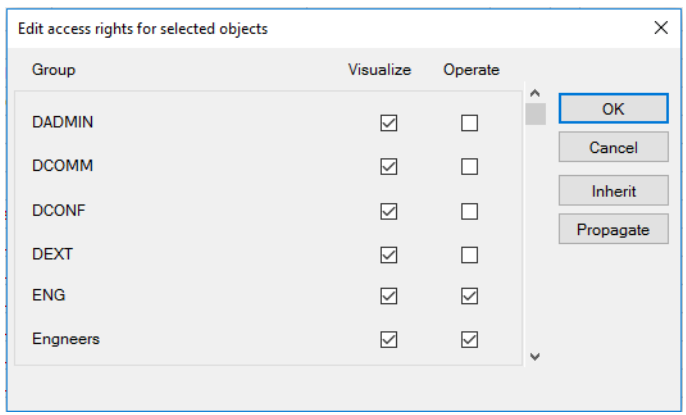
If Security Lock is enabled to manage the user access, up to 16 user groups with maximum 1000 users for each group are allowed.

This definition applies to each project which is processed on the corresponding Freelance Engineering PC. Refer to ***Freelance Engineering Manual User Management*** for detailed information of user rights modification.

Access rights to Freelance Operations via User Management

The following access rights are defined for Freelance Operations for the installed user groups:

> Edit > Access rights



Access Rights.png

No access ☐ The user cannot operate or observe on the operator station.

Visualize ☑ The user may open displays and logs but not operate them.

Operate ☑ The user cannot only open displays and logs but also operate values, states of visualized variables and tags.

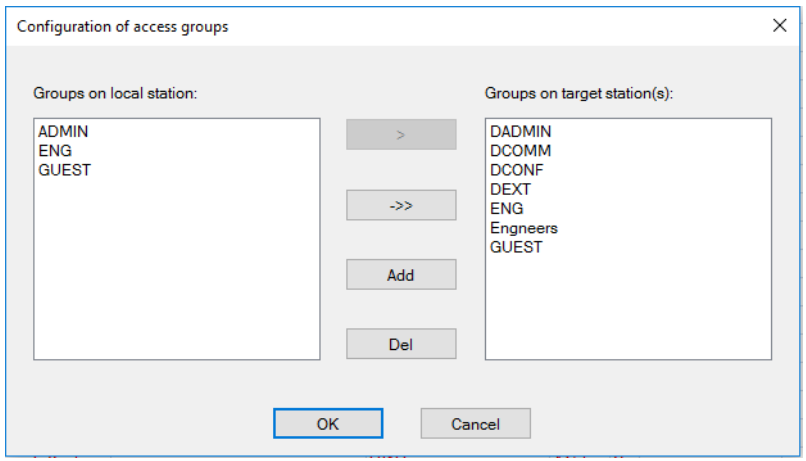
Inherit Inherits the access rights setting from the higher-order objects.

Propagate Propagates access rights to all lower-order objects.

User groups

The user groups on the engineering station (local station) can be assigned to target stations in the project tree. A target station in this case means all operator stations in the project.

> **Edit > User groups**



User Groups.png

>	Adopts the selected group at the local engineering station for the operator stations.
>>	Adopts all the groups in the local engineering station for the operator stations.
Add	Adds a new user group for the operator stations.

User groups assigned to target stations in the project tree must also be made known to each operator station using User Management.

Del	Deletes User group from the selected operator station.

At the operator station (Freelance Operations), only the user's name of the access rights in the Status bar is displayed. The user's actions are logged in the signal sequence log.

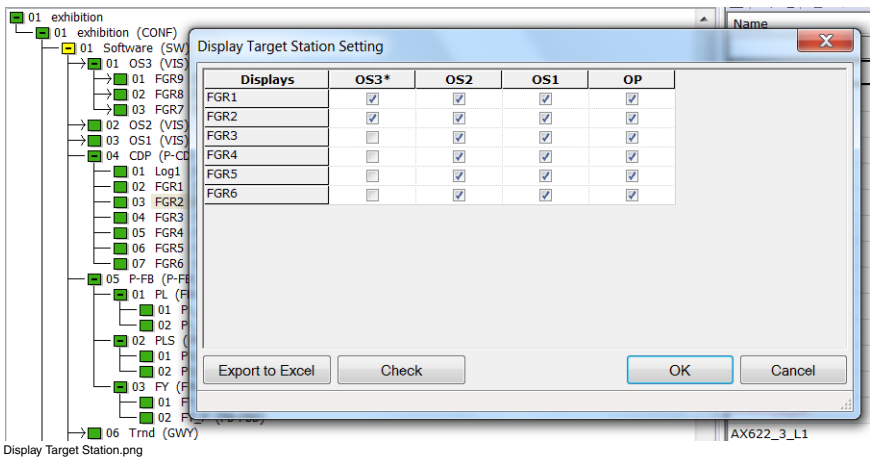
3.2.12 Display target stations

In Freelance Engineering, the graphic display in the Common Display Pool (CDP) can be assigned to each operation resource, and be downloaded to the Freelance Operation stations.

Through Display Target Station Setting dialog, user will be able to assign individual displays available in CDP to specific Operation station.

> Right click on graphic display in CDP > Display target station...

User can assign the FGRs to the OS by ticking the checkbox or unassign the FGR by leaving the checkbox unchecked. Select all means assign all the FGRs to every Operator station, while unselect all means assign no FGR to Operator station.



Display Target Station.png

The Operation Station (OS) which marked with a "*" after OS name is a lite version OS, and it supports up to 5 FGRs, including the FGRs under lite OS node.

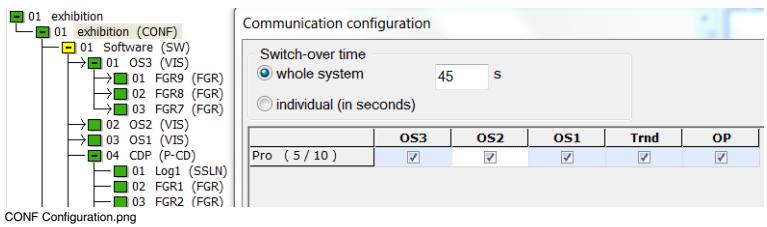
At Display Target Station Setting dialog, if user assigns a 6th FGR to the lite OS, a dialog will pop-up as "The amount of assignable Free Graphic Display has been exceeded. Only 5 Free Graphic Displays can be assigned.

And under OS node, user won't be able to insert FGR if the total FGR number reaches the limit of 5 for a lite station. Error shows "The limit of Free Graphics (FGR) supported by a Freelance Operations Lite Station has been reached.

During the lite station configuration, if user ticks the checkbox of Exclude FGR from CDP, all the FGR assignation in Display Target Station Setting will not be counted. As a result, even if the FGR number exceeds the limit of 5 FGRs, no error shows when carrying out the plausibility check.

After assigning FGRs to the specific Operation Station, user needs to check the configuration under Conf node. For each OS, check if it can access the D-PS resources which have variables and tags allocated to. Plausibility check passes if access is allowed. Otherwise, error will pop up, saying "The Variable xxx allocated to D-PS yyy is not assigned to D-OS zzz" or "The Tag xxx allocated to D-PS yyy is not assigned o D-OS zzz".

> Right click on CONF Node> Edit...



CONF Configuration.png

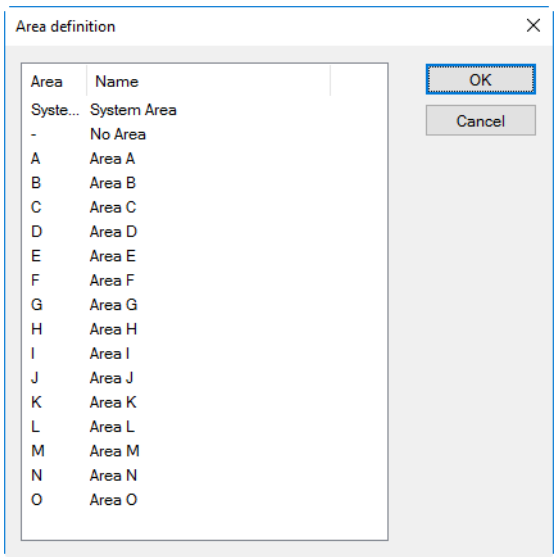
The settings of display target station can be exported to csv file via Export to Excel button. After pressing Export to Excel button, a dialog pops up to let users specify the exported file path. In the exported Excel, "1" means FGR is assigned to the OS, "0" means FGR is unassigned.

3.2.13 Area definition

A name with up to 16 characters can be assigned to all plant areas. If no dedicated names are specified, the plant areas will be designated "Area A" to "Area O". In the tag list, plant areas are always described with their long names, while during filter selection in Freelance Engineering, the abbreviations "A" to "O" are displayed together with the long names. In Freelance Operations with the new user interface, the used plant areas are always displayed with their long names. In the classic user

interface, the short designations of the plant areas are still used due to restricted space.

> **System > Area definition...**



Area Definition.png

No names can be changed in the first two lines.

In Freelance Operations, all messages originating from the system are combined under “System”. Measuring points not assigned to any area are combined under “No area”.

The first column of the following lines lists the possible areas from A to O to which new names can be assigned.

> Double-click the area > Edit name in the second column (maximum 16 characters).

3.3 Configuring the project objects

When a new object is added to the project tree, assign a name to the object and, if required, a short comment. These parameters can be modified later in the menu. Depending on the object, additional information is displayed as follows:

- Type of object
- Version (date, time of creation or last modification)
- Number of subordinate objects
- Processing sequence

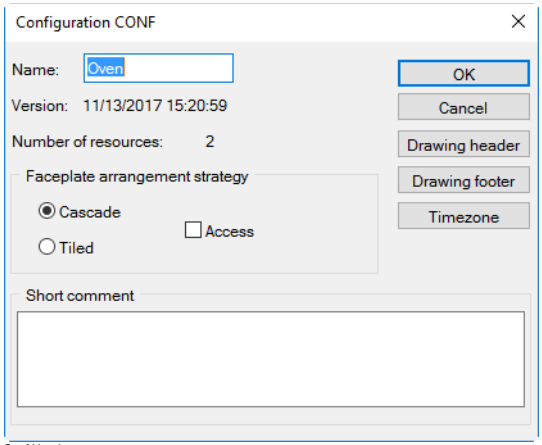
Drawing header or footer for all objects can be defined. For more information on drawing header or footer, refer to Section 6, Documentation on page 259.

> Select project object > **Project** > **Header...**

3.3.1 Configuration (CONF)

The Configuration object summarizes information related to the entire user program, and all subordinate project resources. The configuration contains all the station resources (process, operator and gateway stations) along with the user-defined function blocks pool, OPC Function block library and the global display pool.

The communication links of all stations of the Freelance project can be configured at this project object/or within the hardware manager.



Conf Header.png

Name Maximum 12 characters

Version Date and time of object creation or modified

Number of resources

Number of resources configured in the project tree

Short comment

Maximum 159 characters

Faceplate arrangement strategy

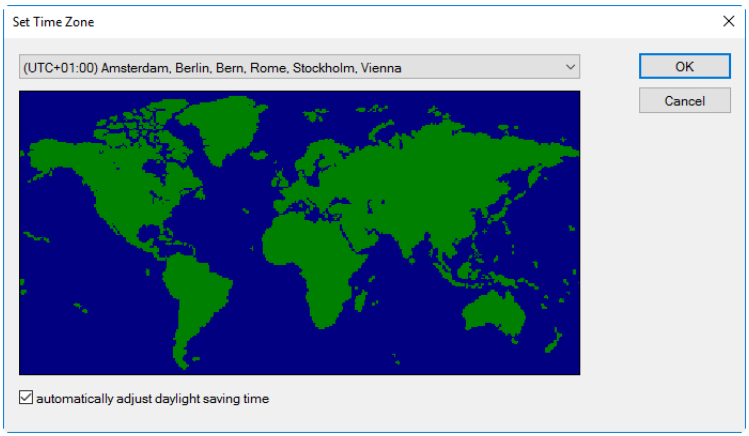
If several faceplates are shown on a Freelance Operations station, they can be arranged in tile or cascade mode. The default setting for all Freelance Operations stations is defined.

Access Configuration whether the Freelance Operations user is allowed to switch the faceplate display mode between cascade and tile.

Drawing Header/Footer

Refer to Section 6, Documentation on page 259.

Timezone Set the time zone and daylight saving time for the whole project.



Set Timezone.png

The dialog for setting the time zone and daylight saving time is structured in the same way as the Windows dialog.

The time zone of Freelance Engineering may be set differently from of the time zone setting in the Windows operating system. To activate the local (i.e. different) time zone in the Freelance project, the time zone must be selected in the CONF node of project tree.

The selected time zone in Freelance controllers differs from the Freelance Engineering project and Windows operating system. The Freelance Engineering project displays the “version error” during the plausibility check. All controllers including the controller coldstart must perform a “load whole station” plausibility check (refer to Load whole station on page 237). The gateways and Freelance Operations require a “load changed objects” plausibility check to align to the new time zones (refer to Load changed objects on page 239).

3.3.2 Configuring the hardware and software nodes

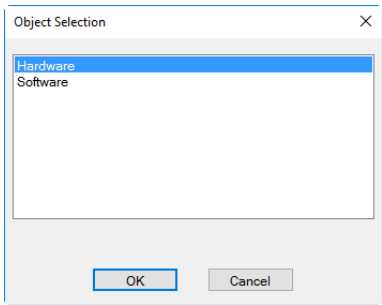
The hardware and software nodes are configured below the CONF node.

Configuring hardware node

To insert the hardware node, proceed as follows:

> Select CONF node > right-click to insert the next level

The Object selection dialog box appears as shown in the following figure.



Insert Hardware.png

> Select Hardware in the Object selection dialog box > click **OK**

The hardware node is added below the CONF node.

Plausibility check of hardware node will display the errors of the hardware structure.

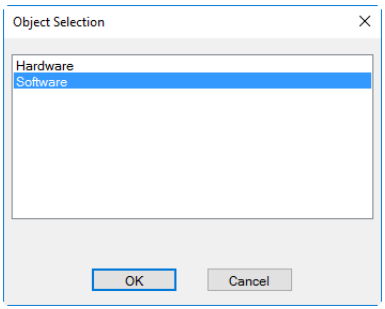
The Hardware node is only an option. Double-clicking on the node the hardware structure can be reached. Alternatively the hardware structure can be reached through the toolbar or the menu bar. For more information on hardware structure, refer to Section 4, Hardware structure on page 147.

Configuring software node

To configure software node, proceed as follows:

> Select CONF node > right-click to insert the next level.

The Object selection dialog box appears as shown in the following figure.



Insert Software.png

> Select Software in the Object selection dialog box > click **OK**.

The software node is added below the CONF node.

3.3.3 Resources

The first structural level below Software node is formed by the resources, which represent the various different stations in a project. For the processing of the process itself there are the **D-PS (Process station) resources**, for operation and observation of the process there are the resources D-OS (operator station), and **800XA-AS (800xA Aspect server)**, for interfacing to external systems there are **D-GS (Gateway station)** resources, for integrating data from external systems there is the **OPC server** resource.

Data exchange between the process stations is performed by means of lateral communication. Each process station is equipped with 10 connections for data exchange with operator stations and gateway stations. If more stations are configured, then the effective communication links of all the subscribers in a Freelance system are specified in a separate parameter definition dialog box.

There are also additional structural elements such as the user-defined function blocks pool P-FB, the graphic macros pool P-MAC and the **Common display pool P-CD**.

The resources in the project tree are assigned in the hardware structure of a station. Use the network configuration (menu item in the hardware structure: **Hardware structure > Network**) to set the IP addresses and resource IDs. They are accessible through the Control Net on commissioning. Also refer to Section 4, Hardware structure on page 147, Network configuration on page 200, Resource allocation on page 187 and Section 5, Commissioning on page 215, Loading the project on page 236.

It is also possible to configure more than one resource on a PC. For example, a D-OS resource and an OPC Server can be operated simultaneously on the same PC, provided these two resources have different resource IDs so that they are addressable by the system.

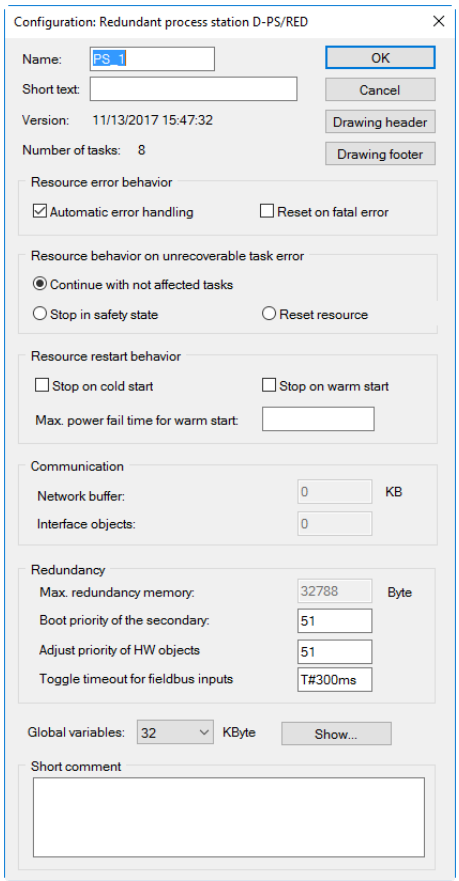
3.3.4 Process station

Process stations (PS) are added with **resource types D-PS** and **D-PS/Red** (non-redundant and redundant stations respectively). Correlation to the physical stations is carried out in the hardware manager. PS can be selected for rack systems, AC 900F controllers (**AC 900F, AC 900F L, AC900F P**), AC 800F controllers (**AC 800F**) and AC 700F controllers (**AC 700F**) can be selected for connecting Fieldbus modules. The network addresses and resource IDs of the stations are also defined within the hardware manager, and in commissioning mode the configured program modules are loaded from the project tree into the relevant stations. Refer to Section 4, Hardware structure on page 147, Network configuration on page 200, Resource allocation on page 187, Section 5, Commissioning on page 215, Loading the project on page 236 and Commissioning procedure on page 218.

The execution of user programs within a process station is controlled by **tasks**. Within a task the sequence of user programs are defined either by **program lists** or by a structure that are configured using the **sequential function chart**. While the programs in a program list are run cyclically, sequential function chart programs are run only for specified periods of time according to their structure.

The user programs within a process station, both under a program list and under a sequence flow, are created using the IEC 61131-3 programming languages FBD, LD, IL or ST.

> Select object in Project tree > **Project** > **Header...** or double-click



Pprocess Station Interface.png

Name	Maximum four characters
Short text	Maximum 12 characters
Version	Date and time of object creation

Number of tasks

The number of tasks configured for this resource

Resource error behavior

Automatic error handling

When error correction is on, errors occurring in the user program at run time (for example, a division by zero) are automatically corrected. For more information, refer to ***Engineering Manual Process Stations, Automatic Error handling for the task***.

Reset on fatal error

If an internal error is detected in the sequence while the user programs are being processed in the resource, the CPU module will suspend processing. In the case of a non-redundant system, the outputs revert to the configured safety values, and the Failure LED flashes on the CPU module. This parameter allows you to set the CPU module to exit from this type of error state after 10 seconds by means of a cold start.

Resource behavior in case of unrecoverable task errors

The radio buttons are used to define the process station behavior in case an unrecoverable error occurs during the runtime of a user program (for example, division by zero without activated error correction).

Continue with not affected tasks

Only the task containing the user program with the error will be put to the state “not executable”. All other unaffected tasks continue the program execution. This behavior is compatible to the task behavior of previous Freelance versions. This selection is the default setting.

Stop in safety state

The station reverts to the safety status to prevent the output of inconsistent data to the process through unaffected tasks. The CPU module suspends processing, the outputs of the I/O modules revert to their preset safety values (this holds for rack modules as well as for fieldbus modules). For redundant process stations, there is a redundancy switchover. A manual reset is required to resume operation of the stopped station.

The procedure for leaving cyclically entered safety state in AC 900F and AC 800F controllers:

Press **Run/Stop** button to stop, perform controller reset through **Reset** button or power cycle. Controller performs restart and enters Stopped state. Load corrected configuration to the controller and press **Run** button to enter Run state.

The procedure for leaving cyclically entered safety state in AC 700F controller:

Power off the controller, press **Run** button and hold it to Power on again. Controller performs restart and enters Stopped state. Load corrected configuration to the controller and press **Run** button to enter Run state.

Reset resource The station is reset to safety status. For a redundant station, there is a redundancy switchover. The station with the error is reset and automatically reboots after 10 seconds. Depending on the error type, rebooting is done either through cold start or through an initialization of the station.

Resource restart behavior

If the station is reset either manually or automatically, it will first start to reboot. In case this should be impossible (for example, due to a corrupt configuration), the station will delete its entire configuration and wait in the initialization state for a configuration to be loaded. If "*Reset Resource*" is configured, the post-restart behavior is predicted based on the LEDs.

State	Failure/ERR LED	Run/Stop LED
Fatal error	Flashing red	Flashing green
Stop on cold start enabled	Off	Steady red
Stop on cold start disabled	Off	Steady green

Stop on cold start

After a cold start, the resource does not switch automatically to the ‘running state’; it remains in the ‘cold start stopped’ state before computing the cold start task, and must be started from Freelance Engineering.

Stop on warm start

After a warm start, the resource does not switch automatically to the ‘running state’; it remains in the ‘warm start stopped’ state before computing the warm start task, and must be started from Freelance Engineering.

Max. power fail time for warm start

If the power fail time of the controller is above the configured value, the controller will boot with a cold start. If it is below or if no time is configured, the controller will boot with a warm start.

Communication

Communication management parameters. Every communication link requires internal system resources such as additional memory or system objects.
For AC 800F and rack-based process stations you can also set the *Network buffer* and *Interface objects* boot parameters. For more information refer to the ***Engineering Manual Process Stations***.

Redundancy This information is available only for redundant resources.

Max. redundancy memory

Maximum amount of memory in bytes that the project requires for the transfer of redundancy data.

Boot priority of the secondary

To synchronize a redundant station, the bootstrapping of the second (secondary) CPU is initiated by the first (primary) CPU. This procedure is performed ‘simultaneously’ with the execution of the configured programs. This parameter controls the division of CPU utilization between normal program execution and bootstrapping the secondary. The higher this value is set, the more quickly the redundant process station will reach the state *sync*, and the more significantly normal program execution will be affected.

Adjust priority of HW objects

The objects configured in the hardware structure balance their redundancy data by means of a special mechanism. This parameter is used to specify the division of CPU utilization between normal program execution and this data balancing.

Toggle timeout for fieldbus inputs

Following a redundancy switchover, the new initial data of the linked Fieldbus devices are acquired. The time required for this depends on the devices that are connected and the transfer rates used. This parameter determines the maximum length of time that the system must wait for a new data. After the amount of time set here has passed, program execution will be initiated regardless. If not all of the data is successfully re-acquired, a system message will be generated.

Global variables

Refer to Section 5, Commissioning on page 215, Show global variables on page 228.

Size Currently 64 kilobytes

Show Shows the memory assignment of the resource

Short comment Maximum 159 characters

Drawing header and footer

Refer to Section 6, Documentation on page 259 and Documentation on page 137.

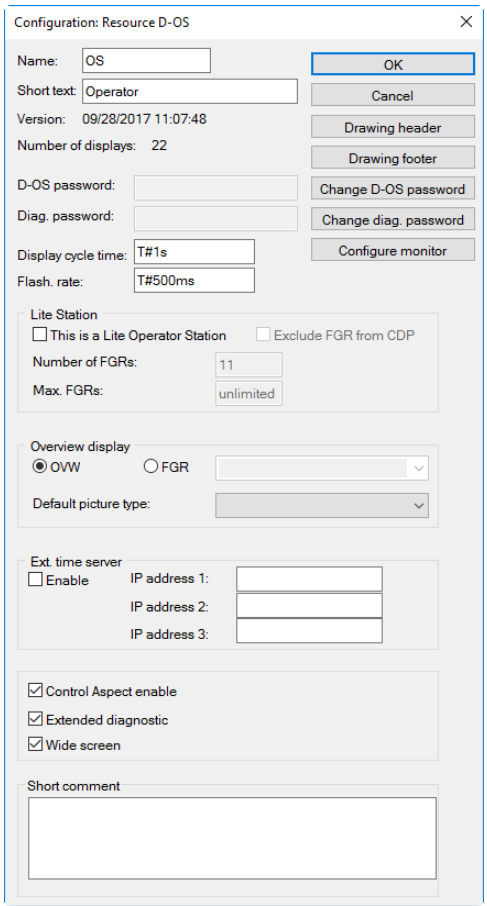
3.3.5 Operator station D-OS

The **D-OS** resource is provided for operation and observation of the process by an operator station using Freelance Operations software. Operator station assigned to a physical station is indicated by the code **VIS**.

Standard operating facilities (for example, **faceplates**), are provided for all known tags and variables on the operator stations. In addition, graphic displays, trend displays, SFC displays, WEB displays, logs, and reports are configured and structured for the Freelance Operations operator stations.

External process stations are time-synchronized by an operator station.

> Select operator station object in Project tree > **Project** > **Header...**



OS Interface.png

Name Maximum four characters.

Short text Maximum 12 characters.

Version Date and time the object was created.

Number of displays

Number of displays and logs configured in the resource.

D-OS password

Type a password that allows the operator to exit from Freelance Operations. If no password is specified, any operator can close Freelance Operations.

Diag. password

Type a password that allows the operator to switch to the Freelance Operations diagnostic mode. If no password is specified, the default password “diag” can be used to switch into diagnostic mode.

Configure monitor

Configure monitor button is the configure entry for Quad Monitor which allows user to select specific display types on up to four monitors. Please be noted that the configured monitor number should be no more than the licensed monitor number in f2k.key.

Display cycle time

The display cycle time determines the frequency with which the data in the message page, system display and the faceplates on the operator station is updated.

Flash rate The frequency with which flash colors are updated in the displays.

Lite Station

This is a Lite Operator Station

Lite Operator station supports up to 5 FGRs. If the FGRs in OS and Common Display Pool exceed the limitation of 5 FGRs, the plausibility check will fail with an error saying “FGR number exceeds the limit of FGRs supported by a Freelance Operation Lite”. Check the checkbox if user is using a lite Operator Station.

Exclude FGR from CDP

With this option checked, the FGRs in Common Display Pool (CDP) will not be counted into the 5 FGRs limitation.

Numbers of FGRs

It displays the current total number of FGRs in both OS and CDP.

Max. FGRs

The maximum FGRs supported for Lite Operator Station.

Overview display

OVW ◉ The configured overview display is displayed as an overview display in Freelance Operations.

FGR ◉ A graphic display is shown as overview display in Freelance Operations. Enter the name of the graphic display directly or select from the list.

Default picture type

A certain default display type or <no> default display type can be selected for each operator station. If no default display type is assigned in the display access, the display assigned to this type is used as default display.

Ext. time server

External stations with IP addresses 1/2/3 are time-synchronized through the operator station.

Refer to External timer server on page 133.

Enable Switch on time synchronization

Control aspect enable

The use of the control aspect of a tag, or to display the step or transition program in SFCs can be set in the resource header per Freelance Operations station.

In addition to the OPC server (or Trend server), the Control Aspect requires the current CSV file of the project. The Freelance Engineering generates the current CSV file in the background when the plausibility check is initiated from the uppermost project node.

In case of large projects, the time required to generate a CSV file may no longer neglected. For that reason the generation of CSV files is connected to the plausibility check of the project node. For example, if the plausibility check is started from CONF node, no CSV file will be generated.

During commissioning, several plausibility checks can be initiated from the CONF node - or deeper - and only occasionally, but at least once in the end, from the project node. The generated CSV file is loaded to the Freelance Operations PC with the download to the Freelance Operations station.

Extended diagnostic
The “Extended diagnostic” option is selected to display the extended DTM information in Freelance Operations when diagnostic mode is selected in system display.

Wide screen Here, “Wide screen” can be selected as default setting on project level. If a new operator station is then created in the project, “Wide screen” will automatically be preset. If a new group display is then created under this operator station, “Wide screen” will also be automatically preset for this group display.

Short comment Maximum 159 characters

Drawing header and footer
Refer to Section 6, Documentation on page 259 and Documentation on page 137.

3.3.6 Gateway station

The Gateway stations **D-GS** are used to make data from the Freelance system available to other systems. All the data from the Freelance system can be read and written through a gateway station. In addition to each gateway station in a Freelance system, the appropriate server software from the add-on packages OPC server and Trend server must be installed in the network.

For example, if a gateway station of **OPC gateway** type is configured in a Freelance system, then the OPC server software must be installed on a PC that is linked on the network with the Freelance process stations.

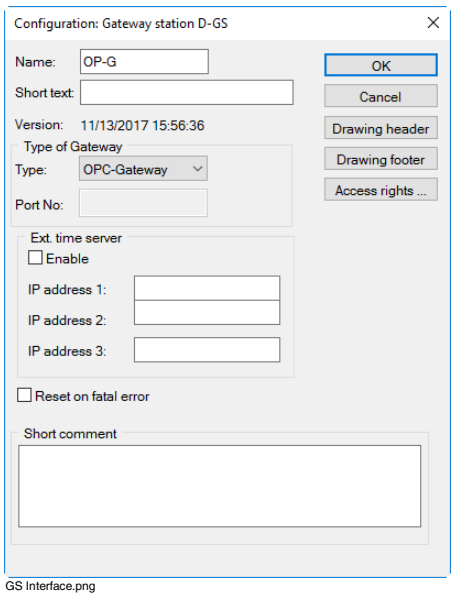
For each gateway station it is configured which tags and variables are available to the other system for reading and/or writing through the gateway.

Correlation to a physical station is carried out in the hardware structure. An assigned gateway station is represented by the label GWY. In the course of

commissioning, the data specified is made available to the other stations by loading the gateway station.

Refer to Section 4, Hardware structure on page 147, Network configuration on page 200 and Resource allocation on page 187, Section 5, Commissioning on page 215, Loading the project on page 236 and Commissioning procedure on page 218.

> Select Gateway station object in Project tree > **Project** > **Header...** or double-click.



GS Interface.png

Name	Maximum four characters
Short text	Maximum 12 characters
Version	Date and time of object creation
Type *DCP*	Type of gateway. The following gateway types can be selected: A DCP gateway is used for interfacing to the Maestro UX process control system. It runs on a DCP02 CPU module and is installed in one of the process stations. Only the gateway type DCP can be

assigned to a redundant gateway resource, and two DCP10 CPU modules must be used.

OPC An OPC gateway is required to implement a link to another system through the OPC interface. This runs on the PC on which the OPC server software is installed. A redundant interface can be achieved by configuring two OPC gateways in the project tree.

TRN A TRN (trend) gateway is required when trend displays are used in the system without trend data acquisition (refer to ***Freelance Engineering Manual Operator Station Configuration***). The Trend server software must be installed on the PC.

UNI A UNI gateway is required if data from a Freelance system are to be made available through Data Message Application Interface (DMS API).

Ext. time server External stations with IP addresses 1/2/3 are time-synchronized through the gateway station.

Refer to External timer server on page 133.

Reset on fatal error

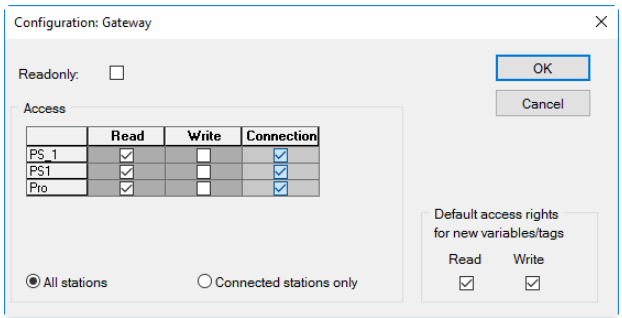
If an internal error is detected in the sequence while the user programs are being processed in the resource, the CPU module will suspend processing. This parameter allows you to set the CPU module to exit from this type of error state after 10 seconds by means of a cold start.

Short comment Maximum 159 characters

Drawing header and footer

Refer to Section 6, Documentation on page 259 and Documentation on page 137.

Access rights of a gateway



Gateway Access Rights.png

Readonly ☑ The gateway can access the process variables only in read-only mode. Any attempts to write to process stations and to variables and tags are ignored.

Access For each process station already configured in the project, the type of access (read-only or read/write) is specified. If a process station is configured here as read-only, any write accesses to the variables and tags are ignored. If *read-only* mode is specified for the complete gateway, this overrides any write access defined here.

Default access rights for new variable/tags
All newly-installed variables and tags are assigned by default the access rights configured here for this gateway.

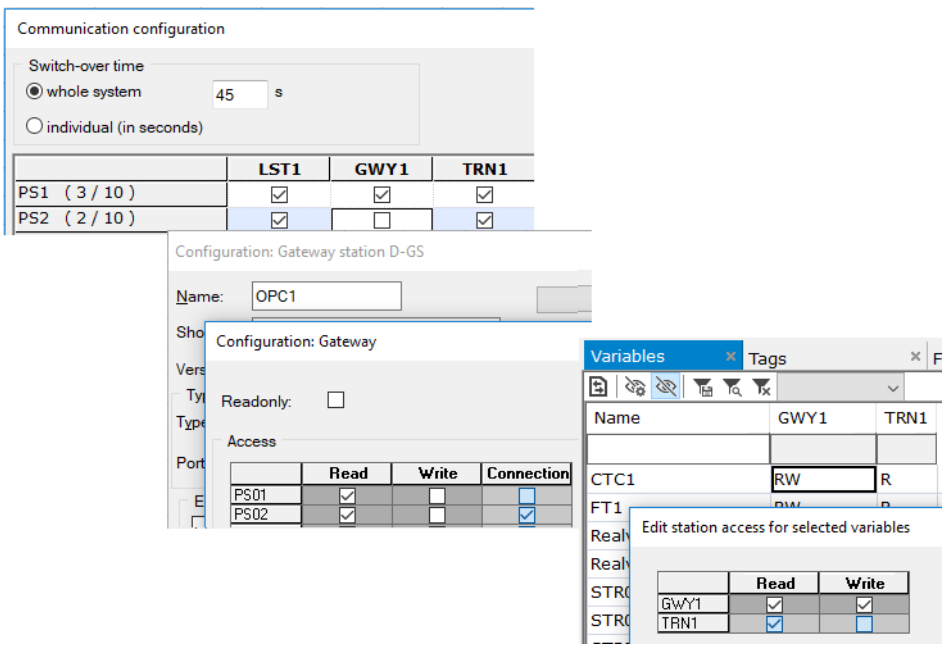
Variables that are written through a gateway may not be written simultaneously through the process image. This results in the values of these variables not being capable of redundancy. If these variables are used in a redundant task they must be mirrored to other variables.

The access rights to variables and tags are organized hierarchically:

- The prerequisite for communication between the process station and the gateway station is that this is permitted in the communication configuration dialog.

- The configuration of the gateway access rights configures whether only read or read and write accesses are possible.

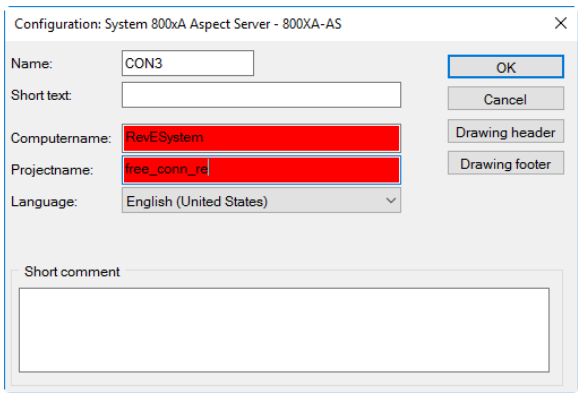
- In the station view it is possible to configure for each individual variable or tag, whether it can be read and / or written over the gateway. It should be noted that the configuration of the gateway access rights is leading. E.g. if a variable is configured for read and write (RW) access, but in the gateway access configuration only read is permitted, this variable can only be read.



CommunicationConfiguration_us.png

3.3.7 System 800xA Aspect server 800XA-AS

System 800xA Aspect server can be used as operator station for a Freelance system. The configuration data of System 800xA Aspect server system is centrally stored in a database. The project element 800XA-AS, representing the configuration database, must be configured to transfer data to the 800xA Aspect server system.



800xA Aspect Server.png

Name Maximum four characters

Short text Maximum 12 characters

Computername Name of the PC in the network containing the 800xA configuration database.

Projectname Name of the Operator workplace project into which this Freelance system shall be integrated.

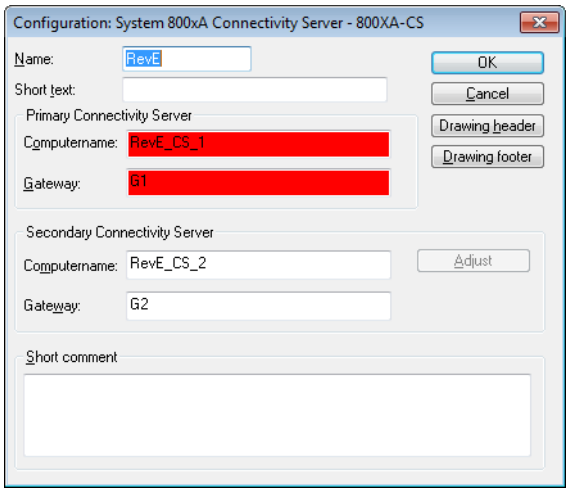
Language Select the language of installation for the Freelance project, usually the language of the 800xA system.

Short comment Maximum 159 characters

3.3.8 Operate connectivity server

The database of the 800xA system is configured with the project object Aspect server (800XA-AS). The connectivity servers (real-time data acquisition systems), are available in the system as data connection. This data acquisition software does not normally run on the same computer on which the database is installed. At least

one Freelance Connectivity server (800F-CS) element must be configured below a 800XA-AS element.



ConnServ_us.png

Name Maximum four characters

Short text Maximum 12 characters

Primary/Secondary CS

Computername Name in the network of the PC which has the Connectivity server software for a Freelance system.

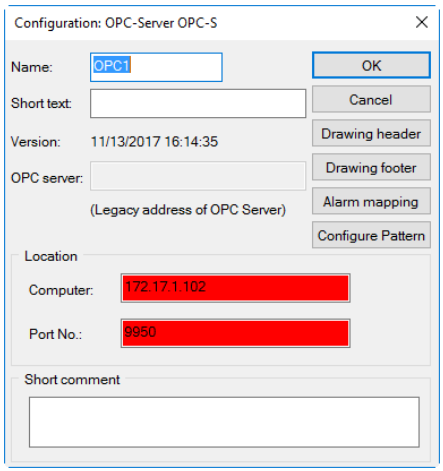
Gateway 1/2 Enter the name of a type-OPC gateway station or select from a list after pressing the **F2** key. A second OPC gateway only needs to be entered if a redundant link is to be established.

Adjust The configuration of the second gateway is matched with the configuration of the station entered under *Gateway 1*. Any access rights previously set up for this gateway will be lost.

Short comment Maximum 159 characters

3.3.9 OPC server (OPC-S)

The OPC server node is used to configure third party OPC servers in the Freelance system. The following parameter dialog box is used to configure the OPC-S node in the project tree.



OPC Interface.png

Short text Maximum 12 characters

Version Date and time of object creation

Location Specifies the computer on which the OPC software is installed.

Computer Name of the computer where OPC server is running.

Port No Port of the computer.

Short comment Maximum 159 characters

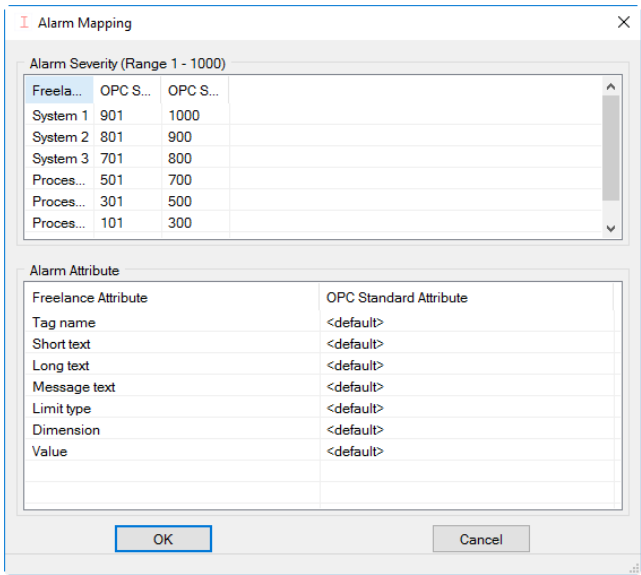
Drawing header / footer

Refer to Section 6, Documentation on page 259.

Alarm mapping

Freelance Engineering provides an option to map the attributes of the OPC AE Items to the Freelance system attributes.

Specify the OPC AE severity Range to be mapped for each Freelance priority. The range can be defined in the columns *Start of OPC range* and *End of OPC range* as shown below.



Alarm Mapping.png

Alarm severity

Freelance priority levels

The following priority levels are available in the Freelance system for grouping of OPC severities.

Priority level	Message type	Comments
S1-S3	System message	System messages have the highest priority level and are subdivided into three message groups S1-S3
1	Fault message	Messages of this type are used to indicate faults. For example, Alarm limit setting being violated
2	Fault message	“

Priority level	Message type	Comments
3	Fault message	Messages of this type are used to indicate faults. For example, Alarm limit setting being violated
4	Switch message	Messages of this type are used to indicate switch events. For example, valve open/closed.

Start of OPC range
Lower limit of severity range

End of OPC range
Upper limit of severity range

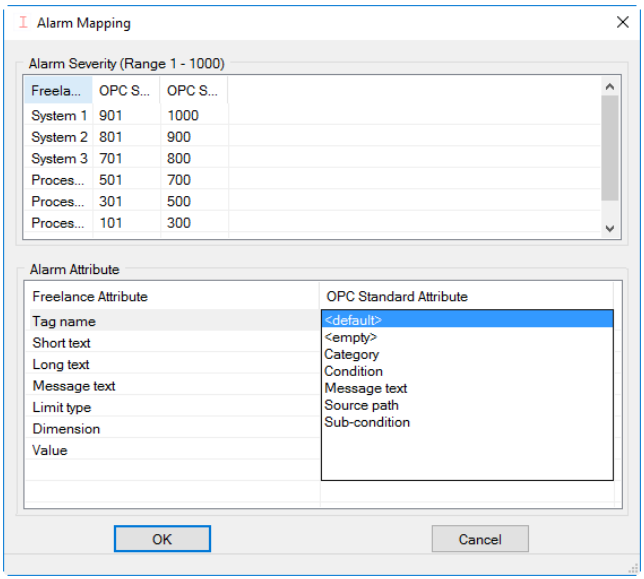
Alarm attribute

This property is used to map the OPC alarm attributes to the Freelance attributes. OPC server provides the following standard attributes for each OPC item:

- Category
- Condition
- Sub condition
- Message text
- Source path

These attributes are available for mapping to Freelance Alarm attributes.

The constraints on the Freelance alarm attributes must be considered while mapping the OPC alarm attributes.



Alarm Mapping Attribute.png

The Freelance alarm attributes are as follows:

- Tag name
- Short text
- Long text
- Message text
- Limit type
- Dimension
- Value

From OPC standard attribute list box, select the OPC attributes for mapping to each of the Freelance Operations alarm attributes. In addition to the OPC alarm attributes, two more options: <default> or <empty> can be selected.

The default values for the Alarm attribute mapping are as follows:

Freelance alarm attribute	Default mapping
Tag name	<default>
Short text	<default>
Long text	<default>
Message text	<default>
Limit type	<default>
Dimension	<default>
Value	<empty>

<default>

If <default> mapping for an attribute is selected, depending on the OPC item definition in Freelance (that is Tag, Variable, others), corresponding fields are mapped as shown in the table below:

OPC item definition > Freelance attributes	**Tag-Alarm** (AE item is used in a tag instance)	**Variable-Alarm** (AE item string is used as a DA item string in a variable)	**Free-Alarm** (AE items string is used nowhere)
Priority	Severity mapping	Severity mapping	Severity mapping
Status	OPC status	OPC status	OPC status
Time stamp	OPC time stamp	OPC time stamp	OPC time stamp
Area [16]	Defined area of theTag	“No Area”	“No Area”
Tag name [16]	Tag name	Variable name	End of OPC source string-16 characters
Short comment [12]	Short comment from tag definition	“-----”	“-----”
Long comment [30]	Long comment from tag definition	Beginning of variable comment	Beginning of OPC message text
Value [12]	“-----”	“-----”	“-----”
Dimension [8]	“-----”	“-----”	“-----”

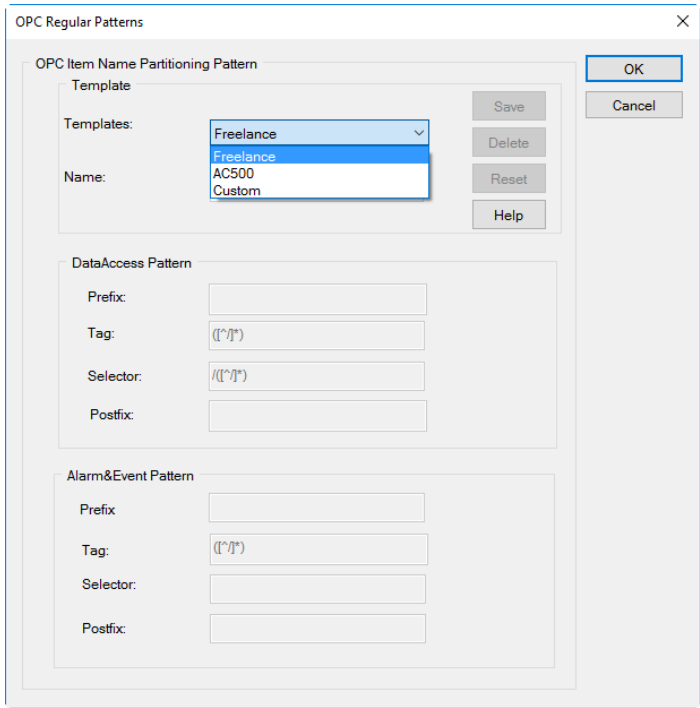
OPC item definition > Freelance attributes	**Tag-Alarm** (AE item is used in a tag instance)	**Variable-Alarm** (AE item string is used as a DA item string in a variable)	**Free-Alarm** (AE items string is used nowhere)
Message text [8]	Beginning of OPC message text	Beginning of OPC message text	Beginning of OPC message text
Alarm type (LL,L,H,HH) [8]	Beginning of OPC sub condition	Beginning of OPC sub condition	Beginning of OPC sub condition

<empty>

The <empty> attribute is assigned to the Freelance system alarm attribute “Value” as there is no corresponding attribute available from a standard attributes of a OPC server.

For more information on the message list format, refer to ***Freelance Operations***, ***Operators Manual, Messages and Hints***.

OPC regular patterns (Configure pattern)



Regular Pattern Configuration.png

Various OPC server expose their DA and AE items of function blocks in a different way.

CoDeSys OPC server uses “.” as a separator and exposes DA items using this separator in the form “PLC.TaskName.TagName.selector”. Freelance server uses “/” as separator and exposes its DA items in the form “TagName/Selector”.

AE item patterns are also different for different servers. CoDeSys OPC server exposes AE items like “PLC.TaskName.TagName.selector” with condition “LIMIT_EXCEED”. Freelance server exposes AE items with only tag name (with out selectors) and condition.

For example,

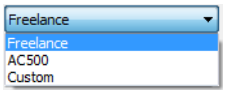
CoDeSys OPC server Alarm Item
PLC1.Task1.PID1.ACTUAL.LIMIT_EXCEEDED

Freelance OPC server Alarm Item

PLC1/PID1/HH_1

This dialog gives an option to set regular expression patterns to identify various parts in DA and AE items for a particular OPC server.

Templates Pre-defined template for Freelance and AC500 OPC servers



Configure pattern template_us.png

For other third party servers select *Custom* and configure the patterns.

3.3.10 User function block pool P-FB

New block classes are defined below this project object. These classes are used when configuring the user programs like the function blocks contained in the firmware. This means that the range of function block types in Freelance system is supplemented by user-defined function blocks (UFB). These function blocks are configured in the same way as programs and are displayed as function blocks. The resulting compressed display of programs permits the following:

- A clearer display of programs
- The creation of typical solutions
- A better and reduced documentation

For working with User Function Blocks a distinction is made between classes and instances. The UFB class determines the functionality and the appearance of a user function block and also includes the total program created by the user, including the functions, function blocks and variables, the faceplate and the parameter definition dialog box.

The user function block program is configured in the project tree under **User function block pool, P-FB**. Each UFB class receives a freely-assigned class name, under which it is called in other programs. The program can be configured using FBD, LD, IL or ST.

The faceplates are configured in a dedicated faceplate editor, corresponding to the graphics editor.

Only after the plausibility check is performed in the project tree, the user function blocks will be available under > **Function blocks** > **User function blocks**.

This menu selection is used to integrate user-defined function blocks into other programs which are created using the FBD, LD, IL or ST.

User function blocks can be locked by the password. Following this, they only still appear through their outer form. Embedded function blocks are invisible.

For more information on configuration of the user-defined function blocks, refer to ***Engineering Manual, IEC 61131-3 Programming***.

3.3.11 Common display pool P-CD

Displays and logs set in this project object must be available on all operator stations. The objects from the display pool are loaded only on those stations that have access to the process data required in these displays and logs. Users are able to assign the FGRs out of P-CD to specific Operation Station. Be noted that only 5 FGRs are supported when you assign FGRs to a lite Operator Station.

Any modifications to displays in the general display pool must be loaded separately into all the operator stations.

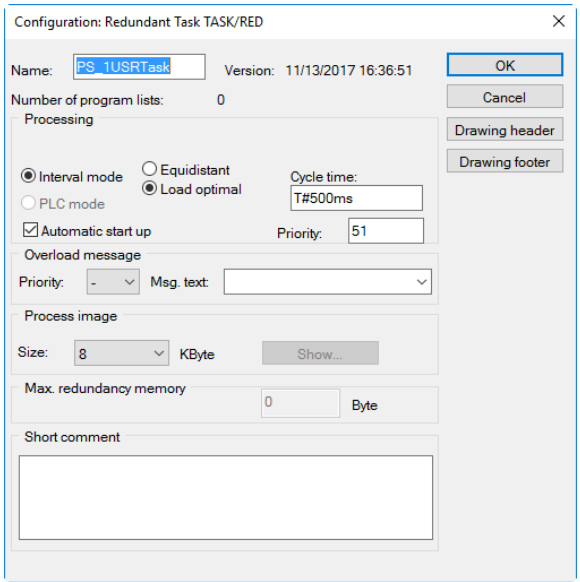
Once an object from the display pool is selected, select *Display Target Station* from the shortcut menu to show a list of operator stations that this program is loaded to.

3.3.12 Task TASK and redundant task TASK/RED

The task object comprises all the subordinate program lists and SFC programs. The tasks determine the processing speed of a resource. A distinction is made between user and system tasks.

In user tasks, the programs are processed cyclically; in system tasks, certain events in the resource or commissioner actions determine whether a certain system task is executed. The programs within a task are controlled by a program list or by an SFC program. The number of user tasks is limited to nine tasks per PS/controller (AC 900F/AC 800F/AC 700F/FieldController default setting– three tasks). A maximum of eight cyclic tasks and one default task can be configured. The default task is always executed if none of the other tasks is executed (either cyclically or once only).

Also refer to Section 5, Commissioning on page 215 and Loading the project on page 236.



Redundant Task.png

Name Maximum 12 characters

Version Date and time of object creation

Number of program lists
The number of assigned program lists is displayed.

Processing

Interval mode The task is performed in time intervals. Select between *Equidistant* and *Load optimal*.

Equidistant The task is triggered at regular intervals (for example, the first time the task is computed), a series of equidistant times is specified for triggering task computation.

Load optimal In contrast to equidistant triggering, the times at which the task computes are continually recalculated during execution. Each task computation decides the time of the next computation, as the configured interval time is added to the current system time. In

normal operation equidistant tasks do not differ from those which are load optimal in execution. The distinction only becomes relevant when brief load peaks in the system cause a task to be delayed in execution. Equidistant tasks will keep to the defined time scale, while load-optimal tasks stretch the intervals slightly in the event of overload, thus relieving the system.

Cycle time Input format: for example, T#4s500ms (=4.5s) d = days, h = hours, m = minutes, s = seconds and ms = milliseconds. The interval time must be at least 5 ms and a multiple of 5 ms.

PLC mode At most one user task can be made the **default task**, in that it computes in PLC mode. It has the lowest priority of all cyclic tasks, and starts itself again at once after a run. This leads to the fastest possible cyclic processing of the underlying programs. Because of its low priority (50) it can be interrupted by all other tasks. The default task is only in execution when no other task is ready to compute.

Automatic start up

☑ Defines whether the task will be started automatically after loading or after starting the resource, being transferred to the "running" status.

The default is automatic start.

Priority The priority (not to be confused with message priority) controls the processing of several tasks which are in the state "running" at this time. The multitasking of several tasks of the same interval time is controlled by the priority. The lowest priority that can be assigned for user tasks is 51; the highest is 99.

Overload message

A task triggers an overload message if it detects that an interval time is overshot. An interval time overshoot takes place if a task has not been executed within double its interval time.

Priority A message is output in case of overload. 1 to 4: A message with the relevant priority is output. Also refer to ***Freelance Operations, Operators Manual, Messages and Hints***.

Msg. text Maximum eight characters

Process image Refer to Section 5, Commissioning on page 215 and Process image on page 124.

Size Four and multiples of four, up to 32 KB

Show... Shows the process image of a task. Refer to Section 5, Commissioning on page 215 and Process image on page 124.

Maximum redundancy memory
Maximum size of memory required by the redundant task for the transfer of redundancy data. The redundancy memory is only displayed in redundant and plausible tasks.

Short comment Maximum 159 characters

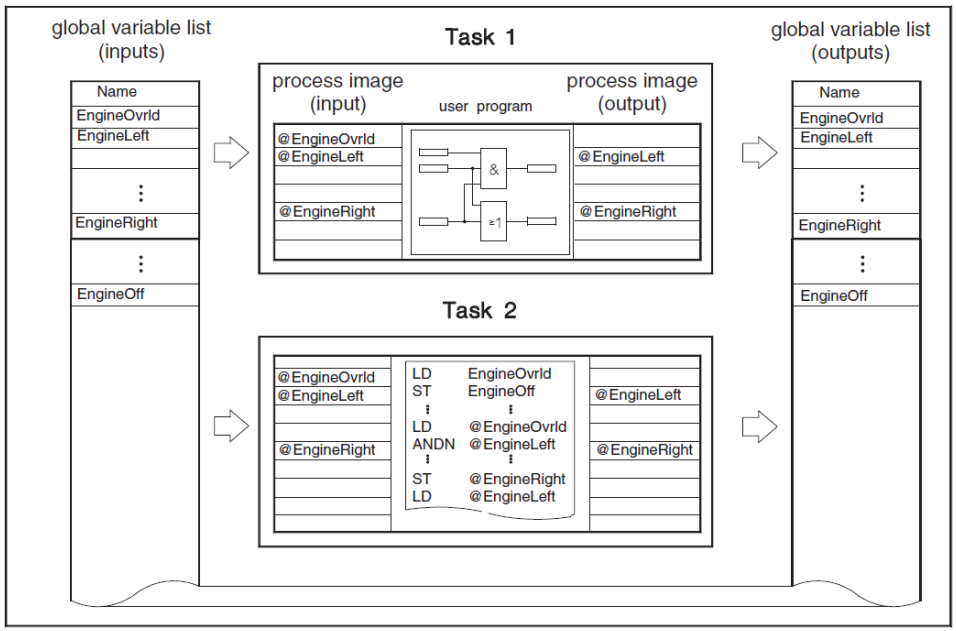
Drawing header and footer
Refer to Section 5, Commissioning on page 215 and Documentation on page 137.

Avoid the configuration *Equidistant* associated with very small task cycle times (5ms). This could cause a CPU overload.

Process image

The process image is an intermediate storage memory for task variables where the task variables are read before the task is run and where those variables are written back again after the task has finished. All variables written through the process image are identified by @ in the programs.

Access through process image (indirect)	@Engine_E1243_On
Access bypassing process image (direct)	Engine_E1243_On



di1504_us.png

The process image is read at the start of task calculation. Then the entire user program, the sum of all program lists and programs belonging to this task are calculated once. The current states and sizes of the task variables are then rewritten from the process image to the global variables. In the next interval or task run, the variables from the last task calculation are available. To achieve the shortest reaction times, provide indirect access through the process image. For this reason, select for each variable whether access is permitted through the process image.

Access through the process image increases clarity of configuration. This path should always be preferred instead of the configuration bypassing the process image.

In a redundant task the variables may only be accessed through the process image. Write accesses which are not made through the process image are not transferred to the secondary, so that the variable would change value in a toggle.

Variables that are written through a gateway must not be written through the process image as this state can be overwritten by the task.

Predefined system tasks

All system tasks are grouped in the project tree in *.SYSTask node. The predefined system tasks allow to start and run parts of user programs as a reaction to system status changes. The event which triggers the task (for example, a certain state transition of the resource or an error in the user program), is always defined. All predefined system tasks are created in the resource after initial program loading. After the resource is initialized, the system tasks are automatically started, but do not compute until the event assigned to them occurs.

The following tasks are predefined:

*.**ColdSt** [Task] [Once]

*.**WarmSt** [Task] [Once]

*.**Run** [Task] [Once]

*.**Stop** [Task] [Once]

*.**Error** [Task] [Once]

*.**LatCSnd**[Task] [Cyclic,T#1s]

*.**LatCRcv**[Task] [Cyclic,T#1s]

*.**RedSt** [Task/Red][Once] (only for redundant process stations)

No other user task is calculated during the computation of a once-only task. The maximum permitted computation time for this task (that is, the time from beginning of execution to end of execution) is limited to 10 seconds. Operations which take more time (for example, waiting for process data in a loop), must be configured in one of the cyclic user tasks.

For more information, refer to ***Engineering Manual, Process Station***.

Cold start task, ColdSt

The cold start task is executed after the resource performs a cold start. The task is executed on transition from **cold start** or **cold start stopped** to **running** state. During a cold start, all the variables and working data in the function blocks are initialized. The initial variable values can be specified singly in the variable list for each variable.

The I/O module and device outputs are set to safety values. After the computation of the cold start task, the outputs are rewritten depending on the programs.

Refer to ***Engineering Manual IEC 61131-3 Programming, Variables*** and ***Engineering Manual Process Station, Status Diagram of the Resource***.

Warm start task, WarmSt

The warm start task is executed after the Resource performs a warm start. The task is executed on transition from **warm start** or **warm start stopped to running** state. It is computed before the cyclic user tasks are started.

The process station always performs a warm start after each voltage outage as soon as the voltage is back. A cold start is forced depending on the duration of the voltage outage (function **trigger cold start**).

When a power-fail signal is received, the output modules first hold their value. If the voltage restabilizes after a specific period of time (15 ms), task processing resumes where it was interrupted. If not, a warm start is executed on the modules and the output will be configured safety values.

Also refer to ***Engineering Reference Manual Communication and Fieldbusses***, Section 5, Commissioning on page 215, Load whole station on page 237 and ***Engineering Manual, Process Station***.

All system variables are retained so that the functions and function blocks can continue operating with the same values as before the warm start.

In the function blocks, the inner states are retained and the function blocks resume after the warm start from where they were interrupted. In addition, special block-specific features are included in some function blocks.

- Blocks which use the task interval time. For example, controllers continue working smoothly.
- For blocks with monitoring times, the time of the voltage failure is counted. This means that, for example, a timer set to 30 minutes will run a total of 30 minutes even there is a power-fail of 5 minutes.
- The gap is marked in acquisition blocks such as trend or disturbance course log acquirers.
- Analog filters such as INTEG (I component) and PT1 components continue their calculation from where they were interrupted.

Refer to ***Engineering Reference Manual Functions and Function Blocks, Miscellaneous, Warm start behavior of the function blocks***.

The system variable "xxxPowerOffTim" contains the duration of last power failure which led to a warm start. It is counted from the time the power failure occurred to the restarting of the operating system.

For AC 800F and DCP the number of warm starts is counted in a system variable (xxxx.NoPowerFail) that is reset to zero in the event of a cold start.

Also refer to ***Engineering Manual, Process Station*** and ***Engineering Manual IEC 61131-3 Programming***.

Run task, Run

The Run task is executed after the resource changes state from *Stopped* to *Running*. All user tasks are only started after the run task ends. For more information, refer to ***Engineering Manual, Process Station***.

Stop task, Stop

The Stop task is executed after the D-PS resource changes state from **running** to **stopped**. The stop task is triggered only after all the other tasks have been stopped. A re-start of the resource is possible only after the stop task is executed completely. Also refer to ***Engineering Manual, Process Station***.

Error task, Error

The error task has the highest priority (100) and is used for handling errors in user programs caused by user programs. It is executed only if an error is detected in a user program. In case of unrecoverable errors, the error-producing task assumes the status **not executable**. In case of recoverable errors, the error-producing task can continue to run only when automatic error correction is activated for the DP-S resource. Due to its high priority, it cannot be interrupted by other tasks. Execution of the error task can be suppressed. For more information, refer to ***Engineering Manual, Process Station***.

Lateral communication send task (LatCSnd)

All variables are automatically transferred to other resources if they are released to be sent with the Export = Yes attribute in the corresponding (local) resource. The lateral communication required for sending is automatically executed by the **LatCSnd task** (Lateral Communication Send task).

Also refer to ***Engineering Manual, Process Station,*** Section 5, Commissioning on page 215 and ***Engineering Manual, IEC 61131-3 Programming***, Exported variables of the resource on page 229.

The total quantity of variables the resource can make externally visible is limited to 1400 bytes.

Receive task for lateral communication (LatCRcv)

All variables can be read automatically in other resources if they were released by the attribute Export = Yes in the resource to be sent. The lateral communication required to read and receive is automatically performed through the **LatCRcv-Task** (Lateral Communication Receive-Task).

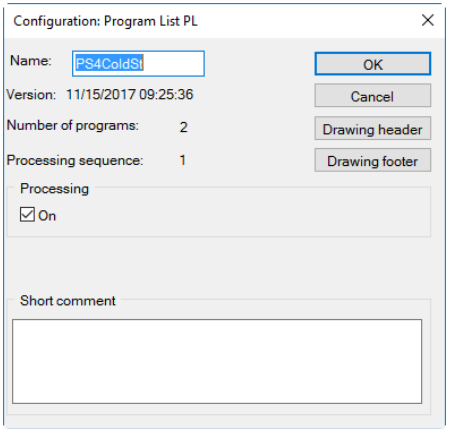
Also refer to Section 5, Commissioning on page 215 and ***Engineering Manual, IEC 61131-3 Programming***, Exported variables of the resource on page 229.

Redundancy start task, RedSt (for redundant resource only)

The RedSt task is executed directly after a redundancy toggle before the user task's first computation. It can cause execution of user programs which should react especially to the redundancy toggle.

3.3.13 Program list (PL)

The Program list object contains all subordinate programs and controls the processing sequence of these programs. The programs are processed depending on their consecutive number in the project tree. The higher-order task determines how fast the programs are processed on the CPU module of the process station. The programs are written in FBD, LD, IL or ST. Refer to Loading the project on page 236.



PL.png

Name Maximum 12 characters

Version Date and time of object creation

Number of programs
The number of the subordinate programs is displayed

Processing sequence
Indicates the position within the task at which this program list is executed

Drawing header and footer
Refer to Documentation on page 137 and Section 6, Documentation on page 259.

Processing

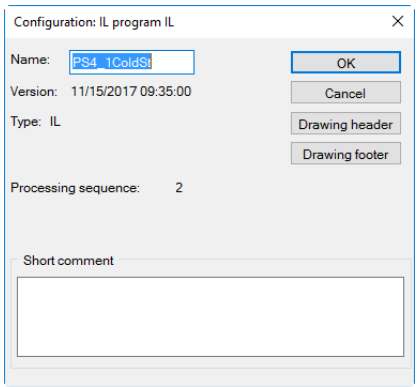
On ☑ The subordinate programs are processed (default setting).
☐ The subordinate programs are not processed.

Short comment Maximum 159 characters

3.3.14 Programs on the process station: SFC, IL, FBD, LD, ST

> Select object in Project tree > **Edit** > **Insert next level**

From the Object selection dialog box, select the required program.



IL.png

Name Maximum 12 characters

Version Date and time of object creation

Type Dependent on the project object selected:

SFC	Sequential Function Chart program
IL	IL program
FBD	FBD program
LD	LD program
ST	ST program

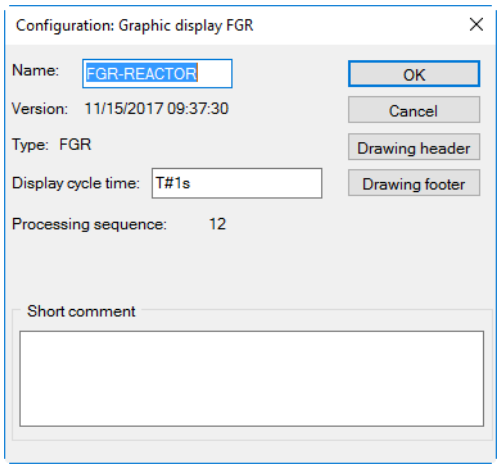
Processing sequence
Indicates the position within the program list at which this program is executed.

Short comment Maximum 159 characters

3.3.15 Programs on the operator station: Displays, reports, and logs

Displays, reports, and logs are the programs in an operator station. Like the programs on PS, the displays are updated cyclically.

> Select object in Project tree > **Project** > **Header...**



FGR.png

Name Maximum 12 characters

Version Date and time of object creation

Type Depends on the following project objects selected:

SFCP	SFC display
OPL	Operation log
FGR	Graphic display
GRP	Group display
TS_D-OS	Time scheduler display
DCL_D-OS	Disturbance course log
SSL1	Signal sequence log 1
SSLN	Signal sequence log N
TR_D-OS	Trend display
OVW	Overview display
REPORT	Excel reports
WEB	WEB display

Display cycle time
Time for repeated update of the values of a display or log.

Processing sequence
Indicates the node position of this project object relevant to the operator station.

Short comment Maximum 159 characters

3.3.16 Structure node (STRUCT)

Structural element of the operator station. To provide a clearer overview the displays, reports, and logs from the operator station can be grouped together with this project element.

The name must be no longer than 12 characters.

> Select operator station in Project tree > **Edit** > **Insert next level** > **Structure node** > **Specify name**.

3.3.17 External timer server

In Freelance system, all stations are time-synchronized by the first PS or by an electronic timer (radio controlled clock).

External stations that are connected by means of a router or modem can be time-synchronized by the operator station PC, a gateway PC, or a DCP gateway. For this purpose, the “Ext. Timer Server” function must be activated in the resource configuration (select > Operator station or Gateway in Project tree > **Project** > **Header**). By specifying the IP address, up to three stations can be time-synchronized by a resource.

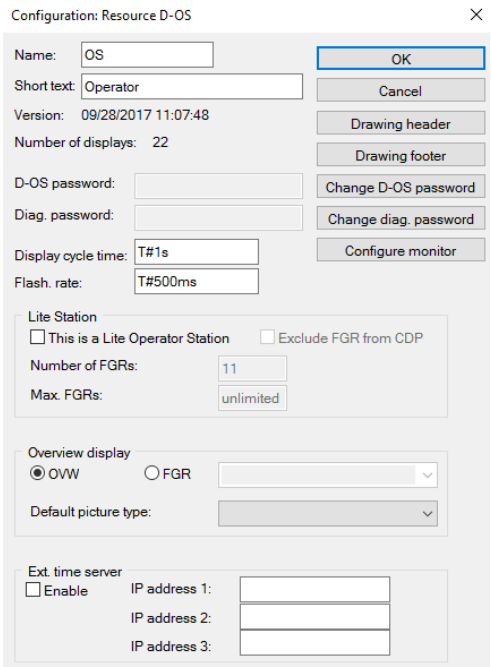
The following forms of time synchronization are possible:

Resource type	Can synchronize
Operator station (D_OS)	External PSs D_PS
OPC gateway	External PSs D_PS
UNI gateway	External PSs D_PS
DCP gateway	External DCP gateway stations and Maestro UX workstations

Inactivating the time synchronization

If several Freelance programs are running on a PC with activated external time synchronization, then time synchronization can be terminated only after all these programs are terminated.

Time synchronization is not complete if only one application is completed or if on-line mode is switched off.



TimeSync Interface.png

Ext. time server External stations with IP addresses 1, 2, or 3 are time-synchronized through the operator station.

Enable Enables on time synchronization

If the *Enable* flag is not set, then the time synchronization messages will not be generated.

IP address Enter the IP addresses of the stations to be time-synchronized by means of this function. *If the Onet* broadcast address is entered

when connecting to a Maestro UX system, then all the DCP gateways linked in this network will be synchronized.

3.3.18 Graphic macros pool (P-MAC)

Below this element all graphic macros are defined. A macro MAC can be created directly in the project tree or indirectly when generating a graphics image with the graphics editor.

The name must be no longer than 12 characters.

> Select **Soft (SW)** node in Project tree > **Edit** > **Insert next level** > **Graphic macro pool P-MAC** > **Specify name**

3.3.19 Macro structure nodes (STRUCT)

Structural element for grouping graphic macros.

The name must be no longer than 12 characters.

> Select **Graphic macro pool P-MAC** in Project tree > **Edit** > **Insert next level** > **Structure node STRUCT (Macro)** > **Specify name**

3.3.20 Graphic macros (MAK)

A MAK graphic macro can be created below the “graphic macro pool P-MAC” in the project tree or when generating a graphic display with the graphic editor.

The name must be no longer than 12 characters.

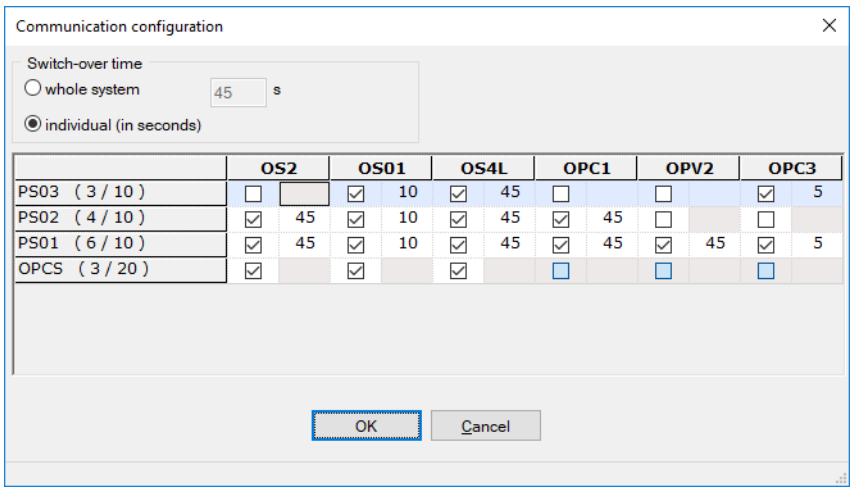
> Select **Graphic macro pool P-MAC** or structure below the macro pool in Project tree > **Edit** > **Insert next level** > **Graphic macro MAK** > **Specify name**.

3.3.21 Configuring the communication links

For data exchange with operator and gateway station master (10) connections are available in each process station. For data exchange between OPC servers (OPC-S) and operator stations 20 connections are available per OPC server. If more operator or gateway stations are configured, then the effective communication links of all the subscribers in a Freelance system are specified in the communication configuration dialog.

Double-click the project object Configuration (CONF), or through menu selection:

Select **System > Communication configuration...** will open the dialog box.



Communication Configuration_T_de.png

For more information refer to Configuration of the communication connections on page 189.

3.4 General functions of the project tree

3.4.1 Save project

> Project > Save Project

This function saves the project during configuration. The project is saved to the project file. For more information, refer to Saving a project on page 57.

3.4.2 Save tab

> **Project** > **Save Tab**

This function saves the editor during configuration. For more information, refer to Saving a tab on page 58.

3.4.3 Documentation

> Select level > **Project** > **Documentation**

The project documentation function is started. The documentation is opened for the selected objects. In the documentation settings, select only the selected objects or also the subsidiary objects to be documented.

For more information on compilation of documentation jobs and the scope of documentation, refer to Section 6, Documentation on page 259.

3.4.4 Check, Check all

If the plausibility check is initiated from the uppermost project node, the Freelance Engineering generates the current CSV file (required for the CBF Viewer) in the background. The time required to generate a CSV file may become relatively long with large projects, the CSV file generation is associated with the check of the uppermost project node. For example, if the CONF node is selected and start the plausibility check, no CSV file will be generated.

During commissioning, initiate several plausibility checks from the CONF node (or lower) and only occasionally, but at least once in the end, from the uppermost project node. The generated CSV file is then loaded to the Freelance Operations PC with the download to the Freelance Operations station.

Check all is used to check the selected object and all objects positioned below it for correctness (or plausibility). **Check** is used to check the selected objects and the subordinate incorrect objects. Program code is generated automatically for all correct objects, and this can be loaded to the relevant stations during the subsequent commissioning phase. Incorrect objects are shown accordingly in the color set.

In case of long error lists, perform plausibility checking in smaller sections. Each node of the project tree can be checked separately. Individual blocks within a program can also be checked.

> Select level > **Project** > **Check**

The selected object and the subordinate incorrect objects are checked for plausibility (for errors, missing entries, or contradictions). The errors are listed in a dialog and must then be eliminated.

> Select project object > **Project** > **Check all**

The selected object and all the subordinate objects are checked for plausibility. If the errors appear in an error list, then they must be eliminated. The plausibility check can take a few minutes.

Direct access to error points after plausibility check

After a plausibility check (**Check** or **Check all**), the error list is newly created in a list box.

- Double-click on a marked message or the **Current error** and **Next error** buttons triggers an access to the object causing the error.
- A marked object is identified through the “system selection” color (light blue) or being framed with a border.
- In the editor, the marked object is positioned in the middle of the screen.

Access to the error object is same whether the plausibility check was opened in the project tree or the editor. If you access an editor page which previously contained a selection (only possible after a plausibility check within an editor), this selection is lost by the plausibility check jump.

Classes of plausibility errors

Three classes of plausibility errors are distinguished:

Error

- Some problem will occur.
- The project cannot be loaded in this state.
- The error must be corrected.

Warning

- The configuration is inconsistent, but the project can be loaded.
- The person responsible for configuration must decide whether the error needs to be rectified or whether the project can run effectively despite this message.

Hint

- Contains information for the user which does not necessarily need to be followed.
- No action is required on the configuration process. Provides an information that certain inaccuracies are identified by the system or the minor adjustments must be performed.

Structure of the plausibility check messages

The messages are formed as follows:

<Number> <Label for class of error> <Message text>

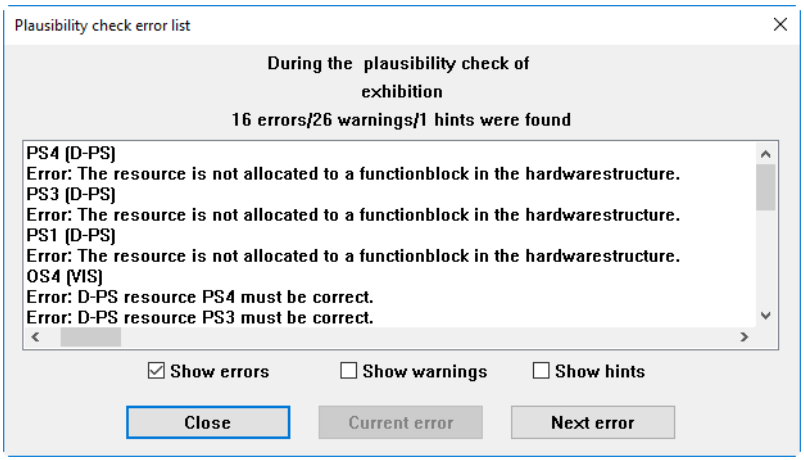
The following arrangements apply to the various different error classes:

[xxxx.2.xxxxx] Error: < Message text >

[xxxx.3.xxxxx] Warning: < Message text >

[xxxx.4.xxxxx] Hint: < Message text >

Error list display



Plausibiity Check Error List.pn

- The classification of an error is recognized both from its number and the introductory text.
- The different message classes can be selectively faded in or out.
- For the whole time the project is being processed the most recent setting of the check boxes is preserved.

3.4.5 Show error list

> Select level > **Project** > **Show error list**

The error list contains all the errors in the selected objects occurring after a plausibility check. The list can be printed out through the Documentation function. For more information, refer to Section 6, Documentation on page 259.

3.4.6 Object header

The general parameters of an object can be modified at any time. It is the same as specified when the object was added. Select an object:

Double-click to select some objects such as resource, task or program list. Access the editor or extended parameter-definition dialog directly on other objects such as FBD program, graphic display or trend display.

> Project > Header...

For more information, refer to Configuring the project objects on page 92.

3.4.7 Comments on a project object

> Project > Comment

Use the comment editor to create or modify a free text for each object. This multi-page text can also be used to describe the program or the process technology and can be output together with the object using the Documentation function. For more information, refer to Editing a project comment on page 59.

3.4.8 Network

The resources are assigned to the communication addresses (resource-ID's) through **Network setting**. The stations IP addresses or host names which the resources are assigned are also specified in the network setting.

> Project > Network...

For more information, refer to Network configuration on page 200.

3.5 Project tree options

3.5.1Print

> Options > Print

Prints the screen contents.

3.5.2 Long or short form in the status bar

> Options > Long state line

Long format — For example, the project object **Reactor** is selected. The Status bar then reads **docu_vis/conf/V_GR/Reactor**, that is the names of the project objects are displayed.

Short format — For the same example, the short format will display **C01/R08/B04**.

Abbreviations:

C	Configuration level
R	Resource
T	Task
P	Program list
B	Program, Display
A	Sequential function chart
S	Step
T	Transition
S	Structure node

The **short form** is the default setting.

3.5.3 Colors in the project tree

> Options > Color setting...

The colors of nodes, connecting lines, mandatory parameters and fields can be modified in the drawing footer. If required, the default settings for the colors can be restored. For more information, refer to Color settings of nodes on page 79.

3.5.4 Lock/Unlock user function block

> Options > Lock/Unlock UFB-Class

A user function block class (UFB-CLASS) can be locked or unlocked. For more information, refer to ***Engineering Manual IEC 61131-3 Programming, User-Defined Function Blocks***.

3.5.5 Lock/Unlock tag type

> Options > Lock/Unlock Tag Type

A tag type can be locked or unlocked. For more information on tag types, refer to ***Engineering Manual IEC 61131-3 Programming***.

3.5.6 Initial filter for variable and tag list

> Options > No initial variable/tag-filter

This option is used to define the handling of filter criteria when opening or closing the variable and tag list:

Menu alternative activated

No filters are active when opening the list, the whole list will be shown.

Menu alternative NOT activated

The filter settings in the variable and tag list are saved when closing the list and when opening the list again they are executed.

3.5.7 Auto Accept

> Options > Auto Accept

Or

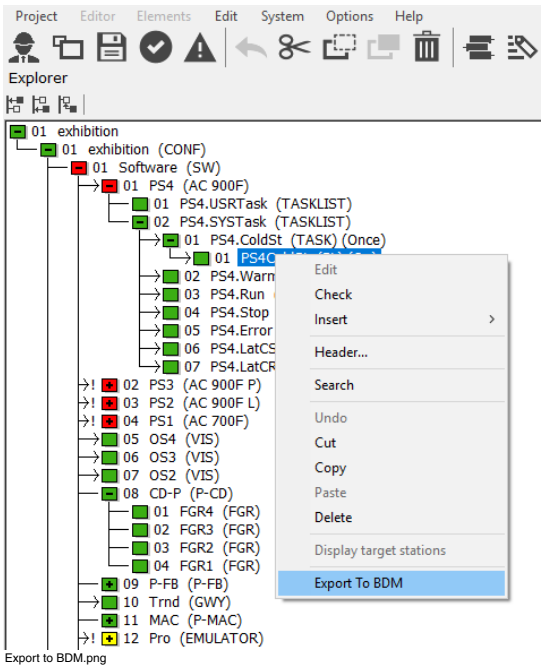
Double-click **AUTOACCEPT ON/OFF** in the Status bar to enable or disable the Auto accept option

This option is used to suppress the Save dialog box when leaving an editor tab. For more information, refer to Saving a tab on page 58.

3.5.8 Export to BDM

Certain project tree objects can be selected and exported to the Bulk Data Manager tool using *Export to BDM*.

> Select project object > right-click > **Export To BDM**



Export to BDM.png

Freelance BDM is a offline bulk engineering tool that allows manipulation of small or large application entities during engineering with Freelance Engineering through Microsoft Excel. Key functionality comprises Import/Export of tag and variable lists, replication of program typicals, function block parameter configuration and so on.The Commissioning mode will be disabled if the system detects the running of BDM.

The Export To BDM context menu item is enabled only for the project tree objects that are supported. For more information, refer to ***Engineering Manual Bulk Data Manager***.

4 Hardware structure

4.1 General description – Hardware structure

Within the hardware structure the required resources defined in the project tree are allocated to the hardware. A system consists of the process, operator and Gateway stations. These stations are allocated to certain resources according to IEC 61131-3. The resources serve as structuring elements in the project tree for allocation of the application program parts and displays to the required hardware.

The D-PS resources which are configured in the project tree are assigned to the process stations in the hardware structure. A process station can have a rack-based setup, that is, conventional with input and output modules assembled in module assembly frames for process control, or by the AC 900F/AC 800F/AC 700F controller through one or more field buses with intelligent field devices and/or remote I/O.

In the conventional **Process station**, the I/O modules are mounted in module supports (racks). The CPU module processes all programs of this resource and is simultaneously the communication module to the system and the I/O modules. The **Process station** consists of the central unit and a maximum of four I/O units. The central unit is always the station with the CPU module. The I/O units do not need an own CPU module: they can accept other I/O modules so that their numbers can be expanded (from eight for the central unit) to a maximum of 44 for a process station. All units require a link module for feeding in the power supply and an identical rack for holding the modules.

For a direct equipment with various different fieldbus modules, there is currently the AC 900F, AC 900F Lite and AC 800F controller, which are capable of redundancy and non-redundancy and the AC 700F Controller.

The **AC 900F process station** is available in two variants, namely Standard and Lite version.
The AC 900F consists of the following components:

- CPU Module
- I/O Bus connection (up to 10 direct I/O modules)
- A maximum of two fieldbus interface modules (PROFIBUS and/or CAN for I/O rack)
- AC 900F: four onboard Ethernet interfaces
- AC 900F Lite: three onboard Ethernet interfaces
- Three Serial interfaces (two for Modbus and Telecontrol and one Diagnostic)

Depending on the type and quantity of process signals, up to ten plug-in type input/output modules can be directly connected. Additionally, two extensive PROFIBUS lines can be connected via two PROFIBUS modules. The CAN module allows connection of a maximum of 5 rack I/O units and thus the connection of 45 I/O modules as used in the rack-based Freelance process stations.

The **AC 700F process station** is a low-cost solution for small applications. The basic features of an AC 700F controller are:

- CPU module PM 783F with
- up to eight directly connected I/O modules with I/O terminal units,
- one fieldbus communication interface.

The **AC 800F process station** consists of the case and the main board. Together they form a unit which can be equipped with various modules. The module is required for the power supply. An Ethernet module is connected to the Control Net. Both modules are available in various designs.

The controllers are equipped with up to four fieldbus modules. There are various versions of the fieldbus module for PROFIBUS DP, Serial (for example, Modbus), CAN and Ethernet (for example, FIELDBUS FOUNDATION HSE).

The CAN module allows the connection of a maximum of five I/O units and the connection of 45 I/O modules in the way in which they are also used in the conventional process station.

Each PROFIBUS module allows the connection of a PROFIBUS. Each of these slaves can also be modular.

The serial module has two interfaces which can be occupied at option with the Modbus master interface protocol, the Modbus slave interface protocol, the telecontrol interface protocol, the Protronic interface protocol or the Sartorius scale interface protocol.

The Fieldbus Ethernet module have a 100Base T interface and can currently be operated using the FOUNDATION Fieldbus HSE and TCP/IP send and receive (subprotocol UDP) protocols. The list of available protocols is constantly being expanded.

For further details, see ***Engineering Manual Process Stations***.

In the Freelance system, there are up to 100 operator stations and process stations that can be configured. This theoretical limit, however, will not be able to be reached in practice, as the project size is kept earlier within limits due to the bus load and the maximum project file size of two gigabyte. Also the fact that Freelance Engineering does not support a multi-user configuration, practically does not allow 100 operator stations and process stations to be configured in one project. Besides, when allocating process stations to operator stations it has to be taken into consideration that a maximum of 10 operator stations (Clients) can be allocated to one process station (Server). In other words, if there are more than 10 operator stations in Freelance Operations, a restriction has to be configured in such a way that a maximum of 10 Freelance Operations operator stations may have access to one process station. With that, Gateways and OPC servers count as a Freelance Operations operator station.

However, it is possible to divide large plants in several smaller projects and then work in parallel with several Freelance Engineering working stations in different projects. The required lateral communication between the projects can be designed with TCP/IP send and receive blocks. The use of Freelance Operations for divided projects is not recommendable. If it is required, then at least from one operator station all the process stations can be observed. Instead System 800xA Operation must be applied.

The **Operator stations** are commercial PCs in which the software program Freelance Operations is installed for process visualization. All displays and records are configured with the software program Freelance Engineering and loaded in the operator station. Data acquisition for recording and trend display takes place in the process station, and all display and archiving in the operator station. Each process station can communicate with up to 10 operator and/or gateway stations.

D-GS resource stands for a **Gateway** connecting the system to other systems. Currently there are gateways for any **OPC Clients** and for individually created applications (**DMS API**).

D-ES resource is used for an **Engineering station**, that is the configuration tool Freelance Engineering. It appears in the hardware structure. This display is used exclusively for documentation purposes.

Each object can subsequently be parameterized. In addition to the object name given by default setting, a short text and a long text can be allocated.

There are no switches on the I/O module: all module settings can be configured in the hardware manager. To each channel of an output module, a safety value is allocated; this is the output in case of a CPU module power fail. The update times can be set individually. They depend, on the module type (analog or digital), the transmission rate set on the process station and the number of units connected to the central unit.

The hardware can be configured in two different view areas. In the tree view area the complete system can be configured in a tree structure. In the graphic view area the hardware can be configured in several display levels:

The **system view** shows the complete hardware structure. In the system view the individual stations like the process station, AC 900F/AC 800F/AC 700F controller, or operator station are activated and allocated to the project tree resources.

The **station view** shows the complement of a process station or an AC 900F/AC 800F/AC 700F. In the station view slots are allocated to the rack modules or AC 900F/AC 800F/AC 700F and FieldController modules.

The **detail view** shows more information on the rack modules of a process station or the modules of the AC 900F/AC 800F/AC 700F or FieldController so that they can be identified more easily. The detail view of the PROFIBUS master shows a bus line with the slaves configured on the bus. The slaves are represented using bitmaps

which are specified in the device database file. If no bitmap files are specified in the device database file of the respective vendor, standard bitmaps are displayed. However, it is possible subsequently to assign vendor-specific bitmaps.

The detail view of the PROFIBUS slave shows a device view in the form of an individual bitmap file which can be assigned in the parameter dialog of the slave.

The variables which are to be linked with the process through the I/O modules are entered in the I/O editor. A number of I/O components is provided by each module or slave according to the number of channels. Through these I/O components it is possible to directly use the information in programs and graphic displays. If the I/O component is also to appear in the variable list, a variable name must be indicated. The I/O component name is composed of the object name (16 characters) and the component name (16 characters), that results a total of 32 characters.

Diagnosis components are available in addition to the I/O components.

Through the **network configuration**, communication addresses, so-called resource IDs (previously station numbers), are assigned to the resources. IP addresses of the units assigned to the resources are also indicated in the network configuration (see Network configuration on page 200). When the project objects are loaded into the stations, the corresponding application program parts (as allocated in the project tree to the individual resources) are loaded into the resources. The resources or individual parts of the application program are loaded with Freelance Engineering in commissioning mode.

Below the tree view, it is possible to superimpose another tree from which templates which have already been created can be transferred into the project. The templates are already configured hardware objects, the specific parameter settings of which are to be reused. An application might, for example, involve preserving an already created slave configuration together with the parameter and I/O configuration created for certain devices, or pertaining to all slave modules, in order that the dialog created, based on the information from the device database file, can be used again and again.

The Windows operating system allows several resources to run simultaneously on one PC. Hence, parallel Freelance Operations and other operation on one and the same unit is possible. These two resources have the same IP address in network setting but they can still be addressed because of their different resource IDs. From the point of view of the system, they do not differ from two resources running on

different PCs. Even if several resources have been assigned to the same physical unit, each resource has an individual representation in the hardware structure.

As well as specifying the communication connection, the configuration of the hardware also serves the purpose of documentation and plausibility checking. Thus a comparison can be made between what was actually mounted and what was configured. Incorrectly inserted or missing objects are marked in different colors in Commissioning Mode of the hardware structure.

4.2 Hardware structure user interface

4.2.1 Tree view

The hardware structure tree view shows all hardware structure objects. Beginning with the system object, other hierarchical levels can be entered down to the device level.

The stations are entered through a position number showing the respective station in the graphic view display. Position query is made every time a station or a gateway is inserted. Each object features object parameters such as name, short text and long text. The parameters relating to the objects are described in the Engineering manuals of the process stations. Refer to ***Engineering Manual Process Stations***. Individual object names are allocated automatically. Names can be subsequently configured.

Abbreviation	Object description
AC 900F	AC 900F process station
AC 900FR	AC 900F redundant process station
AC 900F L	AC 900F Lite process station
AC 900FR L	AC 900F Lite redundant process station
AC 900F P	AC 900F Plus process station
AC 900FR P	AC 900F redundant Plus process station
AC 700F	AC 700F process station
AC 800F	AC 800F process station

Abbreviation	Object description
AC 800FR	AC 800F redundant process station
AI 723F	Analog/Digital Input 16x
AI 731F	Analog/Digital Input 8x
AM 801F	Battery Backup Module (PM 802F only)
AM 811F	Battery Backup Module (PM 802F/PM 803F)
AO 723F	Analog Output 16x
AX 721F	Analog/Digital Input 4x, Analog Output 4x
AX 722F	Analog/Digital Input 8x, Analog Output 8x
CI 773F	PROFIBUS module
CI 910F	CAN module for Rack-I/O
CI 930F	PROFIBUS module, suitable for redundant configurations
CM 772F	PROFIBUS module
DA 701F	Digital Input 16x, Digital Input/Output 8x, Analog Input 4x, Analog Output 2x
DAI01	Analog Input 16 x 0/4...20 mA, Ri=50 Ohm
DAI02	Analog Input 16 x 0...10 V DC
DAI03	Analog Input 16 x 0/4...20 mA, Ri=250 Ohm
DAI04	Analog Input 8 x PT100/mV
DAI05	Analog Input 16 x 0/4...20 mA, MU infeed
DAO01	Analog Output 16 x 0/4...20 mA
DAO02	Analog Output 16 x 0/4...20 mA
DC 732F	Digital Input 16x, Digital In-/Output 16x
DCP02GWY	DCP Gateway CPU for Maestro UX Gateway
DCP10GWY	DCP Gateway CPU for Maestro UX Gateway

Abbreviation	Object description
DCO01	Communication Module 4 x RS 485/RS 422 or RS 232 C
DDI01	Digital Input 32 x 24 V DC
DDI02	Digital Input 16 x 24...60 V AC/DC
DDI03	Digital Input 16 x 90...230 V AC
DDI04	Digital Input 28 x 24 V DC NAMUR initiators
DDI05	Digital Input 32 x 115/230 V AC
DDO01	Digital Output 32 x 24 V DC, 0.5 A
DDO02	Digital Output 16 x 24...230 V AC/DC
DDO03	Digital Output 16 x 24...60 V AC/DC, read back
DDO04	Digital Output 16 115...230 V AC, read back
DFI01	Frequency Input 4 x f <= 45 kHz
DI 724F	Digital Input 32x
DLM01	Link Module for Supply
DLM02	Link Module for Redundant Supply
DNETP	CAN master
DX 722F	Digital Input 8x (24 V DC), Digital Output 8x (Relay)
DX 731F	Digital Input 8x (115 V-230 V AC), Digital Output 4x (Relay)
EMULATOR	Controller Emulator
EI 801F	Ethernet module 10Base2 (Battery Backup PM 802F only)
EI 802F	Ethernet module AUI (Battery Backup PM 802F only)
EI 803F	Ethernet module 10BaseT (Battery Backup PM 802F only)
EI 811F	Ethernet module 10Base2 (Battery Backup PM 802F/PM 803F)

Abbreviation	Object description
EI 812F	Ethernet module AUI (Battery Backup PM 802F/PM 803F)
EI 813F	Ethernet Module 10BaseT (Battery Backup PM 802F/PM 803F)
EI 801FR	Ethernet Module 10Base2 Redundant (Battery Backup PM 802F only)
EI 802FR	Ethernet Module AUI Redundant (Battery Backup PM 802F only)
EI 803FR	Ethernet Module 10BaseT Redundant (Battery Backup PM 802F only)
EI 811FR	Ethernet Module 10Base2 Redundant (Battery Backup PM 802F/PM 803F)
EI 812FR	Ethernet Module AUI Redundant (Battery Backup PM 802F/PM 803F)
EI 813FR	Ethernet Module 10BaseT Redundant (Battery Backup PM 802F/PM 803F)
FI 810F	CAN/3 Module
FI 820F	Serial Module
FI 830F	PROFIBUS Module
FI 840F	FF/HSE Module 10BaseTX
FI 810FR	CAN/3 Redundant Module
FI 820FR	Redundant Serial Module
FI 830FR	Redundant PROFIBUS Module
FI 840FR	100BaseTX Redundant FF/HSE Module
FLRC_MSL	Freelance Remote Control/Modbus slave (Protronic)
FWK_DEV	IEC 870-5 Telecontrol Device
FWK_DEV_M	IEC 870-5 Master Telecontrol Device

Abbreviation	Object description
FWK_DEV_SL	IEC 870-5 Slave Telecontrol Device
FWK_DEV_TCP	IEC 870-5 Ethernet Telecontrol Device
GWY	Gateway
Hart	Hart-Device
HWSYS	System object
HWSTR	Structural element within the template manager
MODM_DEV	Serial Modbus Master Module
MODS_DEV	Serial Modbus Slave Module
MODTCP_M	Ethernet Modbus Master Module
MODTCP_S	Ethernet Modbus Slave Module
PM 783F	CPU Module
PROFI_M_772	PROFIBUS Master
PROFI_M_C	PROFIBUS Master
PROFI_M_DEV	PROFIBUS Master
PROFI_S_DEV	PROFIBUS Slave
PS	Process station
PSR	Redundant Process station
SA 801F	Power Supply 115 V AC/230 V AC (26W, restricted use with PM 803F)
SA 811F	Power Supply 115 V AC/230 V AC (34W, PM 802F/PM 803F)
SA 801FR	Power Supply 115 V AC/230 V AC Redundant (26W, restricted use with PM 803F)
SA 811FR	Power Supply 115 V AC/230 V AC Redundant (34W, PM 802F/PM 803F)
SCALE_DEV	Sartorius Scale

Abbreviation	Object description
SD 802F	Power Supply 24 V DC (26W, restricted use with PM 803F)
SD 812F	Power Supply 24 V DC (34W, PM 802F/PM 803F)
SD 802FR	Power Supply 24 V DC Redundant (26W, restricted use with PM 803F)
SD 812FR	Power Supply 24 V DC Redundant (34W, PM 802F/PM 803F)
SR_SRTCP	Send and Receive, TCP interface
SR_RNDEV	Ethernet Receive Module
SR_SNDEV	Ethernet Send module
SR_SRTCP	Ethernet TCP Send and Receive module
VIS	Operator Station

4.2.2 Graphic view

The hardware structure graphic view features several views with different information within their detailed levels.

System view — Rapid overview over the global structure. For example, number of operator and process stations, number of AC 900F, AC 800F, AC 700F and number of gateways.

Station view — Immediate information equipped/free slots and plugged-in module types. Intuitive equipment is by double-click.

Detail view — The detail view differs depending on the object selected and shows a detailed information of the object. The information of the object can be the module type (status information only online), the slaves connected, their vendor, model name and bus address (status

information and diagnostic information only online) or only the slave connected in its device view with some information such as vendor, model name and bus address (status information and diagnostic information only online).

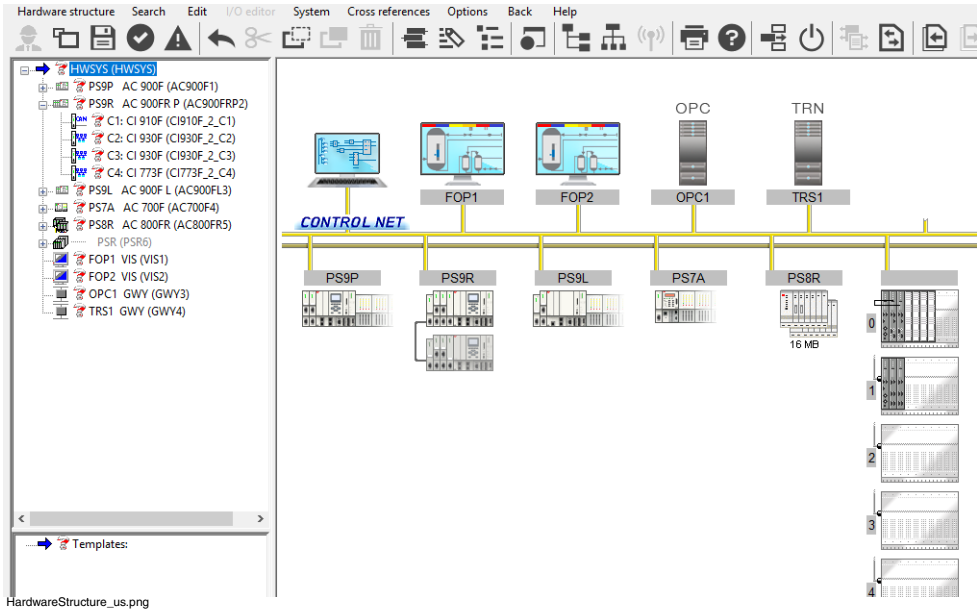
System view

The system view shows the graphic view of the hardware structure. A system consists of as many process and operator stations as required. The process stations consist of a central unit and up to four I/O units. For the controller, only the basic unit without the field busses or connected I/O units is shown in the system view.

Two printers can be connected to each operator station. The engineering station must not be separately activated and is only used to show the network address.

Definite positions (1 to 100) are allocated to relevant stations either directly by selection (double-click) of a definite position in the graphic view or by indication by means of a dialog during insertion in the tree view.

> **System** > **Hardware structure** > select **system object (HWSYS)**



HardwareStructure_us.png

The Process station objects can be moved from one slot to the other by using drag and drop. During this process the controller looses its connection with the resource allocated to it. The Process station object that is moved will retain its old tag name.

Station view of the Process station as Rack system

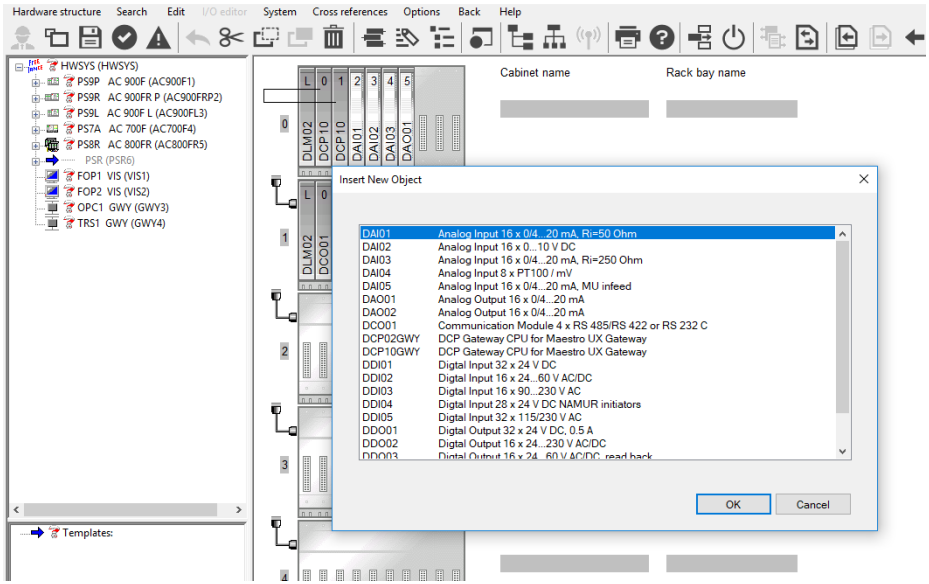
The station view of a process station as a rack system is identical to the view of the I/O units below a CAN module of an AC 800F or AC 900F.

In the station view modules can be equipped by double-click; doing this, only those modules are offered in the relevant selection list which actually can be equipped.

The station view of the process station shows the central unit with the CPU module and the I/O units. If other I/O units are activated, the station bus is shown connected between the stations. For each rack, a container name and a rack bay name can be allocated which can then be used as location indicator in the project documentation.

> Select a station object (PS) in the tree view or system view

> Select a CAN master (DNETP) in the tree view or module view for the CAN module (FI 810F / CI 910F)



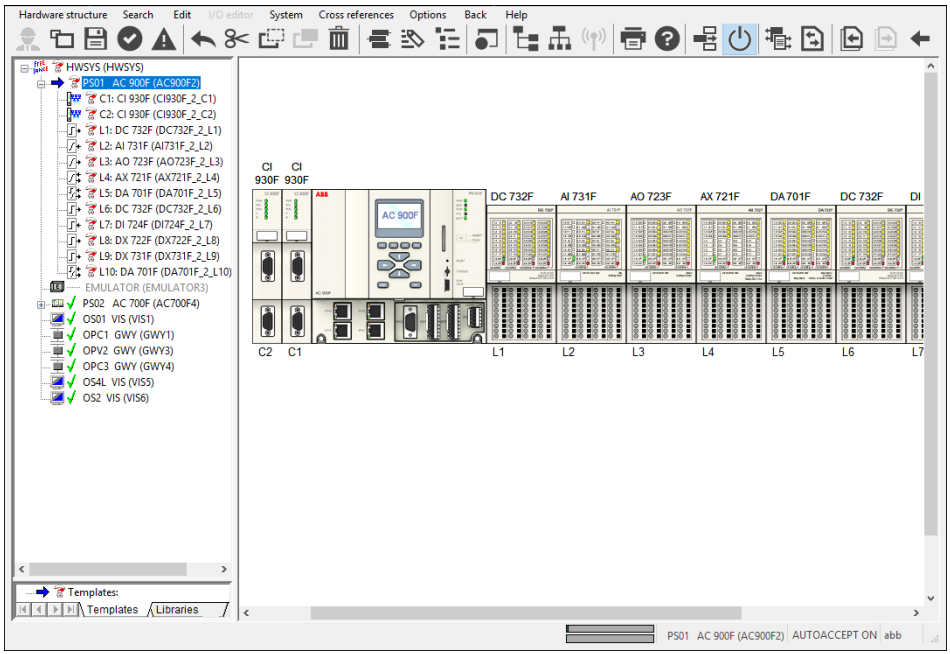
di0931_us.png

Station view of the Process station with AC 900F / AC 900F Lite

In the station view, the modules can be equipped with a double-click. Only those modules which can actually be equipped are offered in the corresponding selection list.

The station view of the AC 900F shows the basic unit with integrated CPU, the COM slot C1 and C2 for the Fieldbus modules, four Ethernet Interfaces, two Serial Interfaces and 10 I/O slots.

Select AC 900F controller object in the tree view or system view



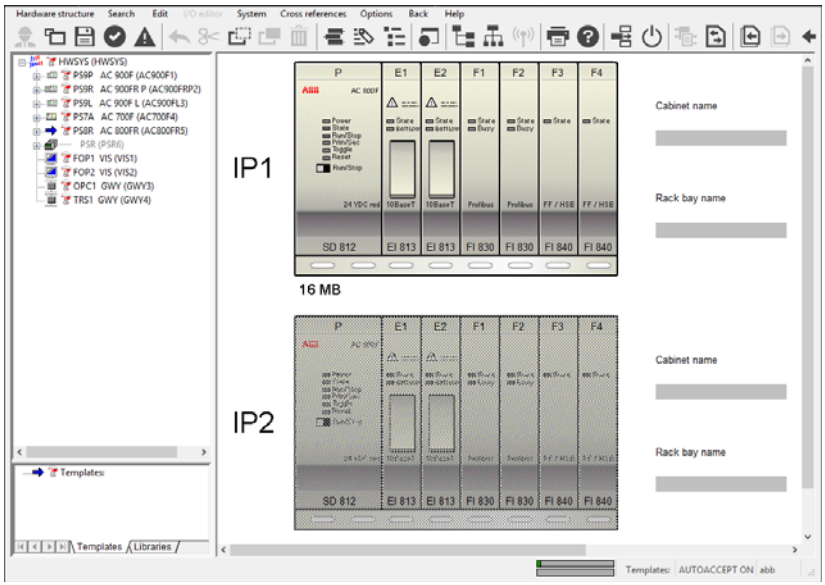
AC 900F_us.png

Station view of the Process station with AC 800F

In the station view, the modules can be equipped with a double-click. Only those modules which can actually be equipped are offered in the corresponding selection list.

The station view of the AC 800F shows the basic unit with integrated CPU, the slot P for the power supply module, slots E1 and E2 for the Ethernet modules and slots F1 to F4 for the Fieldbus modules.

> Select a Controller object (AC 800F) in the tree view or system view



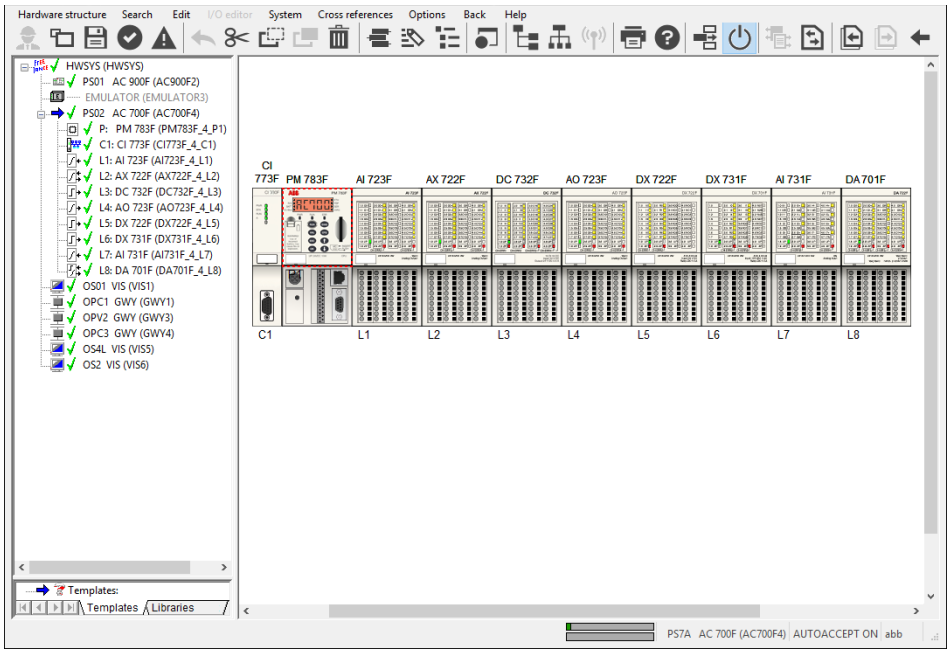
di0932_us.png

Station view of the Process station with AC 700F

In the station view, the modules can be equipped with a double-click. only those modules which can actually be equipped are offered in the corresponding selection list.

The station view of the AC 700F shows the CPU, the COM slot C1 for the Fieldbus modules, one Ethernet Interface, one Serial Interface and 8 I/O slots.

> Select a Controller object (AC 700F) in the tree view or system view



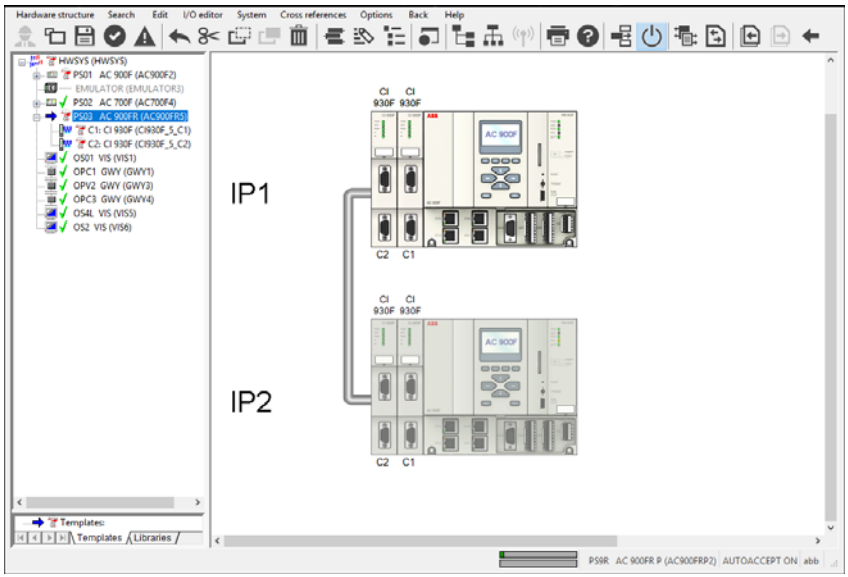
AC 700F_us.png

Station display of a Process station with redundant AC 900F / AC 900F Lite

In the station display of the redundant AC 900FR controller, both modules can be assembled with a double-click. The assembly of the modules is possible only in the upper base unit, in the redundant unit below it the same modules are automatically inserted. The modules for the redundant AC 900FR are set in the same way as those for the non-redundant AC 900F.

The station view of the AC 900FR controller shows two basic units, from which only the upper unit can be equipped. Apart from that, the display is identical with non-redundant AC 900F controller, with the COM slots C1 and C2 for the Fieldbus modules, four Ethernet Interfaces, and two Serial Interfaces.

Select AC 900FR from the tree view or system view



AC 900FR_us.png

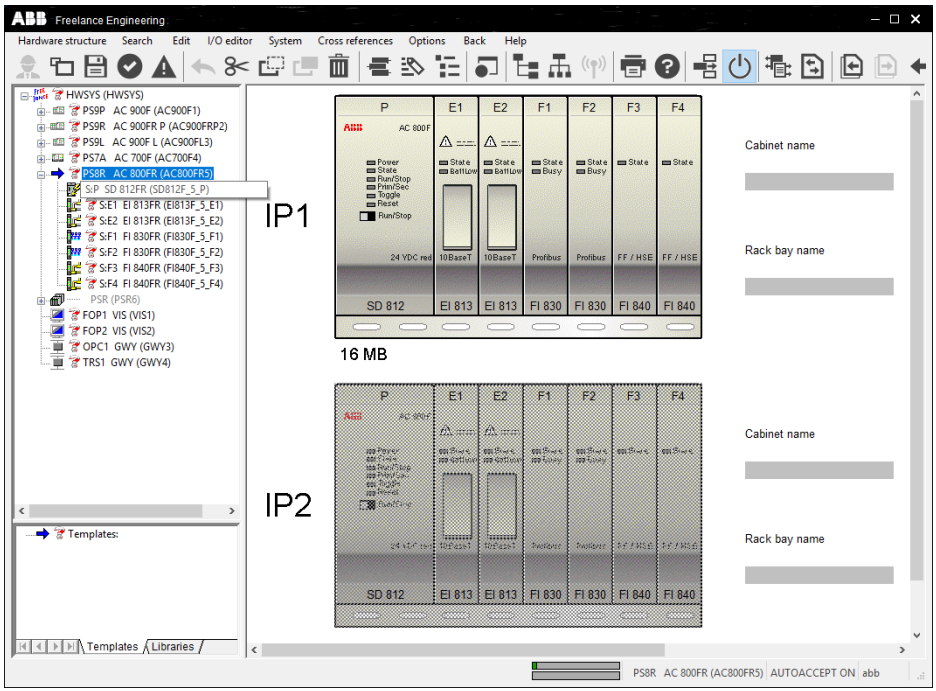
Station display of a Process station with redundant AC 800F

In the station display of the redundant AC 800F controller, both modules can be assembled with a double-click. The assembly of the modules is possible only in the upper base unit. In the selection list of the redundant AC 800F, only the redundancy objects are shown, designated by an “R” at the end, for example, FI 830FR. The modules for the redundant AC 800F are set in the same way as those for the non-redundant AC 800F.

The objects in the hardware structure are made for redundancy, enabling the configuration of both modules at the same time.

The station view of the AC 800F controller shows two basic units, from which only the upper one can be equipped. Apart from that, the display is identical with the non-redundant AC 800F controller, with the slots P for the power supply module, E1 and E2 for the Ethernet modules, and F1 to F4 for the field bus modules.

> Select an AC 800FR from the tree view or system view

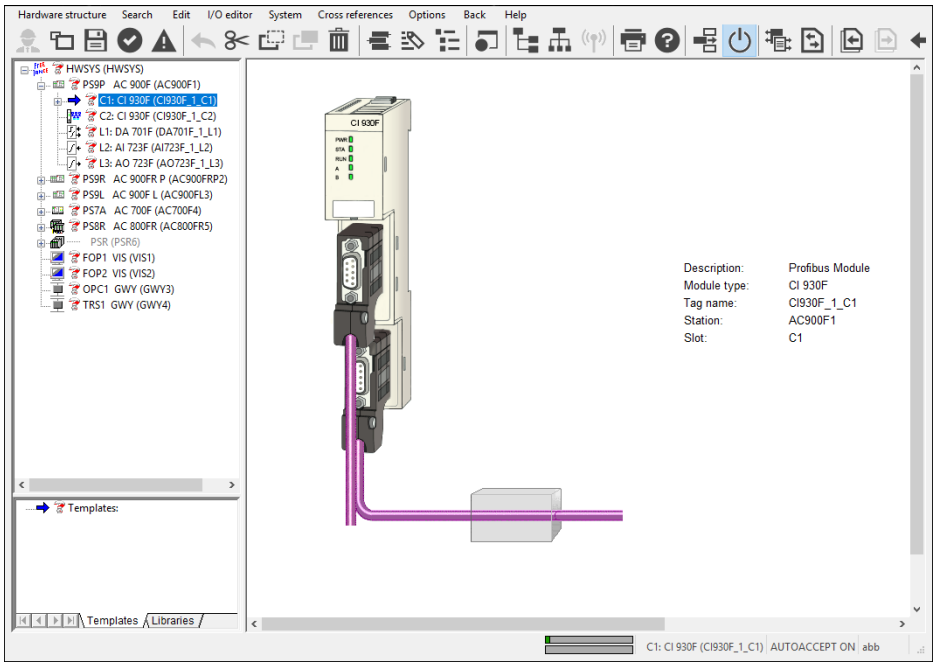


di0933_us.png

Detail view of a module

The detail view shows a stylized and detailed information on a module or an AC 900F/AC 800F/AC 700F module.

> Select a module in the tree view or station view

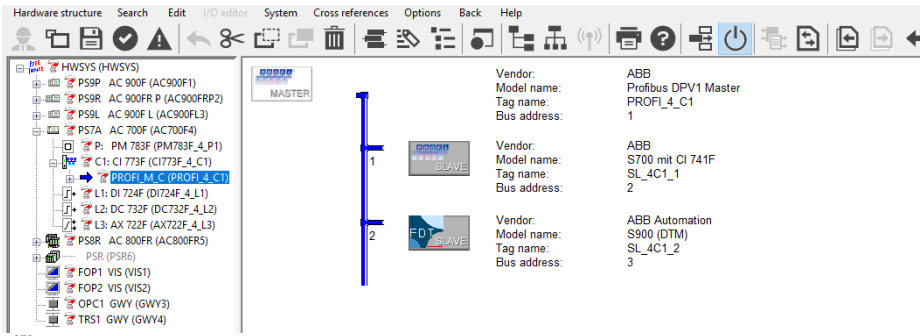


te072_us.png

Detail view of the PROFIBUS master

In the PROFIBUS view, the slaves can be equipped with a double-click. The required view position and bus address must be defined. In the corresponding selection list, only the free positions and bus addresses are available, that is, only those which can actually be equipped. Then a dialog is started in which a slave object is generated by selecting a device database file. The PROFIBUS view of the controller shows the master and the configured slaves on a stylized bus and the slave objects in the form of bitmaps which are either included in the device database file or can be assigned to the object later. If no bitmap is assigned, a Standard display is shown for the slave.

> Select a PROFIBUS master (PROFI_M_930/PROFI_M_DEV/PROFI_M_772) in the tree view or module view for the PROFIBUS module



te078_us.png

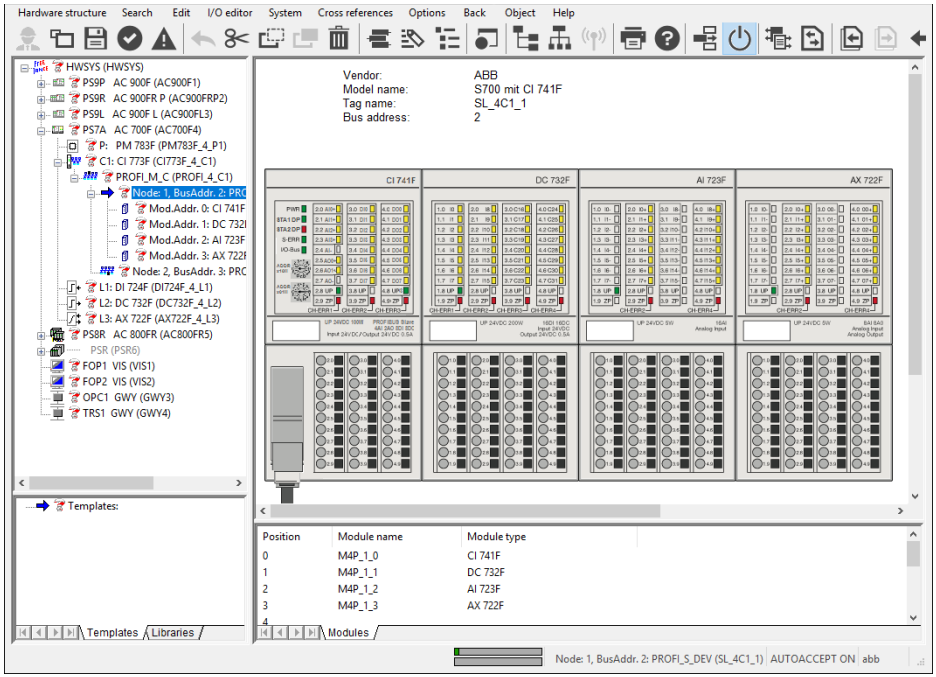
The most important information for the device such as vendor, model name, tag name, and bus address is shown directly on the graphic view. For more information, refer to parameter dialog of the object.

Detail view of a device

The detail view shows detailed information on the device. It is a realistic view which is represented by a bitmap which can be assigned to the slave object.

A list of equipped modules appear below the device view if the slave is modular.

> Select a station object in the tree view or master view



te079_us.png

The most important information for the device such as vendor, model name, tag name and bus address is shown directly on the graphic view. For more information, refer to parameter dialog of the object.

For the modules, the slot position, module name and module type are tabulated.

4.2.3 Menu structure of the Hardware structure

Hardware structure	Save
	Documentation
	Check all
	Check
	Show error list
	Show used I/Os
	Network...
	Communication configuration...
	Exit
Search!	
Edit	Undo
	Cut
	Copy
	Paste
	Delete
	Clipboard viewer
	Templates/Libraries
	Insert...
	Resource allocation...
	Parameters...
	Dialog editor...
	User groups...
	Access rights...
	Block export...
	Block import...
	Activate/Deactivate
I/O editor!	
System	Variable list
	Tag list
	Hardware structure
	Structured data types
	Breakpoint list
Cross references!	
Options	Print
	Tree colors
	Previous object

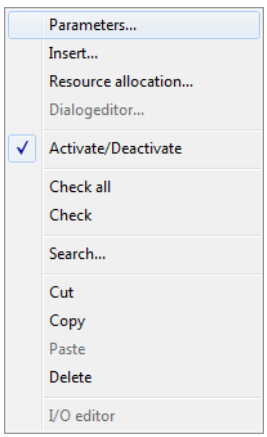
	Next object
Back!	
Help	Contents
	Overview
	About

4.3 General functions of the Hardware structure

4.3.1 Context menu

The context menu provides a context-specific selection of currently available functions.

> Select an object in the tree view or graphic view and right-click



te070_us.png

4.3.2 Cut, Copy and Paste

Cut

> Select station or I/O module to be cut > **Edit** > **Cut**

The selected objects are cut and deposited in the Windows clipboard. From there they can be repositioned with **paste**.

All parameters and I/O channel reservations of the I/O module are moved with it.

Copy

> Select station or I/O module to be copied > **Edit** > **Copy**

The selected objects are deposited in the Windows clipboard. From there they can be positioned as often as required with **Paste**.

All parameters except the module name and all I/O channel reservations of these I/O modules are copied with them.

Insert

> Select position > **Edit** > **Insert...**

New objects can be inserted at the selected position depending on the context. Refer to Insert new objects on page 180.

Paste

The objects previously selected with cut or copy can be reinserted when a destination has been specified.

> Select position where the content of the clipboard is to be pasted > **Edit** > **Paste**

Delete

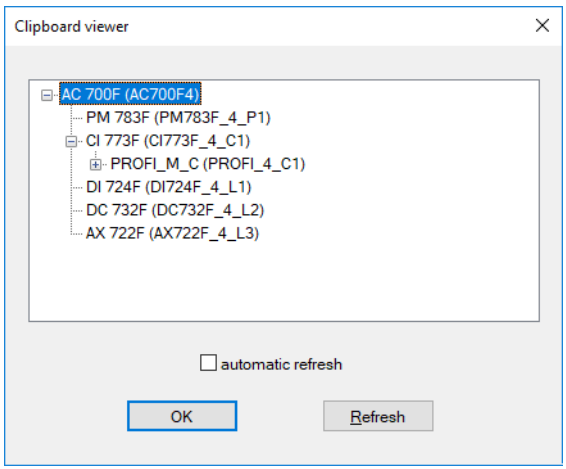
> Select station or I/O module > **Edit** > **Delete**

The parts selected are deleted from the process station. The module function block remains in the tag list and can be reallocated to a module of the same type.

4.3.3 Clipboard viewer

The dialog box shows the objects on the clipboard. In each case a node can be selected and inserted at the required position. The object(s) remain(s) on the clipboard and can be copied again.

> **Edit** > **Clipboardviewer** > Select object > Drag to the destination position holding the mouse button



te104_us.png

If the check box *automatic refresh* is selected, the content of the clipboard will be updated automatically.

4.3.4 Templates/Libraries

Below the tree view an additional tree is overlaid from which templates or components from libraries already applied may be imported into the project.

The templates are already configured hardware objects, the specific parameter settings of which are to be reused. Refer to ***Engineering Reference Manual Communication and Fieldbusses***.

The libraries are used for the FOUNDATION Fieldbus configuration. Object classes may be defined in the various libraries, FF module library, FF H1 device library and FF HSE device library. By applying instances in these classes, a simple configuration of an FF application is possible. Refer to ***Engineering Reference Manual Communication and Fieldbusses, FF-Libraries***.

4.3.5 Export block

To reuse parts of the configuration of the hardware structure at another location, a block of objects or an object with all subordinate objects can be exported.

> Select block/object > **Edit** > **Block export...** > specify the name of the export file (*.hwm)

4.3.6 Import block

After the import, a dialog box appears with the objects which are on the clipboard. In each case a node can be selected and inserted at the required position. The object(s) remain(s) on the clipboard and can be copied again.

> **Edit** > **Block import...** > Select object > Drag to the destination position holding the mouse button

4.3.7 Check all

All objects below the selected object are checked for plausibility and for errors with all activated stations, units, modules and parameter entries, missing entries and inconsistencies. Errors are listed in a dialog box and can be recovered later. Double-click on an error message to select the affected hardware object.

4.3.8 Check

> **Hardware structure > Check**

The selected object and the subordinate incorrect objects are checked for plausibility (for errors, missing entries, or contradictions).

A plausibility check is also offered in parameter dialog boxes and detail views; it then refers in each case to the system stations or modules. Double-click on an error message to select the affected hardware object.

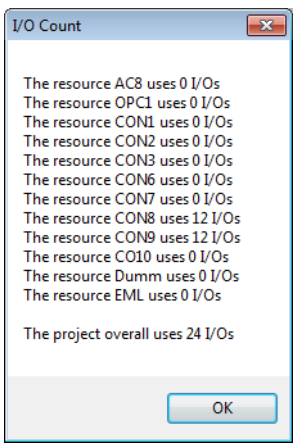
4.3.9 Show error list

> **Hardware structure > Show error list**

The dialog box with the results of the last plausibility check is opened.

4.3.10 Show used I/Os

> **Hardware structure > Show used I/Os**



te113_us.png

The number of I/O components used in the process stations and the total I/O components used in the project appears. For more information, refer to ***Freelance Getting Started Manual***.

4.3.11 Search!

> **Search!**

The text entered in the search dialog box is searched for as part of an object name. A * at the beginning and a * at the end is internally added onto the text which is to be searched. Wildcards cannot be used.

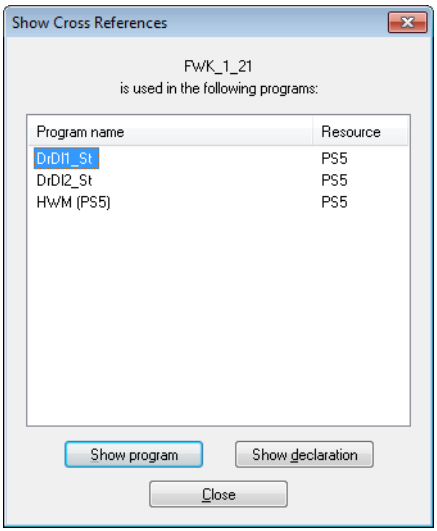
4.3.12 Cross references

Cross references are references relating to an object in programs, displays, lists, and so on. That is, locations where the object or I/O components of this object are used.

Open cross references dialog box from hardware manager. All cross references of an object appear in a list by means of **Cross references**.

> Select object or component > **Cross references!** or **F5** key

A Show cross references dialog box shows the names of the relevant programs in which the object is used. Programs can be selected from the list with the cursor keys or with the mouse.



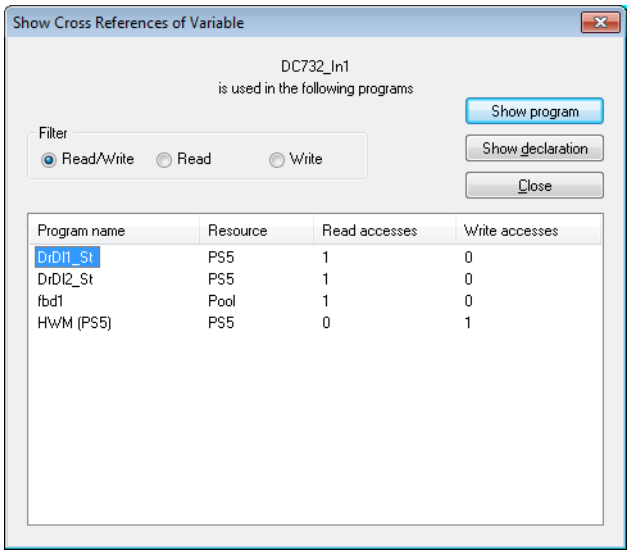
te050_us.png

Open from I/O editor

All cross references of an I/O component appear in a list by means of **Cross references**.

> Select I/O component or variable > right-click > **Cross reference** or **F5** key

A Show cross references of variable dialog box shows the names of the relevant programs in which the I/O components are used. Programs can be selected from the list with the cursor keys or with the mouse.



te051_us.png

Show program

For an I/O component, open the program which uses this I/O component or a variable assigned to this component.

For an object, open the program which uses this object or a component of the object.

Show declaration

For an object, open the tag list, for a variable, the variable list, for an I/O component, the I/O editor of this component is switched to again.

Filter Displays only those I/O components which are used in the programs in read or write mode.

After activation, it is possible to branch to the programs listed as cross references. If the menu item **Back!** is activated in each branched program, the original Editor is returned.

4.3.13 Save

> **Hardware structure > Save**

The changes performed in the hardware are saved in the project database.

4.3.14 Exit

> **Hardware structure > Exit**

Exits from the hardware editor and switches to the Configuration interface view. In case of changes in the hardware structure you will be asked whether they shall be saved or not.

4.3.15 Documentation

> **Hardware structure > Documentation**

The hardware editor is closed and the project Documentation tab opens in the Workspace pane of the Configuration interface.

4.3.16 Undo

> **Edit > Undo**

The **last** action performed (for example, insert or delete an object) is undone.

4.3.17 Parameters

> **Edit > Parameters...**

Or

Double-click on object in the Hardware tree.

The parameter definition dialog box of a hardware object opens.

4.3.18 Tree colors

> Options > Tree colors

4.3.19 Previous object

> Options > Previous object

The hardware object selected last in the Hardware tree is opened.

4.3.20 Next object

> Options > Next object

The hardware object selected, when “Previous object” is opened, will be selected again in the Hardware tree.

4.3.21 Variable list

> System > Variable list

Switches from the Hardware editor to the Variables tab in the Workspace pane. Eventual changes performed in the hardware structure can be saved before.

4.3.22 Tag list

> System > Tag list

Switches from the Hardware editor to the Tags tab in the Workspace pane. The entry of the object selected in the Hardware tree is displayed in the Tags tab. Eventual changes performed in the hardware structure can be saved before.

4.3.23 Structured data types

> System > Structured data types

Switches from the hardware editor to the Structured data types tab in the Workspace pane. Eventual changes performed in the hardware structure can be saved before.

4.3.24 Breakpoint list

> System > Breakpoint list

The Breakpoint management list dialog box is opened to edit the breakpoints.

4.3.25 Configuring access rights to objects

Configuration of access rights to the Hardware manager objects for the various user groups.

> Edit > Access rights...

Install the **Security Lock** by selecting the component in the Freelance setup. For more information, refer to ***Getting Started*** and ***Engineering Manual User Access***.

> Edit > User groups...

4.4 Editing the hardware structure

4.4.1Insert new objects

Context-relevant insertion of objects is possible. In the tree view it is possible to select:

- The system object to insert process, operator or gateway stations

- The AC 900F / AC 900FR, AC 900F P / AC 900FR P, AC 900F L / AC 900FR L, AC 800F / AC 800FR, and AC 700F controllers to insert the modules.
- The Fieldbus modules of the AC 900F / AC 900FR, AC 900F P / AC 900FR P,AC 900F L / AC 900FR L, AC 800F / AC 800FR, and AC 700F controllers to assign master objects for different Fieldbusses. I/O modules rack-based can be inserted under the CAN Master.
- The master objects to insert slaves
- A slave object to insert modules of this device.
- A process station to insert different modules
- The CPU modules DCP 02, DCP 10 or the communication modules DCO 01 to assign different interfaces
- The Emulator to insert the Controller Emulation

Insert function is selected through menu, context menu or toolbar. For one AC 800F or AC 900F Controller, it is allowed to connect only one CAN line with up to five racks.
For the AC 800F the slot of the CAN module is preset to F1 and cannot be changed. If a CAN module is installed in that slot, then the I/O bus (CAN) must be activated. If any other module is installed in slot F1, then the CAN bus must be deactivated.

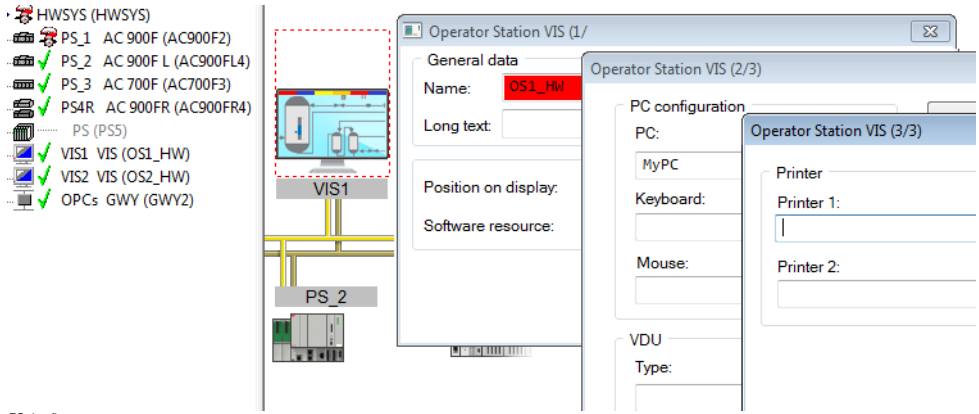
Insert an operator station

> Select system object (HWSYS) in the tree view > **Insert**

Or

> Double-click on operator station position

For documentation purposes, you can enter parameters like keyboard, printer etc. in the parameter dialog.



FO_konfi_us.png

Insert gateway stations

Gateways are always viewed at the control level next to the operator stations even if they run on a CPU module mounted to a process station (DCP gateway).

For **OPC gateways** it is necessary to configure aside from IP address and resource ID also the boot parameters in the hardware structure.

Within the boot parameters is defined how many objects, how much memory and how many communication connections must be available to other stations.

For **DCP gateways** no boot parameters are configured, since DCP gateways run on a CPU module. Therefore a module, type DCP 02GWY or DCP 10GWY, is mounted at the corresponding location. For a redundant DCP gateway two DCP 10GWY modules must be used.

For documentation purposes, the name of this module can be entered at the corresponding DCP gateway.

> Select system object (HWSYS) in the tree view > **Insert**

Or

> Double-click on an operator station position.

The gateway type is defined with resource allocation. For more information, refer to Resource allocation on page 187.

The following gateway types are available:

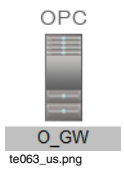
DCP gateway

A DCP gateway is used for coupling the Maestro UX control system. It runs on a CPU module DCP 02GWY or DCP 10GWY which is installed in any process station rack.

Redundant DCP gateway

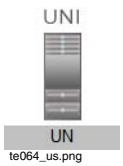
A redundant DCP gateway is used for coupling the Maestro UX control system. It runs on two CPU modules DCP 10GWY which are installed in any process station racks.

OPC gateway



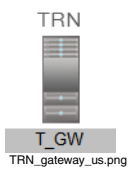
The OPC gateway runs on a PC, on which an OPC-Server F is installed.

UNI gateway



The UNI gateway type is intended for DMS API use.

TRN gateway



The TRN gateway runs on a PC, on which a Trend Server is installed.

Insert process stations

Process stations can be inserted in the system in either the tree view or the system view. After insert, the object appears in both the tree view and the system view. The object is shown at the corresponding station position. According to the selection the following is visualized:

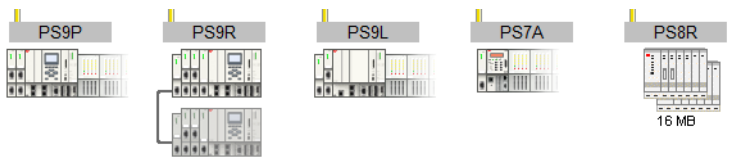
- AC 900F controller
- AC 900FR redundant controller
- AC 900F P controller
- AC 900FR P controller
- AC 900F L controller
- AC 900FR L controller
- A rack-based process station PS
- A rack-based redundant process station PSR
- AC 800F controller
- AC 800FR redundant controller
- AC 700F controller
- Controller Emulator

> Select system object (HWSYS) in the tree view > **Edit** > **Insert**.

> Select station type > Select an insertion position

Or

> Select station position in the system view > Double-click



Hardwarestructure_part_us.png

Default names are assigned when an object is inserted. If the object is inserted in the graphic view, the standard name is created from the object type and the position or mounting location. If the object is inserted into the tree structure, the object type and the first free, available position are used to create the name. If a generated standard name already exists in the project, no name is assigned to the new object. When copying an object, the name is adapted to the new position. When shifting, the name remains the same.

Default names of the modules

S:F1 FI810F (FI810F_4_F1)

S:	Slot
FI810F	Module type, CAN-3 module here
(FI810F _4_F1)	Default name with station position 4, slot F1

Default names of the master objects

PROFI_M_C (PROF_2_C1)

PROFI_M_C	PROFIBUS DPVI master object (device)
(PROF_2_C1)	Default name with station position 2, slot C1

PROFI_M_DEV (PROF_4_F1)

PROFI_M_DEV	PROFIBUS DPV1 master object (device)
(PROF_4_F1)	Default name with station position 4, slot F1

PROFI_M_772 (PROF_1_C1)

PROFI_M_772	PROFIBUS DPV1 master object (device)
(PROF_1_C1)	Default name with station position 1, slot C1

Default names of the slave objects

Node: 0: BusAddr.2: PROFI_S_DEV (SL_4F1_2)

Node: 0	Node 0, the same as the bus address as standard
BusAddr.2	Bus address 2

If the bus address is changed later, the number of the node remains the same. Therefore, it is possible to sort the PROFIBUS devices in the tree and graphic view in a different way than by bus address. The number of the node can be changed in that way that the slave object will move onto the master object.

PROFI_S_DEV	PROFIBUS slave object (device)
(SL_4F1_2)	Default name with station position 4, slot F1 and bus address 2

Default names of the slave modules

Mod.Addr. 0: MODUL (M4F1_2_0)

Mod.Addr. 0:	Module address (= module position) 0
MODUL	PROFIBUS slave module
(M4F1_2_0)	Default name with station position 4, slot F1, bus address 2 and module position 0

Default names of the hart channels

Ch 1: HART (HART_1_001)

Ch 1:	Channel address 1
Hart	Hart channel object
(HART_1_001)	Consecutive numbering from 001 to 999, valid for the whole project.

Default names of the link modules

R:1 S:L DLM 01 (DLM01_4_0_L)

R:	Rack
S:	Slot
DLM 01	Module type (for example, link module)
(DLM01_4_0_L)	Default name with station position 4, rack 0, slot L

Default names of the serial interface

Below the CPU modules (DCP 02, DCP 10), the communication module (DCO 01) or the serial module DFM 02, the serial interface assignments are shown as follows:

Ser1:	MODM_DEV (MODM_2_0_1_2)
Ser1:	Serial interface 2

CPU modules only feature one serial interface; four interfaces maximum are available for DCO 01.

MODM_DEV	Modbus master object device (unit)
(MODM_2_0_1_2)	Default name with station position 2, rack 0, slot 1, serial interface 2

Default names for the remote process controller

BusAdr 3: FLRC_MSL (FLRC_MSL_001)

BusAdr 3:	Bus address 1-254
FLRC_MSL	Freelance Remote Control as Modbus Slave
FLRC_MSL_001	Default name with class name for Freelance Remote Control as Modbus Slave, consecutive numbering from 001 to 999, valid for the whole project.

4.4.2 Resource allocation

D-PS, D-GS and/or D-LS resources configured in the project tree must be associated to the respective hardware stations.

In this way the software parts of the user program configured in the project tree are allocated to the hardware stations.

> Select a station > **Edit** > **Resource allocation...**

> Select the corresponding resource from the list

Or

> Double-click on the gray text area of the station

> Select the corresponding resource from the list

Allocation of a resource to a station automatically activates the station or in the Freelance Engineering operating mode of **Commissioning** the station is searched for in the Control Net.

Delete the resource allocation

Deleting the resource allocation deactivates the stations.

> Select a station > **Edit** > **Resource allocation...**

> Select **No allocation to resource**

Or

> Double-click on the gray text area of the station

>Select **No allocation to resource**

Data and parameters previously entered are **not** lost by deleting a resource allocation. When allocation is continued, the data and parameter reappear and can be reedited.

4.4.3 Activate/Deactivate objects

A deactivated object is not loaded into the process station or Controller or one of its subordinate devices. If the deactivation of an entire station is saved in the hardware structure, the entire station must be reloaded (that is, initialized), after reactivation.

A deactivated object appears in gray in the tree view.

In the graphic view, the object is shown in gray or in a lighter shade within or below the Controller.

Deactivated objects are not edited during the plausibility check process.

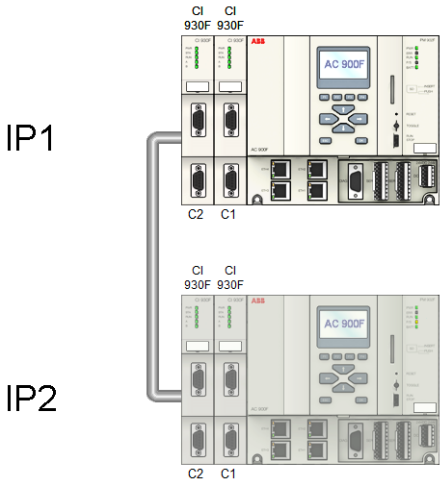
In the graphic view of a process station, the CAN cable connection between the racks is represented open and in the activated state closed. The frame of the CPU or I/O unit and/or a deactivated module are also shown in gray.

> Click on the object > **Edit** > **Activate/Deactivate**

Or

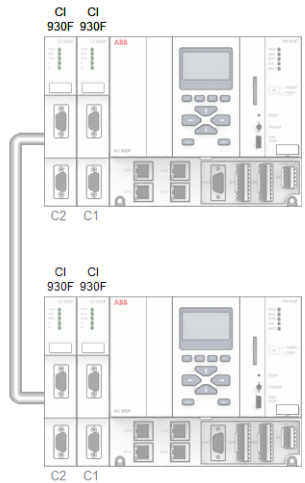
> Right-click on the object > **Activate/Deactivate**

activated:



te107_us.png

deactivated:



te108_us.png

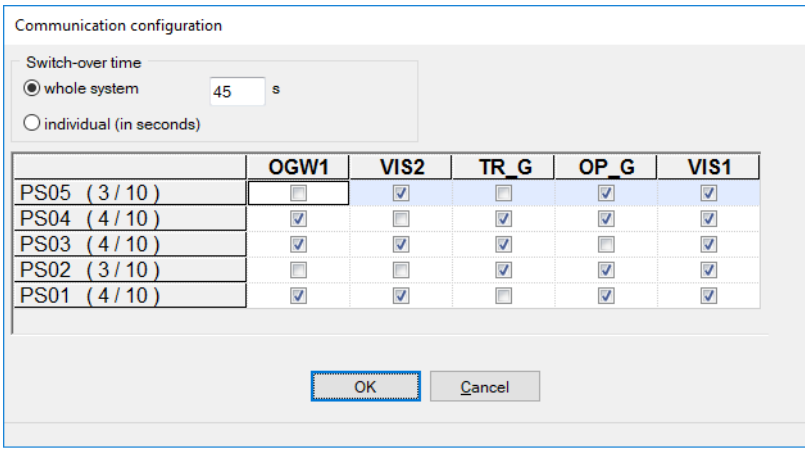
4.4.4 Configuration of the communication connections

For data exchange with operator and gateway stations, 10 connections are available in each process station. For data exchange between the OPC server (OPC-S) and operator stations, 20 connections are available for each OPC server. If more operator or gateway stations are configured, the operative communication connections of all users of a Freelance system can be defined in the Communication configuration dialog.

> **Hardware structure > Communication configuration...**

Or (within project tree configuration)

> Double-click on the **CONF** project object.



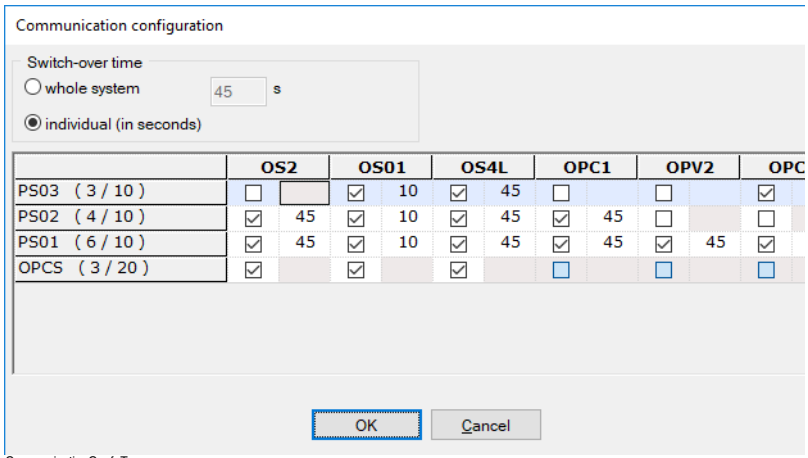
CommunicationConf_us.png

The left column lists all process station resources, and the table header contains all operator and gateway station resources with their names configured in the project tree. Every field of this matrix represents a connection between a process station and an operator or gateway station.

If a connection between the process station and operator or gateway station is to be set up, the corresponding check box must be ticked ☑.

Up to 10 connections can be configured in each line of the matrix.

Additionally, the Switch-over time (time-out) for the redundant network connections can be configured. This can be done for the entire system, i.e. the same switch-over time is configured for all connections, or you can set an individual time for each connection. A switch-over time between 1s and 60s can be specified. The default value for a newly created or updated project is 45s. The switch-over time for Freelance Engineering is 45s; this time is not configurable.



CommunicationConf_T_us.png

In the event of a connection failure, a switch-over to the redundant Control Net line occurs within the predefined switch-over time, and the corresponding signal or message is output. If no redundant Control Net has been installed or configured, the communication failure is only signaled or reported.

4.5 I/O editor

Each module provides I/O components with respect to the number of channels. The channel of an I/O module can be used directly in programs and graphic displays through the I/O component. If the I/O component appears in the variable list, it is possible to indicate a variable name.

I/O components without a variable association do not show up in the variable list and therefore cannot be exported for lateral communication and cannot have an initial value.

If the access to an I/O component is programmed by the <Tag name>.<Component name> syntax, deleting the referenced I/O device will set all programs to incorrect. To get a correct program it is necessary to create a new I/O device instance with the name <TagName> which supports the referenced component <ComponentName>.

If the access to an I/O component is programmed by a variable, deleting the corresponding I/O device sets all programs to incorrect. To get a correct program, the programs must be checked again.

Both I/O components and variables can be opened in programs and displays through the **F2** key.

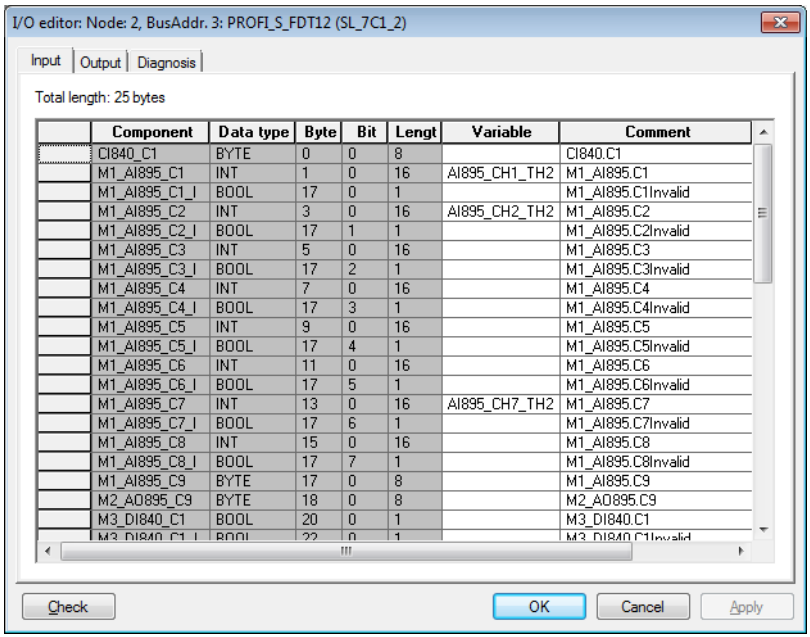
The I/O component name is composed of the module name (16 characters) and the component name (16 characters), so that ‘module name.component name’ provide a total of 32 characters.

The ‘Module name’ is the tag name of the module, the ‘Component name’ is the name of the I/O or diagnosis component.

Variable names cannot have the same name as tag identifications. Such a name allocation is stopped by the check for plausibility.

4.5.1 Opening the I/O editor

> **I/O editor!**



te048_us.png

The I/O editor is composed of several tabs indicating inputs, outputs and diagnosis components. The columns on the tabs can be sorted with a double-click on the column header.

Component This column shows the names of the components. Depending on the type of the module, the module channels are marked Ch0 to Ch7/15/31 by default setting. It is possible to change the component name (maximum 16) but within a module the name must be unambiguous.

By renaming a component, all programs referring to this component become incorrect.

Data type The variables for analog modules are processed as variables of data type UINT from the analog I/O module, except for the temperature

module DAI 04 (REAL). The useful signal is 12 bits wide. These variables can be converted into a REAL format for further processing in the programs. For example, to switch them to analog monitoring.

This is done with converter modules, in which the ranges are configured for further processing or given default values. For example, in case of wire break. These converter modules transpose the process signal (for example, 20 mA) to a physical value, or conversely the physical value to an output signal.

An exception is the DAI 04 module, which carries out this conversion on the module by linearizing the signals over a corresponding characteristic.

The signals of the digital modules are supplied to the system with variables of data type BOOL. The variables do not require a converter, as their states can be processed directly.

By changing the data type, all programs referring to this component become incorrect.

Byte Zero-based byte offset of the component in the actual section. The value of the component starts at this offset and ends at offset plus the size of the component data type. Only byte values equal to or larger than 0 are allowed.

Changing the byte offset will not adjust the byte offset of other components. If the byte value is changed all references to components of this I/O module or unit in user programs or wherever variables can be referenced will be set to implausible.

Bit Zero-based bit offset of the component in relation to the corresponding byte. Only bit values in the range 0 to 7 are allowed.

Changing the bit offset will not adjust the bit offset of other components. If the bit value is changed all references to components of this I/O module or device in user programs or wherever variables can be referenced have to be set to incorrect.

Bit Length Bit length of the corresponding data. This parameter must be specified.

Variable Different name for the I/O component. Enables access to I/O components not only by the new syntax described above but also in the same manner as in earlier Freelance Engineering versions. This is an optional parameter. If this field is left empty, the I/O component can only be accessed by the new syntax. The variable name must be unique for all variables. Pressing **F2** shows a list which contains all 'unmapped' variables of the project which are located on the same process station and have the correct data type. All variables show up in the variable list with the same behavior as any other variable. Variables which have not yet been configured in the project, can be declared directly in the I/O editor.

Only I/O components with allocated variables can perform lateral communication, that is, can be read in other resources.

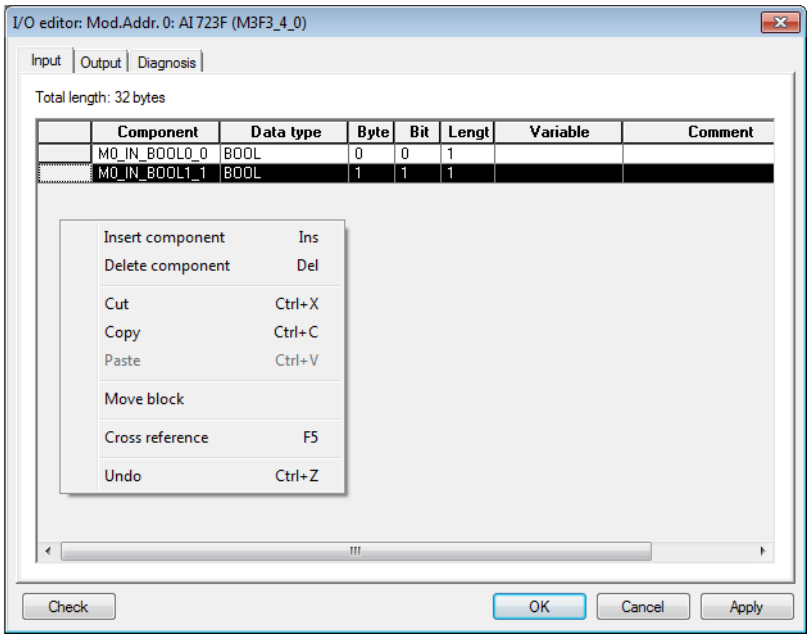
If the variable name is changed all references to the old and to the new variable will be set to incorrect.

Comment Comment for the I/O component. This is an optional parameter. The maximum length of this entry is 31 characters. Changing the comment for references to an I/O component has no effect on the check state.

Check Press **Check** button to check the I/O definition (overlapping bytes, gaps, data types, and so on) of I/O module.

4.5.2 Edit components

> Select object > right-click > Context menu



te101_us.png

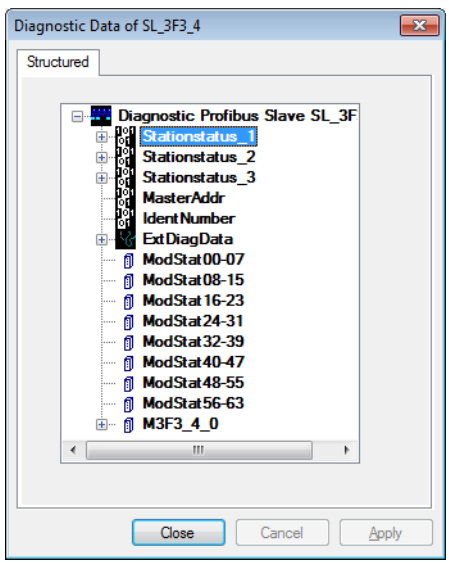
Insert component

> Right-click on a **field** or **line** number > select **Insert component**

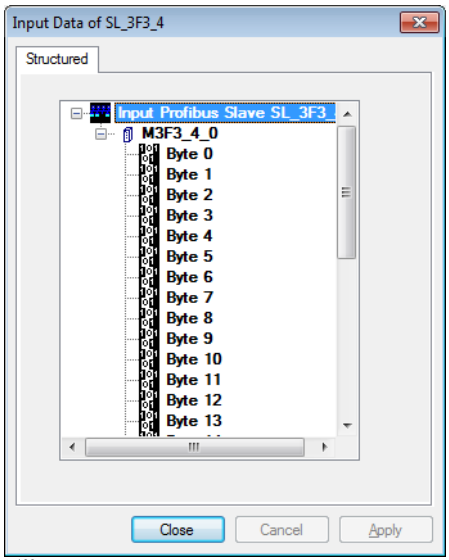
A new component is inserted. All lines below are moved down. “New0000” is defined as the first component name and then “New0001”, and so on.

If a component for the modular slave is inserted, a special dialog appears which makes it possible to access individual items of information on the field device. A distinction is made between structured access and standard access. With structured access, the structures of the device data are defined by the user or device vendor. With standard access, the device data is listed under the unstructured slave object. Both views show all device data which has not yet been assigned to a component.

> Right-click on a field or line number > select **Insert component** > select component > right-click > select data type



te105_us.png



te106_us.png

Delete component

> Right-click on line number, possibly drag block > select **Delete component**

The selected components are deleted. All lines below are moved up.

Components can only be deleted if the entire line has been marked beforehand.

Copy

> Right-click on line number, possibly drag block > select **Copy**

The selected components are placed on the clipboard. Using **Paste**, the lines/components to be copied can be inserted, after having selected the destination position/line.

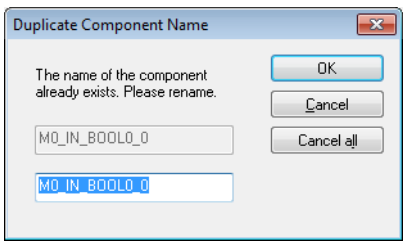
Components can only be copied if the entire line has been marked beforehand.

Paste

> Right-click on line number > select **Paste**

The components previously placed on the clipboard are inserted.

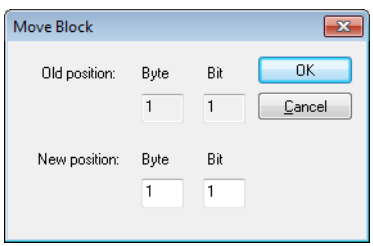
The contents of the clipboard can only be inserted if the entire line has been marked. If it is discovered, during insertion, that the corresponding component already exists, a new component name can be entered.



te103_us.png

Move block

> Right-click on line number > select **Move block**



te102_us.png

The position of a component can only be displaced if the entire line has been marked.

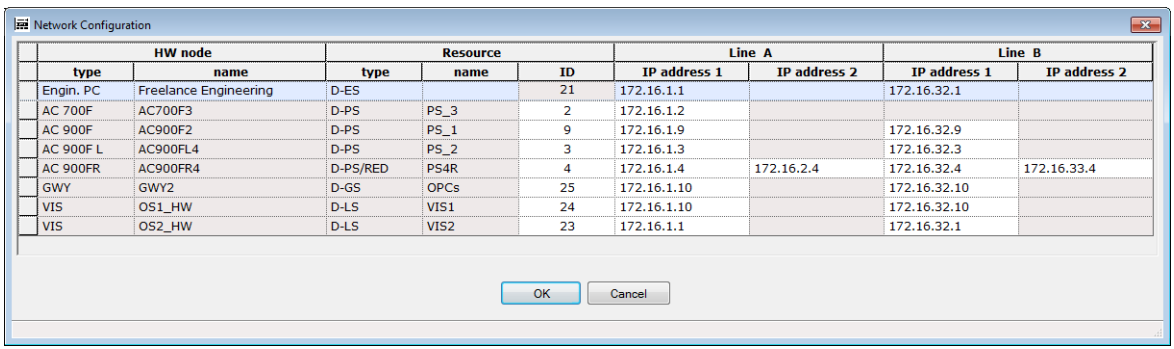
Change data type

> Click on data type field

The data type for digital module channels can be set (BOOL, BYTE, WORD, DWORD). The data type for analog modules is permanently set to UINT. In addition, the following data types are for field devices: INT, DINT, UDINT and REAL.

4.6 Network configuration

> Hardware structure > Network...



HW node		Resource			Line A		Line B	
type	name	type	name	ID	IP address 1	IP address 2	IP address 1	IP address 2
Engin. PC	Freelance Engineering	D-ES		21	172.16.1.1		172.16.32.1	
AC 700F	AC700F3	D-PS	PS_3	2	172.16.1.2			
AC 900F	AC900F2	D-PS	PS_1	9	172.16.1.9		172.16.32.9	
AC 900F L	AC900FL4	D-PS	PS_2	3	172.16.1.3		172.16.32.3	
AC 900FR	AC900FR4	D-PS/RED	PS4R	4	172.16.1.4	172.16.2.4	172.16.32.4	172.16.33.4
GWY	GWY2	D-GS	OPCs	25	172.16.1.10		172.16.32.10	
VIS	OS1_HW	D-LS	VIS1	24	172.16.1.10		172.16.32.10	
VIS	OS2_HW	D-LS	VIS2	23	172.16.1.1		172.16.32.1	

NetworkConfiguration_us.png

4.6.1 IP address and resource ID

All stations are linked by the Control Net. The Control Net is based on the Ethernet standard with the TCP/IP communications protocol. The word Ethernet denotes local networks complying with DIN ISO standard 8802, part 3 and the now valid standard IEEE 802.3.

The Freelance Engineering makes a difference between **Resources** and **Stations**. Stations are units connected to the Control Net with a definite IP address.

Resources are software parts loaded to the stations. In the project tree they appear as resource nodes and identified in the network by a unique **resource ID**.

The differentiation between resource ID and IP address is necessary as there are stations in which several resources can be loaded (for example, one PC with Freelance Operations and a gateway); but resources can also be addressed through two different IP addresses (for example, a redundant process station).

In the example “Freelance Operations and a gateway on one PC”, two different resource IDs must be allocated to these components for installation on the PC under which the two software parts can be addressed. For a project configuration, the same IP address (the address of the PC in which the resource is loaded) is allocated in the hardware structure for both resources and two different resource IDs are entered

In the example “Redundant process station”, two different IP addresses must be entered for the CPU module.

IP addresses

During the system setup, the IP addresses must be allocated to the individual Freelance stations. In case of local networks that are not connected to the outside world via a router, an address from the address block172.16.0.0/20 should be assigned. The standard setting for Freelance is the subnet mask 255.255.240.0.

If redundant AC 800F process stations are used, the subnet mask 255.255.240.0 must be set.

f two Ethernet modules will be used at the slots E1 and E2 of a non-redundant AC 800F, take care that an IP address out of various subnets is assigned to each Ethernet interface.

It is recommended to set e.g. for E1 an IP address within the range of 172.16.[0 - 15].[0 - 255] and for E2 an IP address within the range of 172.16.[16 - 31].[0 - 255].

An address in the same subnet, e.g. between 172.16.0.1 and 172.16.15.254, must be allocated to all Freelance stations on the same Control Net, Note that the first and last address of a subnet (broadcast addresses) must not be used.

In a Control Net in a redundant configuration, addresses in separate subnets must be allocated for Line A and Line B, e.g.:

Line A: 172.16.0.1 to 172.16.15.254

Line B: 172.16.16.1 to 172.16.31.254.

If no Control Net redundancy is required, the address fields for Line B have to remain empty.

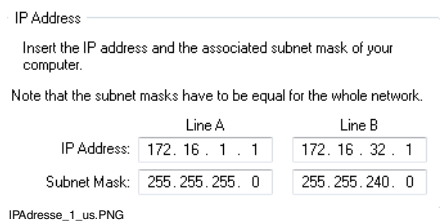
If Freelance stations are in a network with other users, allocate corresponding IP addresses by the network administrator.

For more information about how to set the IP addresses refer to the Mounting and Installation Instructions of the corresponding controller.

They need not be considered when configuring the hardware structure, as the IP addresses are independent of them.

General settings

Directly after the Freelance software has been installed, the Settings dialog for the Freelance components is opened. In this dialog, you have to define the resource IDs for the installed software components. Under General settings / IP addresses the IP addresses and subnet mask for Line A and Line B of the Control Net must be specified. If no Control Net redundancy is needed, the input fields for Line B must remain empty.



IPAdresse_1_us.PNG

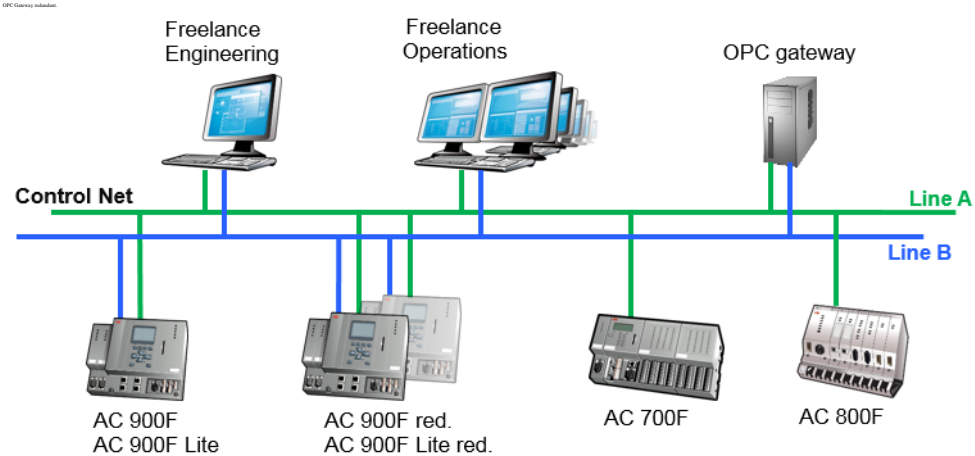
4.6.2 Control Net Redundancy

Optionally, the communication between AC 900F process stations and operator or gateway stations as well as lateral communication between AC 900F process stations can be realized via a Control Net in a redundant configuration. The following Freelance stations support Control Net redundancy:

- AC 900F
- AC 900F redundant
- AC 900F Plus
- AC 900F Plus redundant

- AC 900F Lite
- AC 900F Lite redundant
- Freelance Engineering
- Freelance Operations
- OPC gateway

OPC Gateway redundant

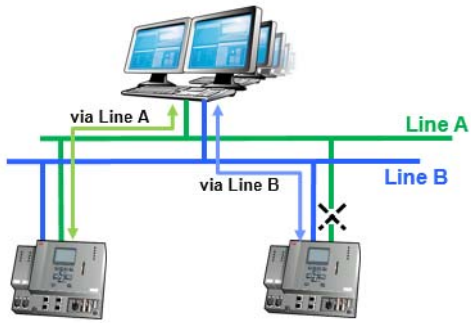


ControlNetRed_1_us.png

The AC 900 F Ethernet connector ETH1 is used for Control Net Line A, ETH3 is used for Line B. If no Control Net redundancy is configured or stations are used which do not support Control Net redundancy, the communication is realized via Line A (ETH1).

For communication between the operator or gateway stations (horizontal communication) only the configured Ethernet line is used. Redundancy is not supported. The trend server, for example, communicates with the process stations via the redundant Control Net, i.e. Line A and Line B. The communication between the trend server and the operator station, however, is non-redundant and only uses the configured network, typically Line A.

The communication between the AC 900F controllers and the operator or gateway stations (vertical communication) is on the active Control Net line. In the event of a failure of the active line, a switch-over to the redundant line occurs within the configured switch-over time (see Configuration of the communication connections on page 189).



ControlNetRed_2_gr.PNG

The data exchange (lateral communication / time synchronization) between AC 900F controllers is always done using both Control Net lines. A failure of any of the lines has no effect.

The lateral communication between the AC 900F controllers and the AC 700F, AC 800F or Freelance rack system is realized via Control Net Line A. The AC 700F, AC 800F and Freelance Rack system do not support Control Net redundancy and solely communicate via the Control Net Line A.

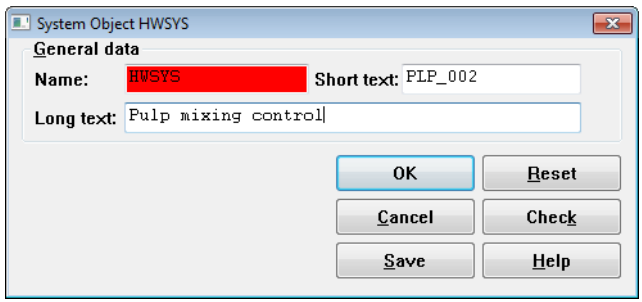
4.7 General parameters of the hardware structure objects

All stations such as process stations, AC 900F controllers, AC 800F controllers, AC 700F controllers or operator stations and their subordinate objects such as modules are elements/objects in the hardware structure. These objects have an object name and a parameter dialog with specific object parameters.

4.7.1 HWSYS system object

HWSYS system object is only a structure element comparable to the project element “Configuration” in the project tree, as the project element configuration stands for the total of all software resources, the system object stands for the total of all hardware stations such as process, operator or gateway stations.

With the system object the system view is selected in the graphic part of the hardware structure.



te028_us.png

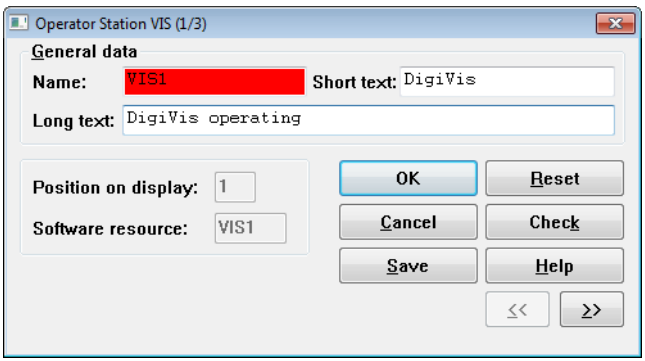
4.7.2 VIS operator station

The **Operator stations** are commercial PCs in which the software program Freelance Operations has been installed for process visualization. All displays and records are configured with the software program Freelance Engineering and loaded in the operator station. Data acquisition for recording and trend display takes place in the process station, and all display and archiving in the operator station.

Each process station can communicate with up to 10 operator and/or gateway stations.

The communication connections are enabled under **Connection configuration**. For more information, refer to Section 3, Project tree on page 67 and Display target stations on page 89.

The station (D-OS) must be allocated to a resource. With this you define which graphic part of the application program is run on which operator station. For more information, refer to Resource allocation on page 187.



te030_us.png

General data

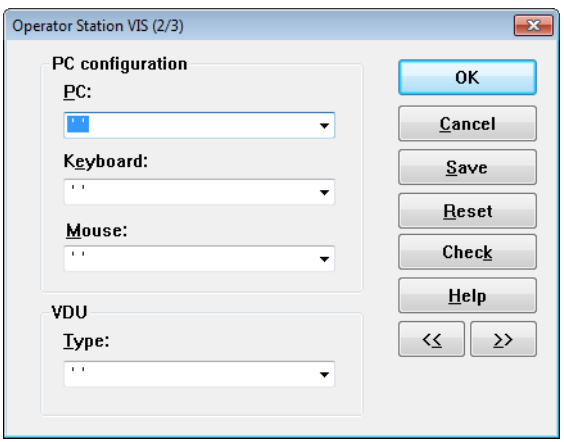
Position on display

Position in the graphic view of the System display.

Software resource

For example, process station in the project tree.

With **>>** turn to the next parameter definition dialog box.



te040_us.png

PC configuration (only for documentation)

PC PC for Freelance as operating system.

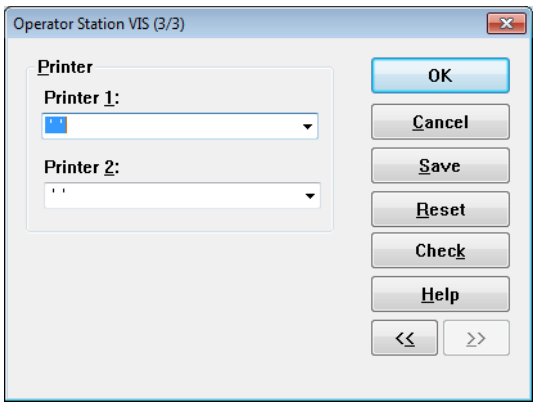
Keyboard Type of keyboard

Mouse Type of mouse

VDU

Type Type of monitor

With **>>** turn to the next parameter definition dialog box.



te041_us.png

Printer

Printer 1 Operator station printer

Printer 2 Operator station printer

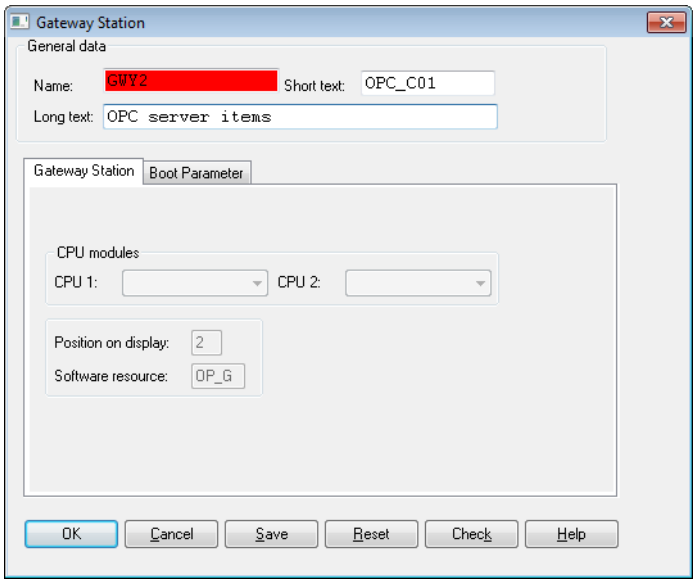
Only for printer documentation, not for printer setting.

4.7.3 GWY gateway station

Gateways exist for the connection of **OPC server**, **Maestro UX**, for any Clients such as Microsoft Excel or Visual Basic applications **OPC clients** and for individually created applications (**DMS API**).

The maximum number of gateway stations are defined in the project tree element configuration. For more information, refer to Project objects of a OPC function block library on page 77. The hardware configuration must be allocated to a resource (project tree), the gateway type is only defined by means of such an allocation. For each gateway station configuration, the number of configurable

control stations is reduced by one. For more information, refer to Resource allocation on page 187.



te059_us.png

Gateway station tab

CPU modules

CPU1 Name of the corresponding CPU module (DCP gateways only)

CPU2 Name of the corresponding CPU module (redundant DCP gateway only)

Position on display

Position in the graphic view of the system display

Software resource

Allocated resource (project tree)

Boot parameter tab

Memory These numbers influence the partitioning of the memory within a DCP gateway. If errors occur in loading of the objects when there is a very large number of objects in the gateway, change these settings to load the project.

Max. no. of Objects
Maximum number of objects which can be loaded on the gateway.

Configuration data (PRAM)
Storage area, in KB, which is reserved for the configuration data. This storage area is cold-start stable (only with DCP gateway).

System limits
Number of connections
Maximum number of configured communication connections. For more information, refer to Configuration of the communication connections on page 189.

4.7.4 Process station

The process station is defined by the controller or the CPU. The process stations are divided into two classes: the conventional process stations where the I/O modules are assembled in module assembly frames and the process is carried out in one or two (redundant) CPU modules, and the AC 900F / AC 800F / AC700F controller where several field busses as well as rack-assembled modules can be connected.

For more information on process stations, refer to ***Engineering Manual Process Station.***

For more information on PS and PSR process station types, refer to ***Engineering-Manual Process Station Rack-System***.

4.8 Controller emulation

The controller emulator allows testing and simulation of user programs even without connected hardware. It runs on a PC and emulates all functionality of the Freelance controller with the exception of the I/O functionality. Multiple emulators can be used, and emulators can coexist with Freelance controllers in a project.

4.8.1Restrictions

The controller emulator support only parts of the real AC 700 / AC 800 / AC 900 controller. Even if the controller emulator can be configured similar to a normal process station, it can be used for functional test purpose in the office or in a test environment only. The plausibility check for the emulator configuration isn't as

comprehensive as the check of a real controller. Therefore it might happen that an emulator configuration contains configuration errors which are not detected (e.g. TCP communication between emulators on the same PC when using the same ports or usage of ports that are already used by other applications on the PC that hosts the emulator).

The controller emulation is for test purposes only and may not be used in real production environments.

The emulator has a different timing behavior than a real process station and the reaction on exceptional situations and the error handling are different to a real controller. Therefore only functional tests are possible.

The following features are supported:

- Usage of all function blocks according IEC 61131 programming that are not hardware related.
- Send and receive communication (UDP and TCP).
- Alarm handling with Freelance Operations.
- Data access with Freelance Operations.
- Access from gateways (OPC, DMS-API).
- Data exchange with other controllers using exported variables (lateral communication).
- Time synchronization by Freelance controller

The following features are not supported or behaves different to a real process station:

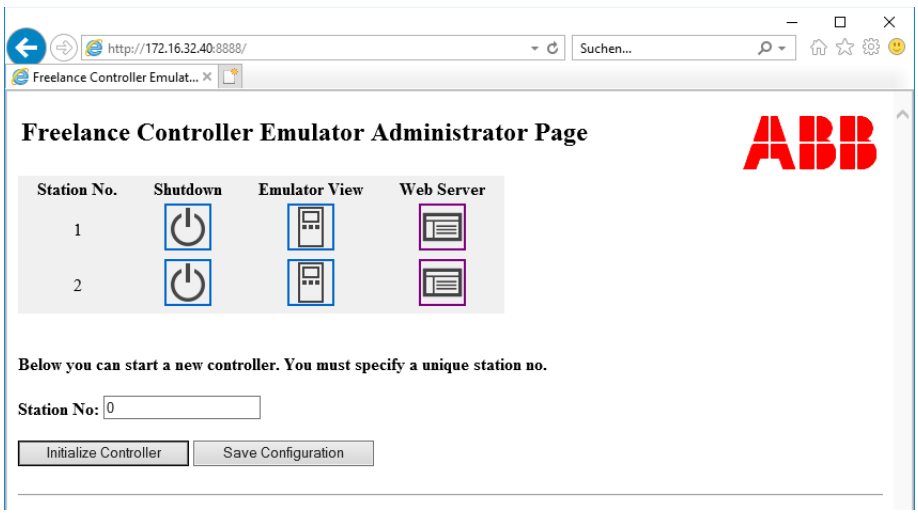
- Hardware related function blocks
- Redundant tasks

- Task priorities, only four priorities are supported: 50 for the default task, 51-94 (low), 95 (normal), 96-99 (high). If the PC has more than one core and the emulator is not bound to a single core using Windows affinity, multiple tasks can execute simultaneously)
- Error handling in case of exceptional situations like
  - Division by zero
  - Overflow of values (depending on used data types)
  - Power down / power up behavior
  - Error correction of memory
- Controller redundancy (redundant projects can be loaded, but a redundant emulator is not supported)
- Resource handling (memory, …)
- No Modbus serial communication
- No Modbus TCP communication
- No serial telecontrol communication (IEC 60870-5-101)
- No Ethernet telecontrol communication (IEC 60870-5-104)
- No support of PROFIBUS interface
- No support of Foundation Fieldbus interface
- No support of CAN interface
- No support of Web server diagnostics (event log, task statistics)
- No time-master function
- No time synchronization via radio-clock or OPC gateway

4.8.2 Insert and start an emulator

The configuration is similar to the other station types. Simply insert an object of type EMULATOR into the hardware structure and assign the process station.

To create and start an emulator go to the emulator administrator page by entering the IP address http: // localhost: 8888 or http: // <IP address>: 8888 in the web browser.



Emulator_1.PNG

Station No. Station number (Resource-ID) of the process station to be emulated

Shutdown Stops the controller emulation

Emulator View
Open the controller emulator interface

The controller emulator interface is a Java application. Depending on the Java security settings, this application may need to be added to the exception list (http://<IP-Adr>:8888/SCGui.html).

Web Server Open the controller emulator diagnostic window

Initialize Controller
Starts a new controller emulator with the specified station number

Server Configuration

Saves the configured emulators, so they will start automatically the next time you start the PC.

5 Commissioning

5.1 General description – Commissioning

Commissioning is an operating mode of Freelance Engineering which offers a range of other functions in addition to the loading of project objects. However, in contrast to the configuration operating mode, the user programs cannot be changed structurally. Only when the user programs have been configured and checked for plausibility, can they be loaded into the process station and started as part of the commissioning process.

When a project is commissioned for the first time, the complete project must be loaded into the respective station.

Later, only changes need to be loaded. Changed project objects are loaded and started in accordance with selections made in the project tree. The selection of single or multiple user program objects is made in the project tree, as already described under configuration.

As well as **starting**, **loading**, **stopping** or **initializing** project objects, such as a resource, a task or a program list, it is also possible to edit individual function blocks of programs. This enables the operator to change operating modes, switch to a specified operating state and set parameters for function blocks without having to reload the modified program. There are two methods of making these changes known to the system:

- **Write** loads the changes without saving them in the project file.
- **Correct** saves the changes in the project file, as well as loading them.

Although they share a common user interface, configuration and commissioning are two separate processes. For commissioning, a **connection** must be established from the engineering station to the process station(s) and the operator station(s) via the Control Net. It is then possible to switch directly between configuration and commissioning modes.

After loading a project with **Load whole station**, additional configuration changes may be loaded incrementally. A configuration change consists of the creation, deletion or a change to a project object. A configuration change which effects other project elements is said to have **side effects**.

During loading, the existence of side effects will cause the changed project object and any objects affected by it to be stopped. During operation the side effect changes should be loaded with caution. The user is made aware of the occurrence of side effects by the way the object nodes concerned are shown in the project tree.

Version checking ensures that the project running on the commissioning station and the process station are one and the same. As regards the loading of changes, configuration changes which have not been downloaded yet are indicated by arrows on the objects concerned.

Commissioning of the programs is facilitated in all cases by a global **value and trend window**. The user can track analog and binary values from one program or another.

In **FBD** and **LD programs**, binary values appear directly with their logical state of 1 or 0. The state of the binary signal is recognized by a different line type.

In the **IL programs**, the current contents of the accumulator appear in their own column.

In **ST programs** all local variables can appear in the value window.

In **SFC programs**, the processing of the transitions and steps can be controlled. The language allows the transitions to be blocked or forced. The steps can be processed in permanence or permanently switched off. In addition, parameters such as the step wait time (TWA) and the step monitoring time (TMO) can be altered.

Binary and analog values can be set on a one-time basis to test the reaction or functionality of the program.

Values can be **forced** to remain constant (for debugging purposes) at the I/O modules and field devices. This is achieved by switching to the hardware structure and selecting the requisite object.

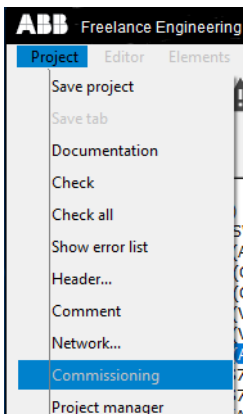
5.2 Starting commissioning

Unlike configuration, commissioning can only be performed once a connection to the process station has been established through the Control Net (Ethernet). If the Control Net is used in a redundant configuration, communication is possible on both lines, Line A and Line B. If the primary line should be interrupted, an automatic switch-over to the secondary line is performed. Communication is set up automatically when **Commissioning** is selected.

From Configuration interface, select **Project > Commissioning**

It is possible to switch between configuration and commissioning modes directly from the project tree.

For more information on installation of the Control Net connection, refer to ***Mounting and Installation Instructions Manual***.



Commissioning_us.png

Connection requirements

- Ethernet card correctly installed in the engineering station D-ES.
- “Internet log (TCP/IP)” added to the network configuration.
- Control Net cable connected correctly to the stations to be commissioned
- Resource IDs and IP addresses entered correctly in the stations.

- All resource IDs or IP addresses are unique (Exception: stations which contain several resources, e.g. a PC with Freelance Operations and a gateway).
- Following an upgrade/update, the current firmware has been loaded to the process station
- Current firmware updates downloaded to modules

5.2.1 Commissioning procedure

- The objects to be loaded into the process station must have satisfied plausibility checks.
- The start-up characteristics of the project must be known and must be checked prior to commissioning. The automatic startup of the function blocks, program lists, tasks and process stations is preset and can be configured as required.
- Open Commissioning
- Select the project objects to be loaded from the project tree and load them.
- If no automatic startup has been configured, initiate processing in the function blocks, switch on the program list, start the task and start the resource.

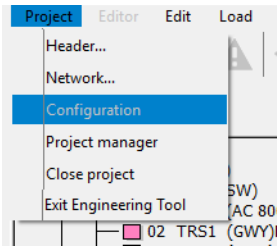
5.2.2 Switching between commissioning and configuration

Direct switching between commissioning and configuration modes is possible.

Commissioning mode is not available if the system detects the running of BDM.

Switching from commissioning to configuration

> Project > Configuration



di1513_us.png

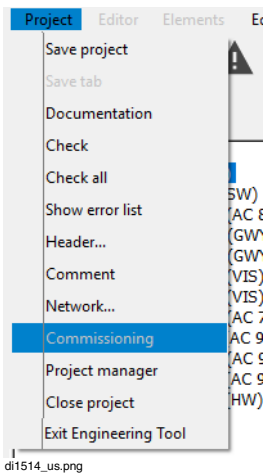
It is necessary to switch to configuration when, for example:

- Objects are to be added or changed
- Variable changes are to be performed, such as adding new variables or changing data types.
- Extra tags or variables are to be configured in a resource

Switching from configuration to commissioning

> Project > Commissioning

After a switch from commissioning mode into configuration mode, it is not permissible to switch from the program directly back to commissioning. The reason is possible side effects which any changes made might have on other objects, such as programs or displays. Visible indication of side effects is provided only in the project tree display; hence a switch to the project tree is required before a return to commissioning.



di1514_us.png

It is necessary to switch to commissioning when, for example:

- The user program is to be loaded
- After plausibility checks, objects are flagged for loading with an arrow symbol
- Tag parameters are to be corrected on a running system
- Processing of individual programs is to be monitored
- Individual signals from or to a field object are to be debugged (forcing).
- DVP1 parameters of a PROFIBUS field device are to be loaded.

5.2.3 Quick switch from commissioning to configuration mode

A quick switching from commissioning to configuration mode has been implemented through the menu item **Configuration** or the toolbar available in the following editor or list displays:

- FBD program
- LD program
- IL program
- ST program
- SFC program
- Hardware structure (system structure and station review)
- Variables list
- Tag list

The mode change always applies to the entire Freelance Engineering system. Thus, switching for example from the commissioning to the configuration mode within an editor causes that if one subsequently moves to the project tree, it will also be in configuration mode.

When changing the mode, the current selection, the current block selection, the current display section, the current search filter and the current sort sequence are all retained within the editors or lists.

If dialogs such as the **'Define debug window'** or the operator dialogs in the SFC program are opened, then the values and trend windows will be closed.

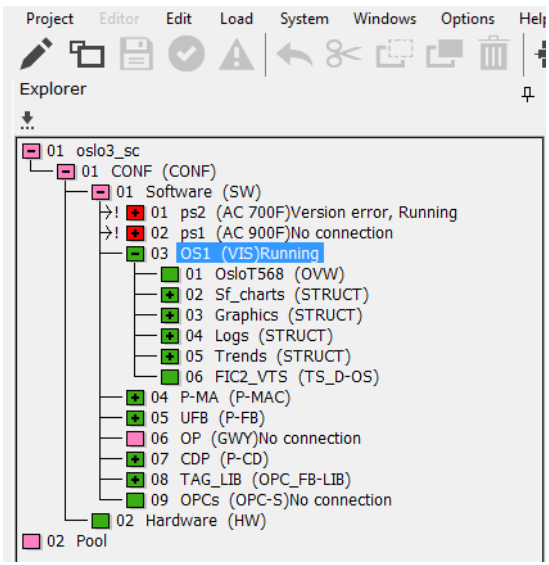
5.3 Commissioning user interface

5.3.1 Project tree

In commissioning mode, as in configuration mode, the project tree forms the starting point for all further operations.

Unlike in configuration mode, state information such as ‘running’ or ‘stopped’ is kept for the individual project objects.

It is clearly indicated which objects need to be loaded. These project objects are shown preceded by an arrow. If a compressed branch contains at least one subordinate object that needs to be loaded, then an exclamation mark precedes the visible representative (top most object) of that branch.



di1515_us.png

For more information, refer to Section 3, Project tree.

5.3.2 Menu structure of the Configuration interface in commissioning mode

Based on the active tab or editor, the menus will be updated.

5.3.3Differences between commissioning and configuration modes

Project tree functions of configuration

Menu options in bold can only be used during **Configuration**, not during **Commissioning**.

Project	**Save Project**	System	Variable list
	Save Tab		Tag list
	Documentation		**Structured data types**
	Check		Global message processing
	Check all		Local message processing
	Show error list		**Hardware structure**
	Header...		**Display access**
	Comment		**Communication configuration...**
	Network...		**Area definition...**
	Commissioning		Show global variables...
	Project Manager		Show exported variables...
	Close project		Show all objects...
	Exit Engineering Tool		Show selected objects...
Edit	**Undo**		
	Program		Breakpoint list
	Insert above		**OPC Item List**
	Insert below	Options	Print
	Insert next level		Long state line
	Expand		Set system time
	Full expand		Color setting...
	Compress		**Lock/Unlock UFB-Class**
	Cut		**Lock/Unlock Tag Type**
	Copy		No initial variable/tag-filter
	Paste		
	Delete		Auto accept

Export block...
Import block...
Import block as redundant...
Search...
Access rights (only on Security Lock)
User groups (only on Security Lock)
Display target stations

Help
Contents
Overview
About...

Project tree functions of commissioning

Menu options in bold can only be used during **Commissioning** and not during **Configuration**.

Project
Header...
Network...
Configuration
Project Manager
Close project
Exit Engineering Tool

Edit
Program
Expand
Full expand
Compress
Search!
Display target stations

Load
Whole station
Variables
Message configuration
Changed objects
Parameters...
Version informations...
Ignore version error...
Adjust version error...

System
Variable list
Tag list
Global message processing
Local message processing
Hardware structure
Show global variables...
Show exported variables...
Show all objects...
Show selected objects...
Break point list

Windows
Define debug windows...
Show value window
Show trend window

Options
Print
Set system time...
Long state line...
Color setting...
No initial variable/tag-filter
Auto Accept

Help
Contents
Overview
About...

5.3.4 State displays in the project tree

In contrast to configuration, state information is maintained on project objects during commissioning. This state information appears after each project object in brackets (like the object types) and is updated as it changes. The state information reflects the state of this object in the process station. If arrows should appear before the nodes, these project objects must first be loaded or reloaded into the station because of a configuration change. An exclamation mark indicates that other objects at levels below the one so marked need to be updated because of changes in their configuration.

The color of the node in the display provides information about the effects of its configuration change on other objects. Higher-level information may also need to be updated in the station.

In commissioning, **nodes** appear in the same formats as in configuration.

Refer to Display of project object states on page 78.

5.3.5 Representation of the side effects caused by configuration changes

Switching to configuration and carrying out a configuration change can affect the higher-level task or resource (side effects).

The configuration change must therefore be loaded into the higher-level resource.

After the plausibility check, all modified objects are always flagged with an arrow symbol next to the affected node in the project tree. If the project tree is compressed, an exclamation point shows that one or more affected nodes exist below the marked object. Refer to Changes with side effects on other project objects on page 243.

- Programs modified with no side effects are represented by a green node.
- Objects changed and having **side effects on the task** are represented by a yellow node.
- Objects changed and having **side effects on the resource** are represented by a red node.
- **Added objects** are shown along with the side effects produced.

- **Deleted objects** are flagged with an arrow next to the affected program list and task.

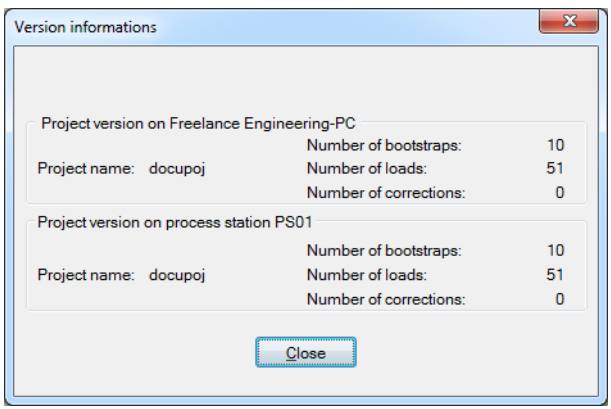
When side effects involve a task or even a resource object, all project objects below the affected object, together with the function blocks they use, are held up for the duration of the loading process. Refer to Load changed objects on page 239.

5.3.6 Project version

Version check

The following version control check is performed by Freelance Engineering to check which project is loaded in the selected station and whether or not this project corresponds to the project currently opened in Freelance Engineering. Information on the individual project versions appear as follows.

> Load > Version informations...



versioninfo_us.png

The information can be deduced from the number of times the function blocks and HW components have been loaded (corrections) and undergone parameter changes.

The critical factor is whether the project name in the engineering station matches the one in the process station.

The true project version number is stored in system variables. These system variables hold the project version. They are overwritten with current values on initialization or bootstrapping of the resource. The new values are then retained until the next initialization or bootstrapping (even in the case of a cold start).

Project name Name assigned to the project when it was set up.

Number of bootstraps CMajorVerNo
Incremented by 1 on each bootstrapping.

Number of loads CMinoVerNo
Is set back to zero on initialization or bootstrapping. After each successful loading of an object into the station it is incremented by one, but is not incremented for a cold start.

Number of corrections CPatchVerNo
Variable is incremented by one after each successful correction of the block parameters.

Adjust version error

If the system bus connection is interrupted during a load operation, a version error may occasionally occur. In this case the number of load operations in the version data differs between the configuration PC and the station by exactly one. This version error can be reset as follows:

Adjust version error should only be used to reset version errors when there is no doubt that the objects in the station are identical with those specified in the configuration.

> Load > Adjust version error...

Show global variables

In an AC 900F resource, a maximum of e.g. 32 Kbytes of memory can be allocated for variables. In the case of real values 4 bytes in size, this equates to 8,000 variables per resource.

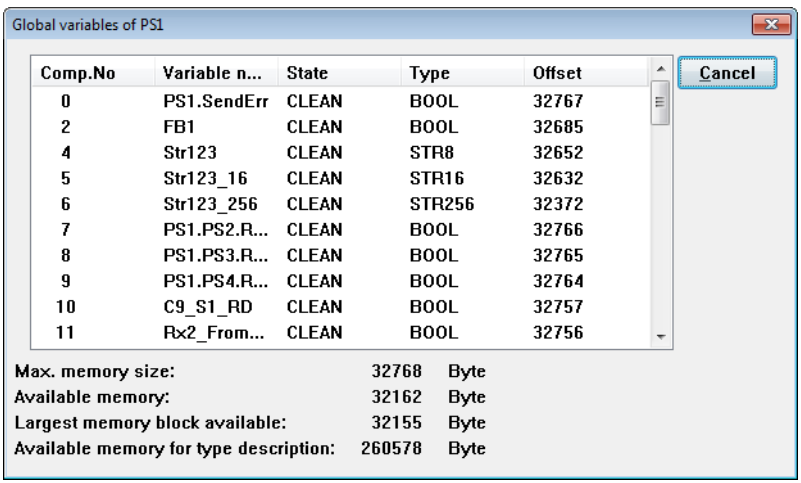
> System > Show global variables...

Or

Select **resource > Project > Header > Show...**

Or

Double-click on the **resource > Show...**



di1525_us.png

Comp. No. Sequence number of variable

Variable name Variable label with up to 16 characters in length

State

CLEAN	The object is correct and loaded into the station.
DIRTY	The object version in the engineering station does not match the object version in the process station.
CREATE	Object not yet loaded into station.
DELETE	Object deleted from project database, but still present in station.

Type	REAL, BOOL, UINT and so on
Offset	Memory address offset

Exported variables of the resource

A variable can be accessed for reading and writing. These operations can be performed within a resource. Other resources have read-only access. Furthermore, they can read a variable only if it is defined for **Export**, that is, if when the variable was originally defined, **Export** permitted to other resources was indicated by a Yes entered in column X of the variable's entry in the list. This state may be changed subsequently.

A maximum of 1400 bytes per resource can be specified for lateral communication (that is, for export from one resource to another).

Where variables are assigned to an I/O component, these or the I/O component itself cannot be written to from other resources. For more information, refer to ***Engineering Manual IEC 61131-3 Programming***.

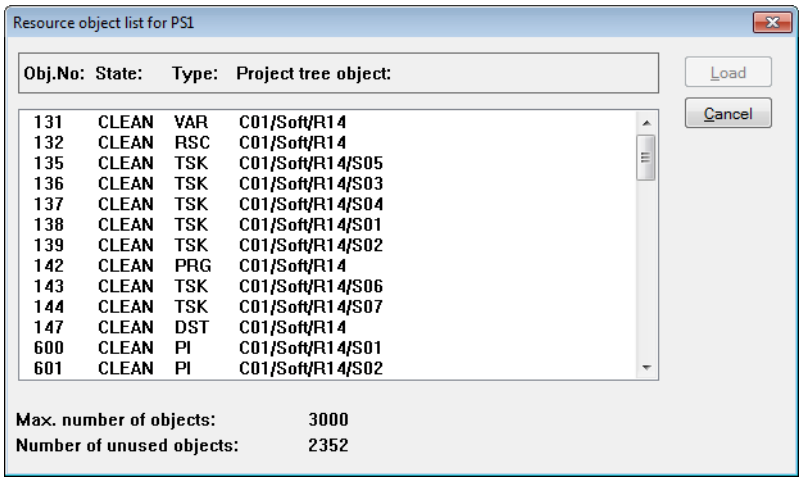
> System > Show exported variables...

The window setup is identical to that of the list of all global variables of the resource. For more information, refer to Show global variables on page 228.

Show all objects

All objects configured are shown for the project.

> System > Show all objects...



di1517_us.png

Obj. No	Object number	
State	CLEAN	The object is correct and has been loaded into the station.
	DIRTY	The object version in the engineering station does not match the object version in the station.
	CREATE	Object not yet loaded into station.
	DELETE	Object deleted from project database, but still present in station.
Type	VAR	Variable block
	TSK	Task
	CLS	Nonresident function-block class
	PI	Process image block
	PRG	FBD, LD, IL or ST program
	SFC	SFC program
	FB	Function block
	DEL	Undefined object
	VRS	Version info
	ACC	Access rights
	CODE	Program code
	ICON	Selection icon in the overview display
	GLBL	Global object
	RSC	Resource object

DST	Daylight saving time table
INT	Object for internal process station management

Project tree object
Path in the project tree

Max. number of objects
Number of objects possible in a project

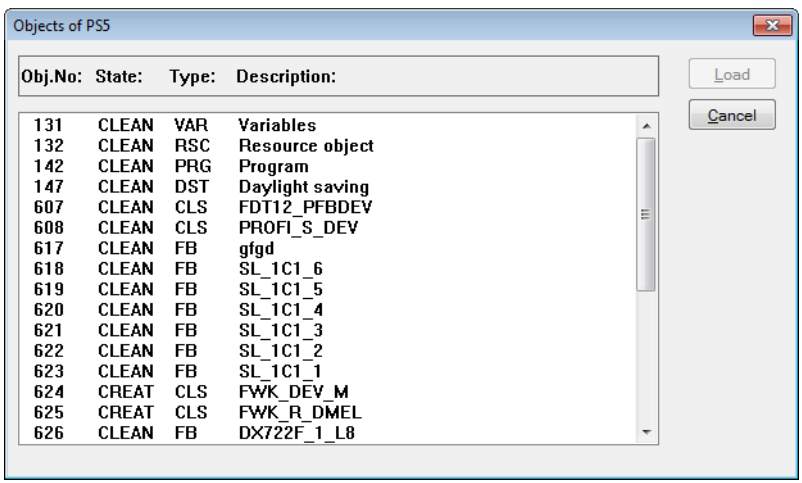
Number of unused *objects*
Number of additional objects possible

Load	Load all selected objects, this option is to be used only by L3 support personnel.
Cancel	Exit from the object list

Show selected objects

Only the objects below a project object selected appear in the project tree.

> System > Show selected objects...



di1518_us.png

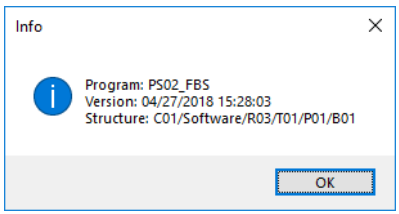
The setup of this dialog box is identical to that of the list of all objects. The only difference is that instead of the project tree path, a brief description of the object is

shown. This description is generated in a fixed form by the system. For example, a function block which is included in a graphic image is involved, then the tag name appears.

For more information, refer to Show all objects on page 229.

Program information

> Open the program > **Options** > **Version...**



di1430_us.png

Program	Program name
Version	Date and time of last saved change
Structure	Path of program in project tree

The name of the opened program, the date of the last change and the current path in the project tree can be found in a program's info window. For more information on project paths, refer to Section 3, Project tree.

5.3.7 Breakpoint list

Breakpoints set in the project are managed in the breakpoint list. The breakpoint list is part of the debugger. Refer to ***Engineering Manual, IEC 61131-3 Programming, Debugger***.

5.4 Time settings

5.4.1 System time, local time and time zone

SystemTime Current time kept by the station, used for internal transfer time stamps. System time is equal to **G**reenwich **M**ean **T**ime GMT (UTC).

LocalTime In addition to **SystemTime**, for each station a **LocalTime is** defined. The local time takes account of time zones as follows:

LocalTime = SystemTime - TimeZone
The default setting, a time zone offset of 0h, is Greenwich Mean Time (GMT). The relation of the local time to GMT is always set upon installation. In the station the local time is available in the system variable **name.DateTime** (name = resource).

ActualTime The local time with any daylight saving time shift applied is the **current time** (that is, the time which the user sees on his or her watch). This time is used when setting the Freelance time-of-day in commissioning and is generally the time used at the Freelance user interface.

TimeZone The time zone (TZ) is calculated from the difference between the Greenwich Mean Time (GMT) setting and the local time (TZ = GMT - Local Time). For Germany TZ is equal to **–1** (at 13:00 GMT it is 14:00 in Germany, 13 – 14 = –1).

Set time zone

The time zone is set for the entire project in the project tree configuration node.

Synchronize system time

All Freelance stations are equipped with a real-time clock. When the system is running, the clock times must correspond throughout the system, so that for example entries can be made in logs and trends in the correct time sequence. For this to occur, the system time must be synchronized at all stations. A system master (a process station) leads all the other stations physically connected to an Ethernet bus during the synchronization process.

Generally the resource or station with the lowest station number is the time master. This station synchronizes the clocks of all other stations. Station number settings can be checked under **System > Hardware structure > Network...**.

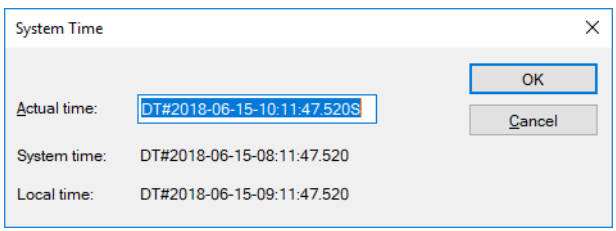
The leading station (master) synchronizes all other stations (slaves) once per minute. Discrepancies of less than one second are equalized by adjusting the system clock. In the case of greater deviations, the time is set by means of a time jump. An accuracy of approximately 3 ms can be achieved, but discrepancies of up to 5 ms are quite possible.

System time synchronizations only occurs if the clock time has been set at least once by the engineering station.

Set system time

The system time can be set system-wide from the engineering station.

> Options > Set system time...



di1537_us.png

The current Freelance time is shown in the **Current Time** field. Here the user can enter the correct time. If the current time is daylight saving time, this is indicated by an “S” following the time value.

The system time is calculated from the current time set during commissioning and is loaded in the time-leading station (master):

Local time = actual time – daylight saving time shift

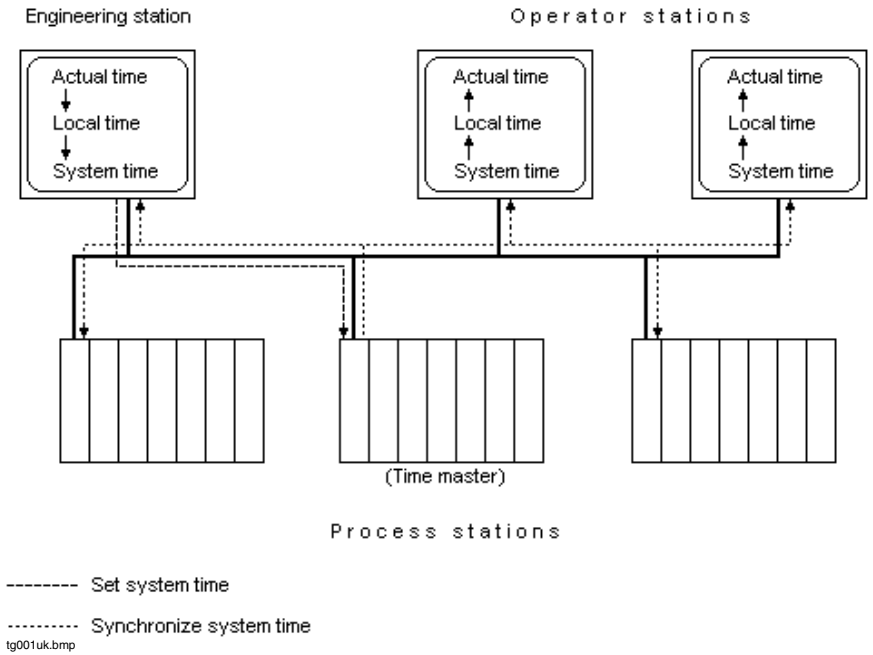
System Time = Local time + Time Zone

Example:

TimeZone (TZ)= -2, daylight saving time shift = 1h:

Current time = 14:00S, LocalTime = 13:00, SystemTime = 11:00

This new system time is transferred to all other resources (Slaves) through **system time synchronization**. These resources then convert the new system time to each of the individual local time.



The system variable **Name.DateTime** (Name = resource) exists in each process station. The local time used for time-controlled events is obtained from this variable. If this variable appears in the value window, the actual variable content is shown. Even when using daylight saving time, the daylight saving time shift is never added here.

Writing DT variables

Activation of daylight saving time only effects the display of time points. When operating on a variable, the user must specify whether the edited time is a daylight saving time or not. A daylight saving time must be identified by an “S” following it. If this “S” is missing, the time value input is interpreted as local time. If an “S” is

specified for a time when daylight time is not in effect, a message is sent to the user requesting a correction.

Example: Input "..16:00.." produces 16:00 at the station; an input of "..16:00..S" produces (daylight saving time in Germany) 15:00.

If the user enters a DT value in Freelance Operations for operation, this value is interpreted as DST value.

Time zone

The time zone is set for the entire project in the project-tree configuration node.

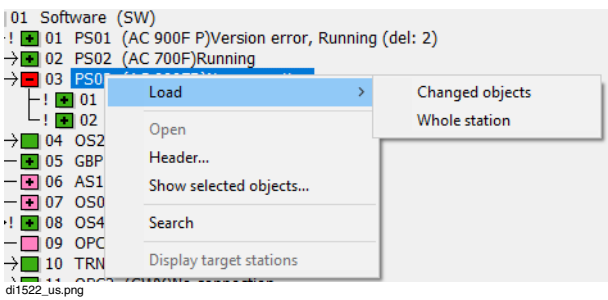
5.5 Loading the project

During loading, the user program, or parts of the user program, are loaded into the process station.In this procedure, either the **Whole station** can be loaded, or only the **Changed objects**.

Changes to **Variables** and **Message configuration** can be loaded individually into the station, but also changes to individual **objects**.

With **Load > Changed objects** only changed objects or program blocks (including individual function blocks) are loaded.

When the system is started, all changes are brought up to date through **Changed objects**. As a result, care must be taken in case of **Side effects** on D-PS resources and tasks. The side effects are indicated in the project tree by showing the affected nodes in different colors. For more information, refer to Display of project object states on page 78.



di1522_us.png

5.5.1 Load whole station

All parts of the user program associated with the selected resource are loaded.

This is achieved by deleting the user program from the resource. The output modules and field devices assume their configured safety values. This may mean: retaining the last value or assuming a fixed value, can be set channel by channel.

If when loading the whole station it becomes clear that a different project is being loaded, the I/O modules are also initialized. The output modules output zero-voltage/current in this process.

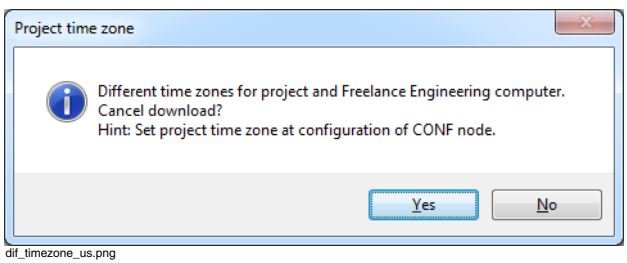
If no operating system has been loaded in the resource at the time of loading the whole station, the station's operating system is installed automatically with initial loading.

When there are field devices connected (PROFIBUS), only the user parameters are loaded. DPV-1 parameters must be loaded manually. Refer to ***Engineering Reference Manual Communications and Fieldbusses,*** PROFIBUS.

> Load > Whole station

Regardless of which project object within the resource is selected, on Load > Whole station the complete station (D-PS, D-OS or GWY) is deleted and reloaded. Ensure that this resource belongs to the loaded project. The result of the system's version check is shown without interrupting the loading operation.

If the time zone of the project does not match with the Freelance Engineering PC, a dialog box which draws attention to the discrepancy appears.



dif_timezone_us.png

Yes The download operation is canceled.

No The download operation is continued.

5.5.2 Load variables

Freelance includes both predefined and user-defined variables. Predefined variables are set automatically in the resource and created in the station resource at the time of bootstrapping.

User-defined variables are those variables defined by the user, either in the programs or added directly to the variable list.

Loading of variables entails initialization of the variables concerned. This process erases all previous information concerning the variable. Programs running in other objects which rely on information from the variables being loaded can thus react in ways which, under certain circumstances, may be undesirable. For example, the export attribute of a variable is changed, then this variable will be reloaded and its value will be reset to the initial value. It is important to ensure that the loading of changed objects does not have a negative influence on the process being controlled.

> Load > Variables

All variables belonging to the selected resource, including exportable variables, are loaded into the station. The size of the global variable area in the resource is adjusted accordingly.

Load variables applies only to user-defined variables, not to this resource's system variables.

All variables are entered in the variable list. No variable may be allocated to more than one resource. For more information, refer to ***Engineering Manual IEC 61131-3 Programming***.

5.5.3 Load message configuration

The station-specific (local) message processing configuration is loaded into the selected station. This includes specifications of message and acknowledgment handling, audible warning (horn) control and relating to the message list and message line.

> Load > Message configuration

Refer to ***Engineering Manual Operator Station, Messages and Hints***.

5.5.4 Load changed objects

The smallest loadable unit of a project is the object. There are visible project objects, such as the resource, task, program list and program, and there are hidden objects such as the process image, function block, variables and messages.

When **Load > Changed objects** is executed, only the project's changed objects are loaded. Care must be taken in case of side effects on the higher-level task or resource.

Loading of objects entails initialization of the objects concerned. This process erases all previous information concerning the object. Programs running in other objects which rely on information from the objects being loaded can react in ways which, under certain circumstances, may be undesirable. For example, the export attribute of a variable is changed, then this variable will be reloaded and its value will be reset to the initial value. Make sure that the loading of changed objects does not have a negative influence on the process being controlled.

When only object deletions have occurred, **Load > Changed objects** will delete the objects on the station only. When changed or selected objects of a resource are loaded without selecting the resource, check whether all function block classes (domain classes) of the resource have CLEAN status. If there is a domain class that does not have CLEAN status, then a message box appears recommending loading at the resource level:

```
Changed function block classes have to be downloaded. Please
select resource node and load changed object first here.
```

The download below the resource level is not carried out. When loading individual domains of a resource, check independently of the object selected.

All Objects not being CLEAN or DELETE will be initialized when loaded!

If after a project change only the version information of an OPC or other gateway has changed, the version information will be automatically loaded into this gateway when loading a process station.

When there are field devices connected (PROFIBUS), only the user parameters are loaded. DPV-1 parameters must be loaded manually. Refer to ***Engineering Reference Manual Communications and Fieldbusses,*** PROFIBUS.

> Load > Changed objects

5.5.5 Uploading current block parameters

The current working data of a block can be modified by manual entry from the commissioning process, or from Freelance Operations through a gateway or other server. The changed parameters will be lost if a cold start is performed or if a download is used for reconfiguration.

Individual values can be modified in the commissioning phase in such a way that they are not lost during a cold start by using the **Correct** function.

Larger amounts of working data can be saved using the **Upload parameters** function.

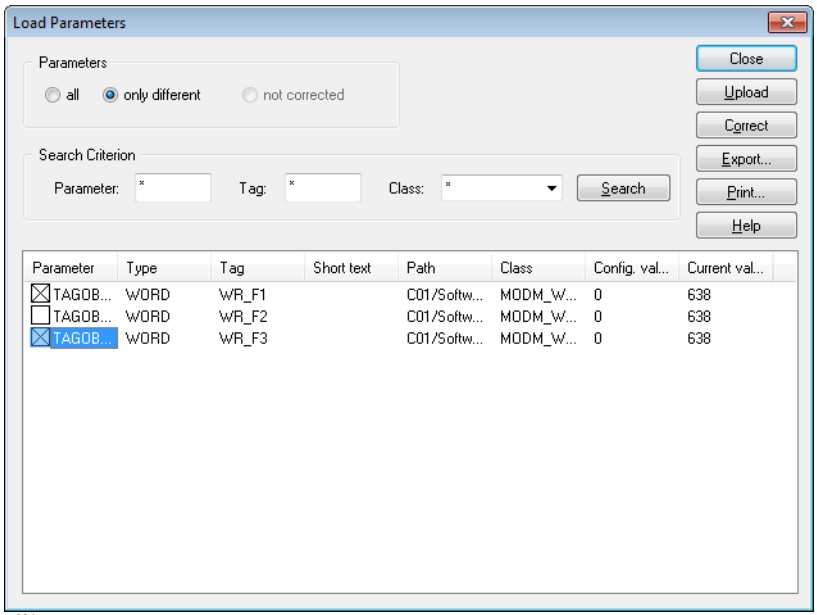
Procedure

- Select a station or a section of the project tree.

> Load > Parameters...

- All the block parameters below the selected project tree node are read and appears in a list.
- Each parameter is shown with the following information: parameter name, data type, tag name, short text, project path, block class, configured value, and current value.
- The list can be sorted on any one of these fields (by pressing the appropriate column button).

At the beginning of each line is a check box that is used for specifying whether or not the parameter is to be corrected.



tg004_us.png

The contents of the list display can also be modified.

Parameters

all All parameters are shown under the selected project tree node.

only different Only those parameters are listed whose current value is different from that contained in the database (standard setting).

not corrected (Select only after a correction process is completed). Those parameters that should have been corrected but where the process to do so failed are listed. On completion of a correct process this radio button is selected automatically.

Search criteria Another method of modifying the parameter list is by using search criteria:

- Parameter
- Tag name
- Class name

Each of these criteria may include the wildcards ‘*’ and ‘?’.

Class can also be selected from a list box.

After the **Search** button is pressed, those parameters are listed which meet the search criteria entered. Search criteria only take effect when radio buttons ***all*** and ***only different*** are selected.

Correction identification (check box) for a marked block can be completely activated or deactivated with the SPACEBAR.

Close The dialog is closed.

Upload ALL parameters are re-read from the working data and the list is regenerated using the search criteria set.

Correct The current values of the selected parameters in the list are copied to the project database.

All selected parameters are copied to the database regardless of any active search criteria!

Export The parameters currently listed - which depend on the active search criteria - are transferred to a file <name>.upl. A prompt to enter the file name follows.

Print The parameters currently listed - which depend on the active search criteria - are printed out.

Help A language-specific online help is opened to give a description of the module parameter.

5.5.6 Delete project from the process station

The station can be erased in various methods. For this the D-PS resource should be selected in commissioning through.

> Select **Resource** > **Project** > **Header...**

Or

> Double-click on **Resource**

With Initialize all, everything is deleted, even the operating system. **Initialize** the resource deletes the complete user program from the station, and in contrast to cold

start, also sets the output modules to zero voltage/current. Although all current data and variable values are deleted in **Cold start**, the user program, however, remains unaffected and the output module channels assume their configured safety values.

In **Load > Whole station** the station is likewise first erased.

Refer to Load whole station on page 237.

5.5.7 Ignore version error

If the version number of the project currently open does not match that of the software loaded on the station, a version error will result.

If loading is rejected, this version error can be bypassed by selecting **Load > Whole station**.

If after a project change only the version information of an OPC or other gateways has changed, in version 6.1 and later the version information will be automatically loaded into this gateway when loading a process station.

Refer to Adjust version error on page 227 and Version check on page 226.

5.6 Changes with side effects on other project objects

The possible side effects of a configuration change can be classified in three groups:

- **No side effects:** The objects involved in the configuration change can be loaded into the task or resource without interrupting them.
- **Side effects on the task:** The objects involved in the configuration change can only be loaded after the task has been stopped. For example, a change in a program also causes a change in the process image of a task. The program can only be restarted when both the program and the process image of the task have been updated.
- **Side effects on a resource:** The objects involved in the configuration change can only be loaded after the resource has been stopped. For example, a change in a program also causes a change of the global resource variables. The program can only be restarted when both the program and the global variables of the resource have been updated.

5.6.1 Display of changes in the project tree

After the plausibility check, all modified program blocks are always flagged with an arrow symbol next to the affected node. If the project tree is compressed, an exclamation point shows that one or more affected nodes exist below the marked object.

- Programs modified with no side effects are represented by a green node.
- Objects changed and having **side effects on the task** are represented by a yellow node.
- Objects changed and having a **side effect on the resource** are represented by a red node.

The indication of changes will be passed from the affected object in the project tree to the resource node. Only the most far-reaching side effect appears.

Refer to Display of project object states on page 78.

5.6.2 Changes with no side effects

- Adding functions and function blocks only
- Adding functions and function blocks with links to variables in those programs
- Adding a new task (or moving one out of the pool)
- Deleting functions or function blocks
- Deleting a task (or moving one to the pool)
- Moving tasks from one resource to another
- Changing task parameters such as autostart or priority
- Changing task cycle time
- Changing the program list parameter **on/off**
- Changing function block parameters
- Modifying the processing sequence for project objects at a level lower than the task

- Changing the data type of a variable
- Adding a new variable where the access is not via the process image mode
- Entering new process image variable, but not used in program (for example connect with FBD)

If after a project change only the version information of an OPC or other gateway has changed, the version information will be automatically loaded into this gateway when loading a process station.

5.6.3 Changes with side effects on the task

- Deleting programs or program lists (or moving them to the pool)
- Adding programs or program lists with process image variables (or moving them out of the pool)
- Moving programs or program lists from one resource to another
- Adding functions or function blocks with links to process image variables
- Adding and use new process image variables
- Modifying process image variables which are used in a program
- Changing, deleting or assigning to a variable of I/O components

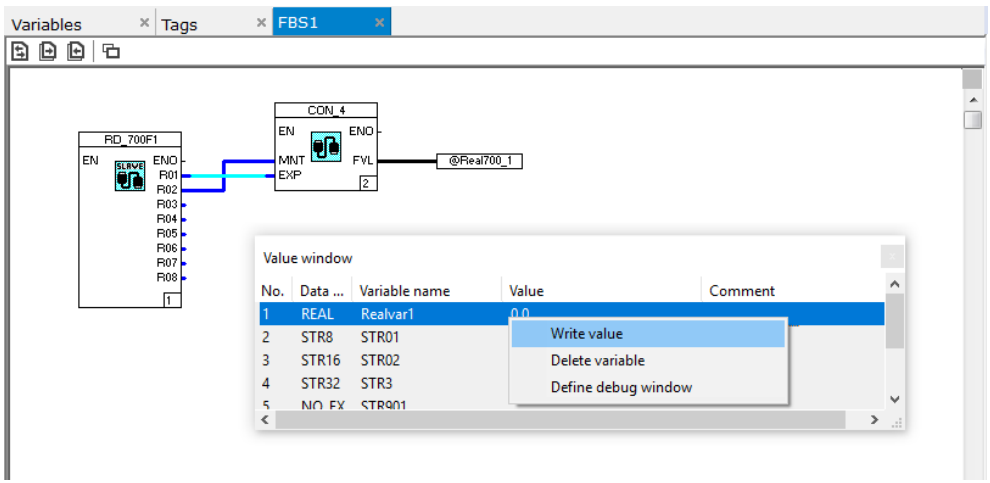
5.6.4 Changes with side effects on the resource

- Loading Firmware (Update/Upgrade)
- Use of an **Export variable** from another resource (which leads to intercommunication between resources)
- Adding or deleting of I/O components
- Changing controller boot parameter
-

5.7 Displaying and writing values in a window

During commissioning, it is important to carry out online checks on parameter changes during processing.

The commissioning window makes it possible to observe a variable. This enables the user to intervene in an online process from the engineering station.



di1536_us.png

Variable window

Define debug windows

Settings established for debug and trend windows

Show value window

Show trend window

Make it possible to show the process signals and variables requested in the form of values and trends

Enter variable Variables are introduced into the value or trend window

Write value Enables the one-time write of a binary value

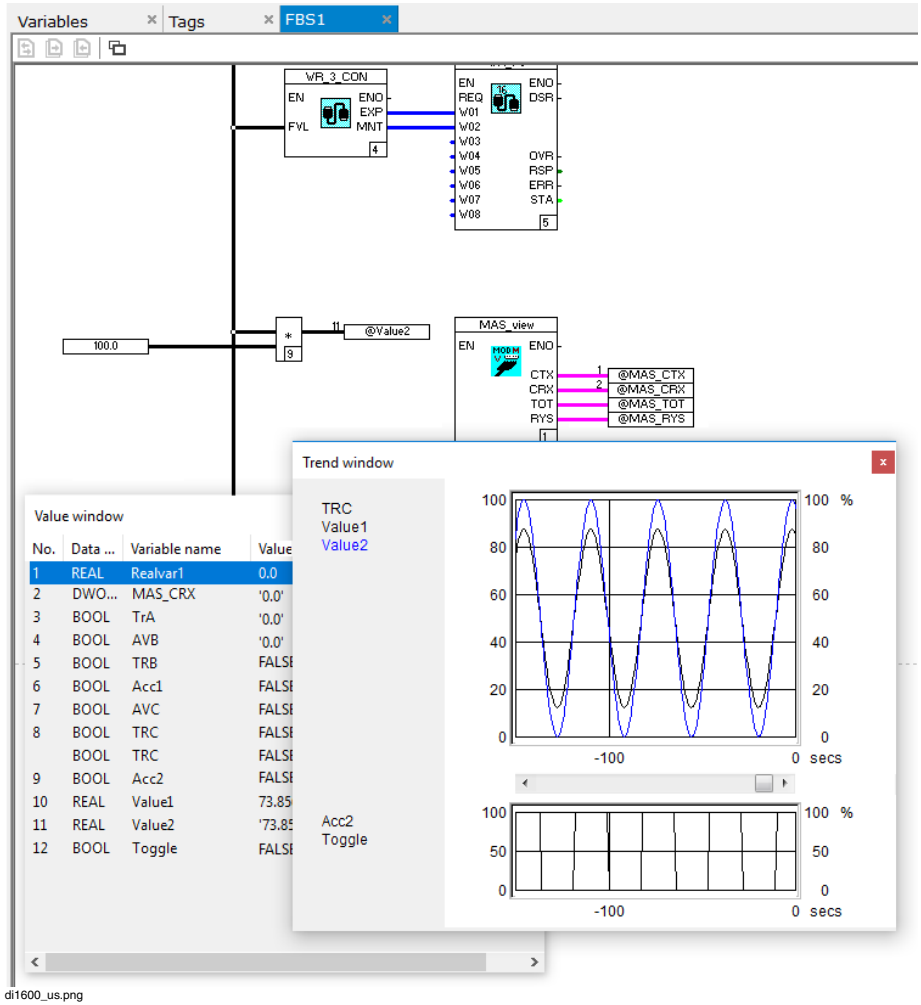
The value window can show numeric data - which can also be shown as an option on a trend window - and strings.

The values for the value and trend windows are updated once per second.

Both windows can be appeared simultaneously. They remain on screen when switching programs.

This enables variables from other programs to be inspected in conjunction with the program being viewed.

If the controller download is disabled (Load Disabled), online values can not be displayed, unless the variables have already been defined in the Value or Trend window before locking the controller. These previously defined variables can also be written (write value).



di1600_us.png

Depending on the application from which it has been started, the **Window** menu opens up a variety of possibilities.

	Project tree	Variables list	Tag list	Program
Define debug window	•	•	•	•
Show value window	•	•	•	•
Show trend window	•	•	•	•
Enter variable	------	•	------	•
Write value	------	•	------	•
Processing On/Off	------	------	•	------

• selection can be made from corresponding menu
---- selection cannot be made from corresponding menu.

5.7.1 Enter variable

The menu command **Enter variable** can be selected from within the instruction list, the ladder diagram, the variable list, the structured text and the function block diagram (FBD).

> Click on variable > **Windows** > **Enter variable...** > **Define debug windows...**

The selected variable is added to the variable list in the commissioning window. The **Define debug windows** dialog box appears on the screen and the variable can be assigned a display format.

As soon as a variable is entered, it has a sequence number appended to it in the instruction list and in the FBD. This number reflects the order in which variables are entered.

In FBD and LD it is possible to show signal-line values in a value window and/or a trend window. Signal-line values are deleted from the value window when exiting

the FBD or LD program. Signal-line values have no names in the value window. They can be identified only by sequence number.

In ST it is possible to show local variables in the value window and/or trend window. Values of local variables are deleted from the value window when exiting the ST program.

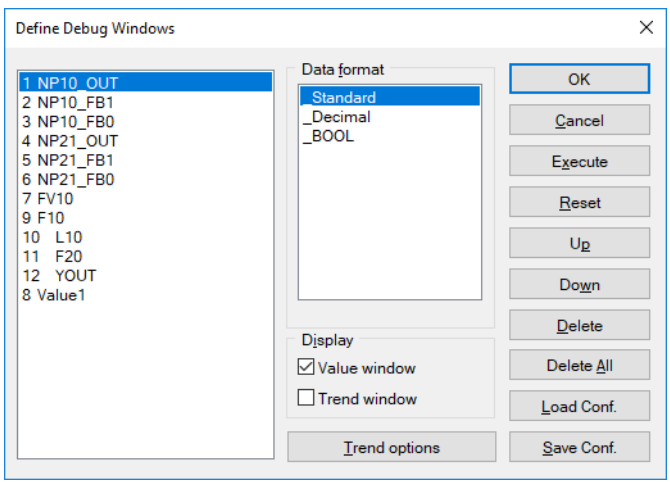
> Click on variable > **Windows** > **Enter variable...** > **Define debug windows...**

In the FBD, double-click on a variable name or the signal line to enter a variable value.

Define variable windows

This part of the program is where a variable is allocated a display format. The variable to be processed is selected in the list box. The display format for the **Value window** and **Trend window** can be selected. Entries can be added to the variable list by opening the menu item **Enter variable** or directly by double-click on the variable.

These entries may be saved and reloaded.



di1650_us.png

Click on variable > **Windows** > **Define debug windows...**

OK	**Save entries** and exit from **Define debug windows**.
Cancel	Exit from **Define debug windows** without saving entries
Execute	Activate entries in the value and trend windows without exiting **Define debug windows**.
Reset	Cancel the last entry.
Up	Shift the selected variable **up one position in the list**. The variable's sequence number remains unchanged.
Down	Shift the selected variable **down one position in the list**. The variable's sequence number remains unchanged.
Delete	Delete the selected variable from the variable list in the Value and Trend windows.
Delete all	Delete all variables from the Trend and Value windows.
Load conf.	Load a stored configuration, selected from a list.
Save conf.	Save the current configuration, (followed by a prompt for a name under which to save it).
Data format	Display format for selected variables in the value window. After the required variables have been selected, all the valid data format options appear here on the screen. An appropriate display format for a variable can be chosen by clicking on the relevant format. In the value window, a variable can appear simultaneously in all the data formats offered. (Displaying a **variable of data type DT** in a format that differentiates daylight-saving time: if the value of the variable falls within daylight-saving time, then this value is increased by the daylight-saving time difference (1 hour in Germany), and the value is flagged by appending an 'S').
Display	Determine the type of display of the variable. The variable can appear in the Value window and/or Trend window. Select the appropriate check box.
Value window	☑ Variable will be shown in the Value window; data display formats available for the variable will be listed.

☐ Variable will not be shown in the Value window.

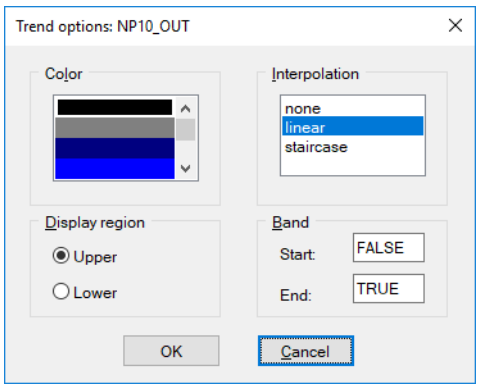
Trend window ☑ Variable will be shown in the Trend window, the **Trend options** menu will be opened.

☐ The variable will not be shown in the Trend window.

Variables of the data type string and date (DT) cannot be shown in the Trend window.

Trend options **Trend options** dialog box is opened.

Trend options



di1651_us.png

Click on variable > **Windows** > **Define debug windows...** > **Trend options**

Color Choice of color for display of selected variables in the Trend window.

Interpolation Three different interpolation methods are available: None, linear and staircase.

Display region

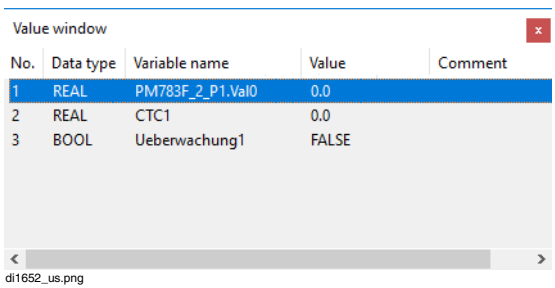
Choose one of the two variable windows, in which the selected variable should appear. The Y-axis of both windows extends from 0 to 100%. Each X-axis covers 150 values, (that is, seconds, since recording in the process station is fixed at one reading per second).

Both Trend windows appear simultaneously on the screen. The Trend window will display up to six variables in both Upper and Lower window.

A maximum of 40 variables can be selected for display. Up to 6 of these can be shown in the trend window.

Band The limits are entered for the display of selected variables in the Trend window. Data must be entered in an appropriate format for the variable.

5.7.2 Show value window



di1652_us.png

Click on variable > **Windows** > **Show value window**

The current values of variables are shown in the Value window dialog box. Each variable is shown in a format depending on its data type. Variables of the BOOL data type are shown with the logical state “True” or “False”. Variables of the data type REAL, INT, TIME and WORD are shown with their value, and variables of the STRING data type are shown with their text content.

The Value window consists of five columns. From left to right on the screen, they have the following function:

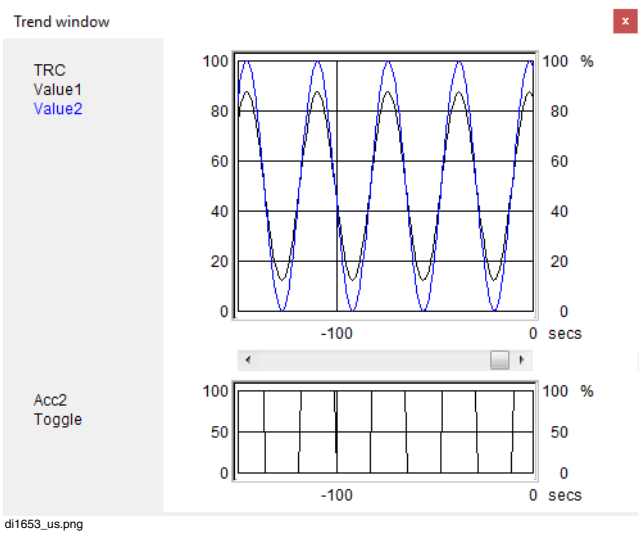
- Input (sequence) *No.*, from Define debug windows
- *Data type*
- *Variable name*
- *Value*

- *Comment*

The setting of **UseDaylightSavingTime** is ignored for the display of DT variables in the value window. In commissioning mode, the user can always see the actual variable contents. (UseDaylightSavingTime is only taken into account in dialogs for setting the clock under Freelance Engineering).

5.7.3 Show trend window

Click on variable > **Windows** > **Show trend window**



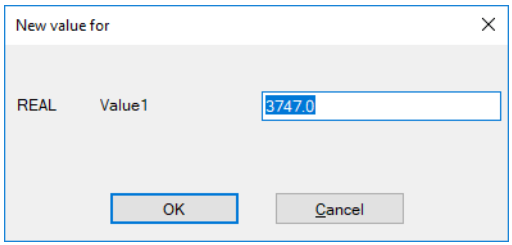
di1653_us.png

In the Trend window, process signals covering a time span of 1,000 seconds can be displayed. At any one time, a time span of 150 seconds is visible. The trend window has no “memory”: upon leaving the trend window, the values recorded are lost.

In the lower border of the window a push button is used for scrolling back the display. By activating this button it is possible to see earlier signal sequences.

5.7.4 Writing a value

Click on variable > **Windows** > **Write value...**



di1655_us.png

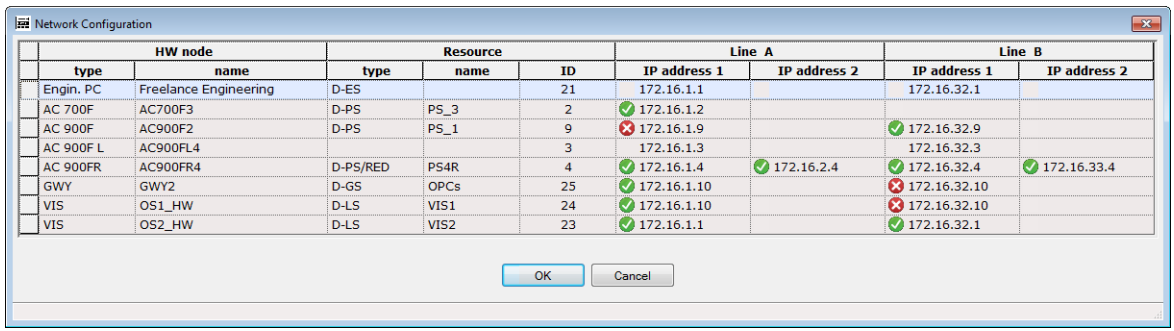
Write value can be used to assign a new value to a variable. When making such an entry, make sure that the entry is in the correct data format (as defined for the variable).

When **Write value** is opened, the dialog box is displayed in the middle of the screen. If there is a Value or Trend window dialog box already displayed, the **Write value** dialog box will be located behind the Value or Trend window dialog box. The Value or Trend window dialog box must be repositioned or closed before the Write value dialog box can be used.

The variable is overwritten only for one cycle. If, in the next program cycle the variable is rewritten, the value specified here, in commissioning, is overwritten. To force a value, the variable must be forced at the I/O module.

5.8 Control Net status information

The availability of the process stations and operator or gateway stations on the Control Net is displayed in online mode in the hardware structure graphic display and in the network configuration. Only the active components are checked for availability. Note that the display update may take up to 45s, independent of the set switch-over time.

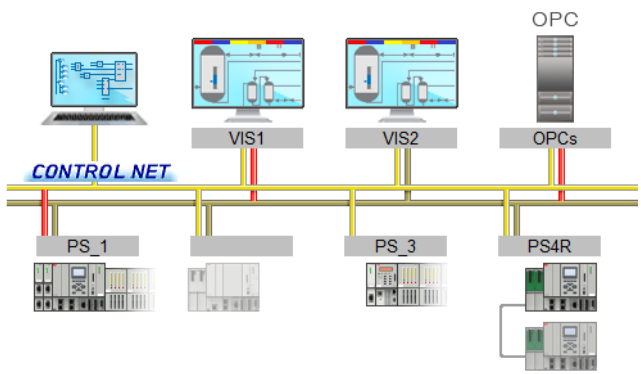


HW node		Resource			Line A		Line B	
type	name	type	name	ID	IP address 1	IP address 2	IP address 1	IP address 2
Engin. PC	Freelance Engineering	D-ES		21	172.16.1.1		172.16.32.1	
AC 700F	AC700F3	D-PS	PS_3	2	172.16.1.2			
AC 900F	AC900F2	D-PS	PS_1	9	172.16.1.9		172.16.32.9	
AC 900F L	AC900FL4			3	172.16.1.3		172.16.32.3	
AC 900FR	AC900FR4	D-PS/RED	PS4R	4	172.16.1.4	172.16.2.4	172.16.32.4	172.16.33.4
GWY	GWY2	D-GS	OPCs	25	172.16.1.10		172.16.32.10	
VIS	OS1_HW	D-LS	VIS1	24	172.16.1.10		172.16.32.10	
VIS	OS2_HW	D-LS	VIS2	23	172.16.1.1		172.16.32.1	

NetworkConfigurationOnline_us.png

5.8.1 Hardware structure

The availability of a resource is indicated by a color change of the bus connection. In the event of a failure, the bus connection is shown in red. No color change occurs for inactive stations or for stations without Control Net redundancy.



HWGraViewOnline_us.png

In the graphic view of the hardware structure non-redundant controllers or operator/gateway stations with redundant Control Net are displayed as follows:

Line A	Line B	Visualization
OK	OK	
OK	Error	
Error	Ok	
Error	Error	

HWGraViewOnlineA_us.png

In the graphic view of the hardware structure redundant controllers with redundant Control Net are displayed as follows:

Secundary Controller		Primary Controller			
	Line A	OK	OK	Error	Error
	Line B	OK	Error	OK	Error
Line A	Line B				
OK	OK				
OK	Error				
Error	Ok				
Error	Error				

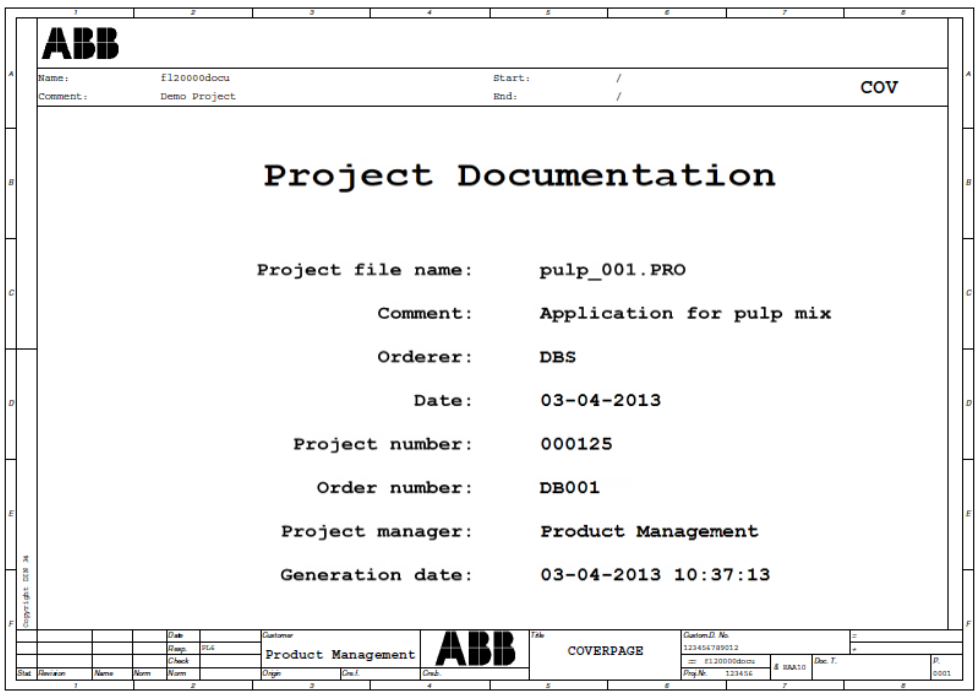
HWGraViewOnlineB_us.png

If a component should not be available on Line A or Line B, this might be due to a network or component failure. In this case, the network connection does not change its color. Unavailable process stations are represented in the hardware structure as a red bitmap.

6 Documentation

6.1 General description – Documentation

All configured objects can be documented from Freelance Engineering. Likewise, provision has been made to ensure easy location of individual documents as well as precise incorporation into an overall plant documentation.



ti009_us.png

A common layout is used for all documents. It is possible to incorporate the company logo into the document header or footer. Document handling is facilitated by a flexible sorting on the resulting documents and by a complete table of contents for all output project parts.

An unequivocal designation of the document type (Doc. Type) using the document identification key facilitates location of documents. For example, &EFT20 for electrical engineering, function chart and FBD program.

An automatically inferred documentation identification (Doc ID) refers to the functional environment (=) through the program name and to the location environment (+) through the cabinet identifier. Specification or output is also possible through the customer drawing number.

A cross-reference is found in the program sheet, while the cross-reference list constitutes a supplement to the program sheet for larger scopes, so that all references can be easily located. On showing variables, the allocated resource can always be recognized. If an input or output channel is allocated directly to a variable, the former is also displayed.

If possible, a tabulated presentation must be selected to reduce the quantity of paper used. Otherwise, presentations are made in dialog boxes, making it easier to read them.

In tabulated form:

- Object attributes of dynamic graphic displays
- Step and transition parameters of the sequential function chart
- Criteria window of the sequential function chart

In dialog box form:

- Function block parameters
- SFC parameters
- Module parameters
- Logs and trends
- Network adjustment

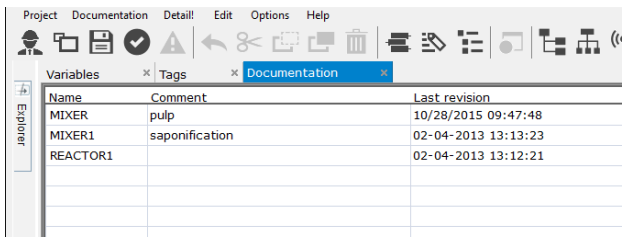
Documentation output is controlled by the print job, that is, a job must always be selected for printing. The contents of the print job are defined in such a job. Each print job is valid for all projects on the engineering station.

6.2 Documentation manager

6.2.1 Editing the documentation manager

The documentation manager menu can be accessed through:

Select Project tree > **Project** > **Documentation**



ti050_us.png

On entering to the documentation manager, the print jobs available are shown beneath the menu bar. The blank table lines represent the free print jobs.

The documentation jobs are structured in the following manner:

Name	Job name, maximum 12 characters, must begin with a letter, no special characters, no blanks.
Comment	Maximum 34 characters free text (special characters as well).
Last revision	The system automatically enters the date and time of the last change. Adaptation of the column width.

6.2.2 Changing the column width

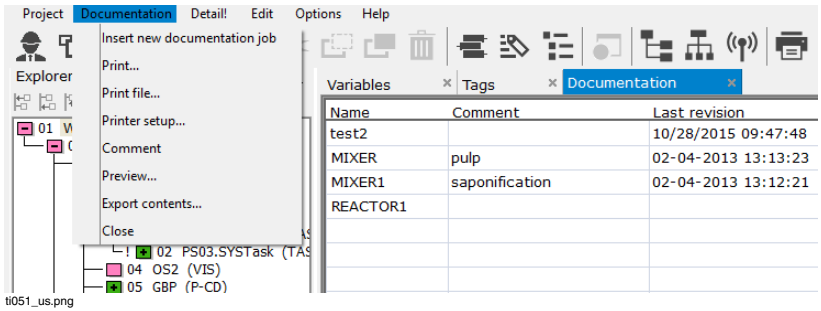
If the mouse pointer is placed on a column separation line, the mouse pointer changes to a horizontal arrow. Hold the mouse button and change the column width. This change is saved on quitting the document and is available for subsequent operations.

6.2.3 Menu structure of Documentation

Project	Save project	Edit	Field
	Save Tab		Delete field
	Documentation		Cut
	Check		Copy
	Check all		Paste
	Show error list		Delete
	Header...		Export block...
	Comment		Import block...
	Network...	Options	Print
	Commissioning		Sort fields...
	Project Manager		Project scope...
	Close project		Hardware scope...
	Exit Freelance Engineering		Notify print errors
Documentation	Insert new documentation job		Save column settings
	Print...		Auto Accept
	Print file...	Help	Contents
	Printer setup...		Overview
	Comment		About...
	Preview...		
	Export contents...		
	Close		
Detail!			

6.2.4 Documentation

> **Project > Documentation > Documentation**



Insert new documentation job

Select an empty line

> **Documentation > Insert new documentation job**

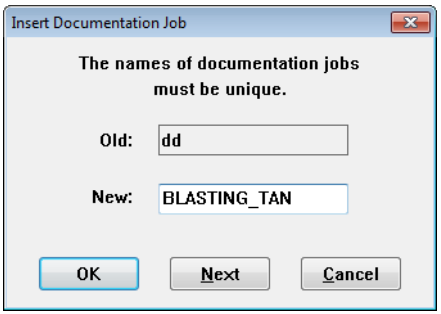
> Enter a documentation name into the field Name

Copy new documentation job

Select an existing line

> **Documentation > Insert new documentation job**

The contents of the selected documentation job are accepted for the new one. In the Insert documentation job dialog box the new name must be entered or be taken from the old name.



ti052_us.png

Also a new documentation job can be created by copying an old documentation job and save it under a new name. All definitions of the documentation contents are also copied and can be modified later.

Comment

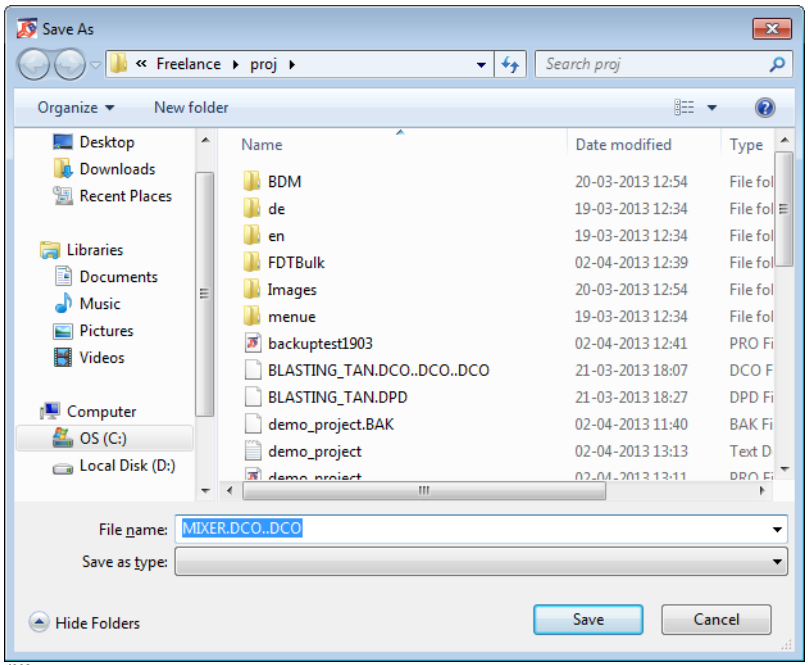
> **Project > Documentation > Documentation > Comment**

A long comment can be added here on the selected print job, describing the contents.

For more information, refer to Editing a project comment on page 59.

Export contents

> **Project > Documentation > Documentation > Export contents...**



ti063_us.png

The table of contents of the selected job is saved in a CSV format with file extension *.dco. The file name and directory can be selected in a Windows menu, with the project directory being presented as a default and the first eight characters of the documentation job name as file name. This file can be opened in table form with Excel, and worked on further.

Exit

> **Project > Documentation > Documentation > Close**

The documentation editor is closed.

6.3 Compilation of the drawing header/footer

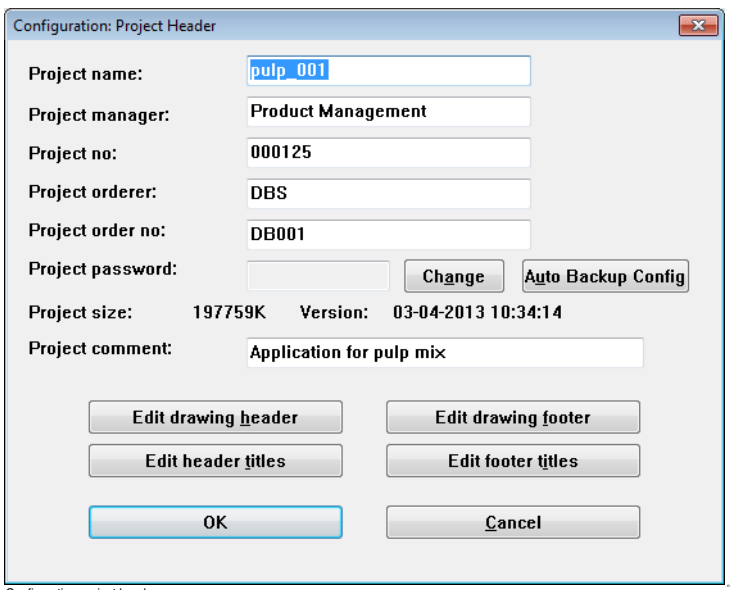
6.3.1 General information

To enter data into the drawing header/footer, the header and footer dialog box is divided into several horizontal levels, so as to enhance the clarity of layout.

6.3.2 Project-wide drawing header/footer

A drawing header/footer system-wide valid for the project is defined on the project level. All settings made in the header or footer are saved in the project.

From Project Manager, select **Project > Header...**

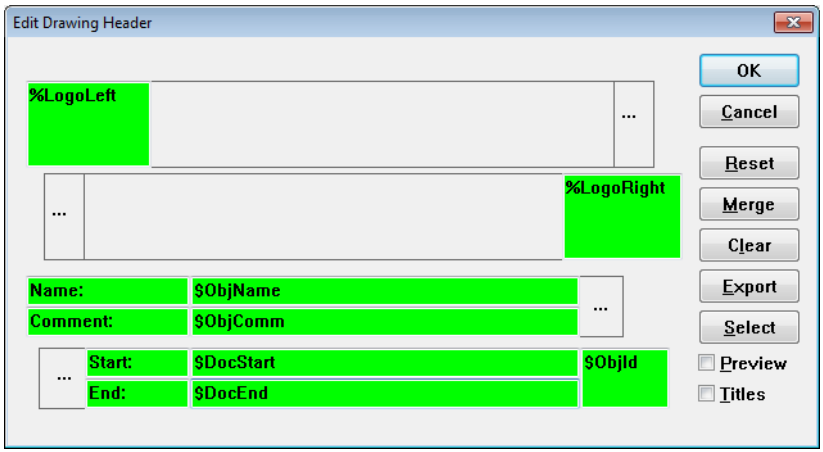


Configuration project header_us.png

The texts entered into the upper section of the dialog box are allocated to the drawing footer or header. Refer to General information of the project on page 42.

Editing the drawing header

From Project Manager, select **Project > Header... > Edit drawing header**



ti011_us.png

Ok Exit header entries, entries are saved.

Cancel Exit header entries, entries are rejected.

Reset All entries are accepted by means of the defaults from the file FRAMES.INI.

Merge All variables are accepted with the defaults from the file FRAMES.INI, that is, all fixed texts are preserved.

Clear All field contents are deleted.

Export The current entries of the header are written into the file FRAMES.INI and are hence the new defaults for **Reset** and **Merge**

By using **Clear** and then **Export** you will lose all entries!

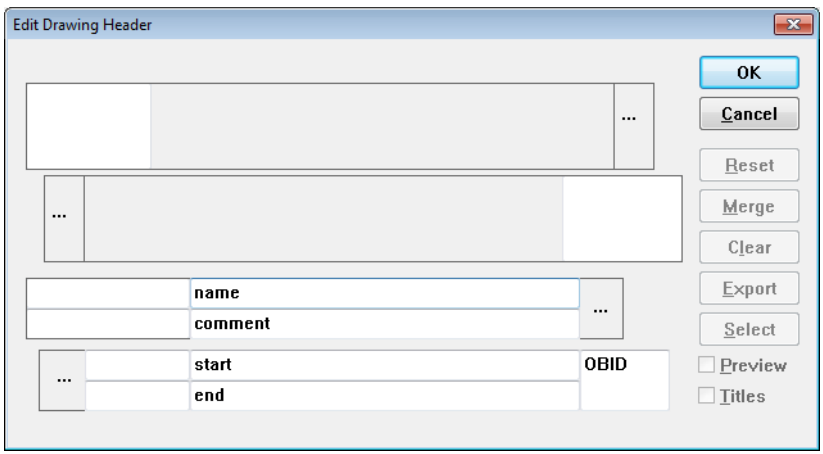
Select From the list field of the FRAMES.INI file a section can be selected. Its contents or default settings are then activated, when **Reset** or **Merge** is pressed.

Preview A preview showing the contents as they are printed is superimposed. The appearance of inserted bitmaps can be evaluated.

Titles The titles defined by the user are superimposed, giving a brief description of the field or of its intended contents. The field designation (for example, F52) is entered into the fields in which the user has not entered a header title designation.

Editing the header titles

From Project Manager, select **Project > Header... > Edit header titles**

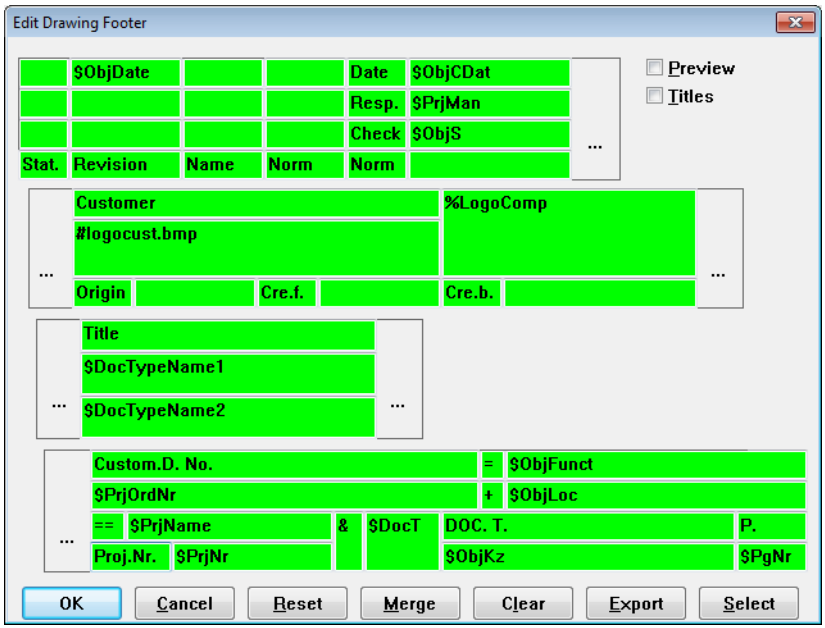


ti012_us.png

Title designations, which can later be superimposed on the header inscription dialog box, can be entered into the Edit drawing header dialog box.

Editing the drawing footer

From Project Manager, select **Project > Header... > Edit drawing footer**



ti013_us.png

Ok Exit footer entries, entries are saved.

Cancel Exit footer entries, entries are rejected.

Reset All entries are accepted by means of the defaults from the file FRAMES.INI.

Merge All variables are accepted with the defaults from the file FRAMES.INI, that is, all fixed texts are preserved.

Clear All field contents are deleted.

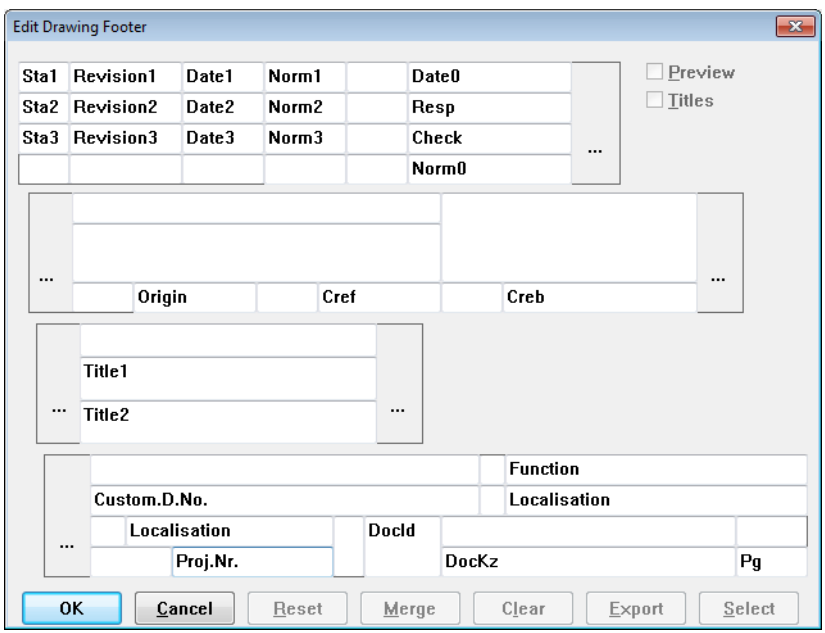
Export The current entries of the footer are written into the file FRAMES.INI and are hence the new defaults for Reset and Merge

By using **Clear** and then **Export** you will lose all entries!

Select	A section from the file FRAMES.INI can be selected from the list field. Its content or pre-settings can then be activated when **Reset** or **Merge** is used.
Preview	A preview showing the contents as they are printed is superimposed. The appearance of inserted bitmaps can be evaluated.
Titles	The titles defined by the user are superimposed, provides a brief description of the field or of its intended contents. The field designation (for example, F24) is entered into the fields in which the user has not entered a footer designation.

Editing the footer titles

From Project Manager, select **Project** > **Header...** > **Edit footer titles**



ti014_us.png

Title designations, which can later be superimposed on the footer inscription dialog box, are entered into the Edit drawing footer dialog box.

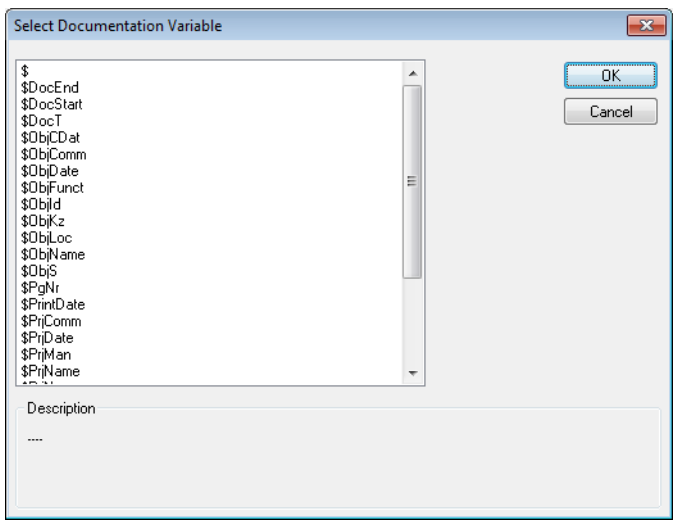
6.3.3 Program-specific drawing header/footer

The project-specific frame inscription is valid initially for all sheets printed out from Freelance Engineering. Each program features a menu item **Program**, **Header**. Both the Drawing header and Drawing footer can be defined under this menu item. However, each modification effected on this level is valid only for this one program (object).

An entry deviating from the default from the file FRAMES.INI is recognized by the fact the yellow field color changes to green.

6.3.4 Automatic allocation of object parameters

To change entries, the cursor must be placed on a field and the required entry made. In the case of the fields for the header and footer inscription, a list of the frame inscription variables available in Freelance Engineering (Documentation variables) can be called additionally with **F2** key. On selecting a variable, a corresponding explanatory text appears in the lower section. For a list of the variables, refer to Variables for drawing footer/header inscriptions on page 315.



ti015_us.png

6.3.5 Bitmaps in the drawing footer

It is possible to implement bitmaps in the drawing footer, header and in the cover page.

The bitmaps are used from the **<Freelance_installation_folder>\bitmaps** directory.

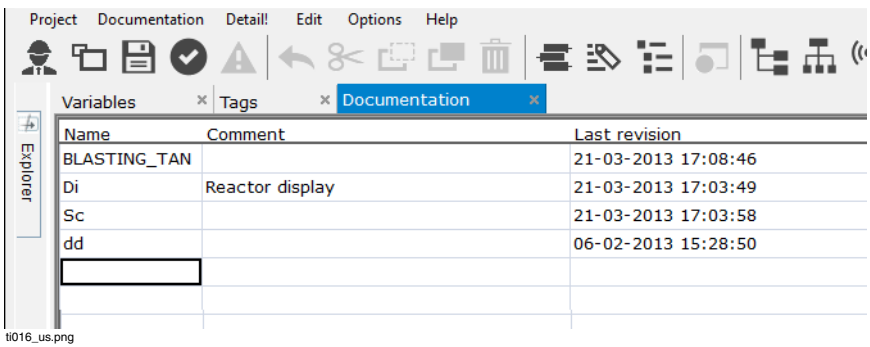
Refer to Fonts, national languages and bitmaps in the drawing footer/header on page 317.

Any other bitmaps can be implemented with #bitmap name, if they are saved in the **<Freelance_installation_folder>\bitmaps** directory.

6.4 Define the documentation job

6.4.1Creating a new print job

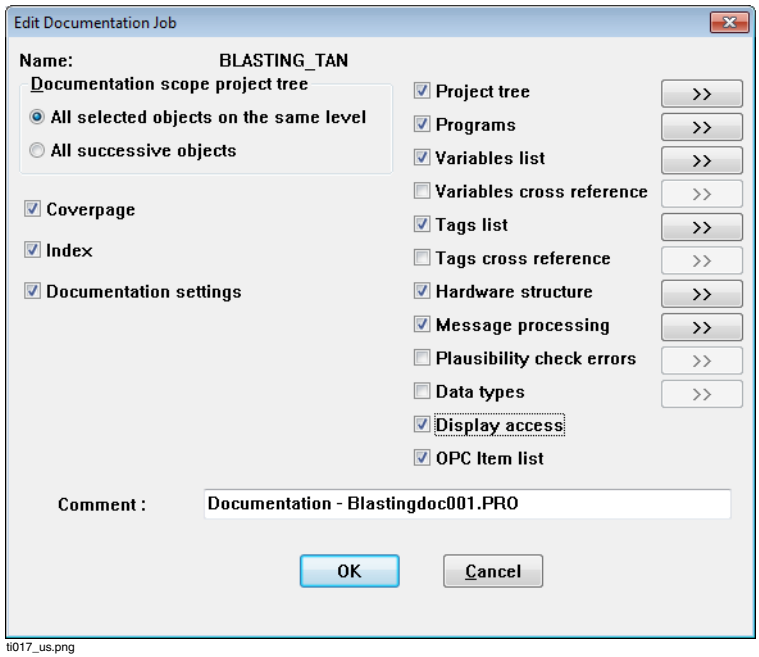
To create a new print job, select the NAME field with a double-click and enter an unequivocal name. Then an explanatory text can be entered into the COMMENT field: In the LAST REVISION field, the date and time are entered automatically by the system. Now the **Detail** menu item is selectable and the documentation scope required under this job is selected.



ti016_us.png

6.4.2 Detail of the print job

> Project > Documentation > Detail!



ti017_us.png

Documentation scope project tree

All selected objects on the same level

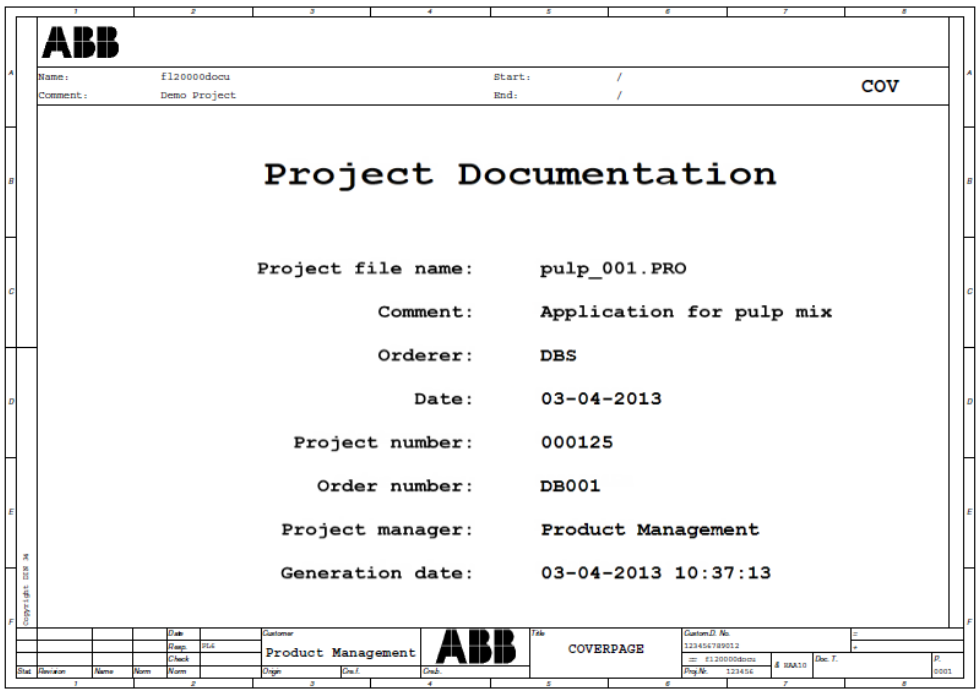
All objects located on the same project tree level as the selected junctions are documented.

All successive objects

All objects underneath the present selection in the project tree are documented. The documentation scope is also determined by the selection in the project tree. By selecting, a summary of the required documentation contents (sheet type) can be indicated for the previously selected job. The **>>** field is enabled if the appropriate sheet type was selected. The significance of the individual sheet types will now be described.

6.4.3 Cover page

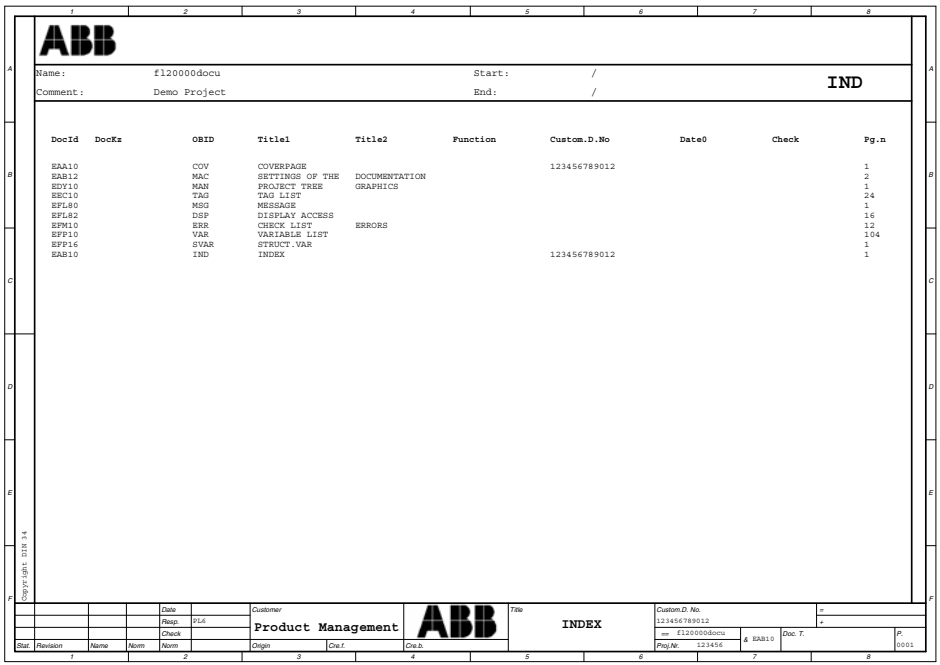
If this field is selected, a cover page will be generated before the documentation job, featuring all data relevant to the project, such as name, comment, date, project number and so on.



ti009_us.png

6.4.4 Index

If this field is selected, a table of contents is generated before the documentation job, featuring a list of all documents printed out, according to the specified order. Refer to Sorting the output on page 297.



Name: f120000docu Start: /
Comment: Demo Project End: /

IND

DocId	DocKs	OBID	Title1	Title2	Function	Custom.D.No	Date0	Check	Pg.n
EAA10		COV	COVERPAGE			123456789012			1
EAB12		MAC	SETTINGS OF THE	DOCUMENTATION					2
EDY10		MAN	PROJECT TREE	GRAPHICS					1
EEC10		TAG	TAG LIST						24
EFL80		MSG	MESSAGE						1
EFL82		DSP	DISPLAY ACCESS						16
EFM10		ERR	CHECK LIST	ERRORS					12
EFP10		VAR	VARIABLE LIST						104
EFP16		SVAR	STRUCT.VAR						1
EAB10		IND	INDEX			123456789012			1

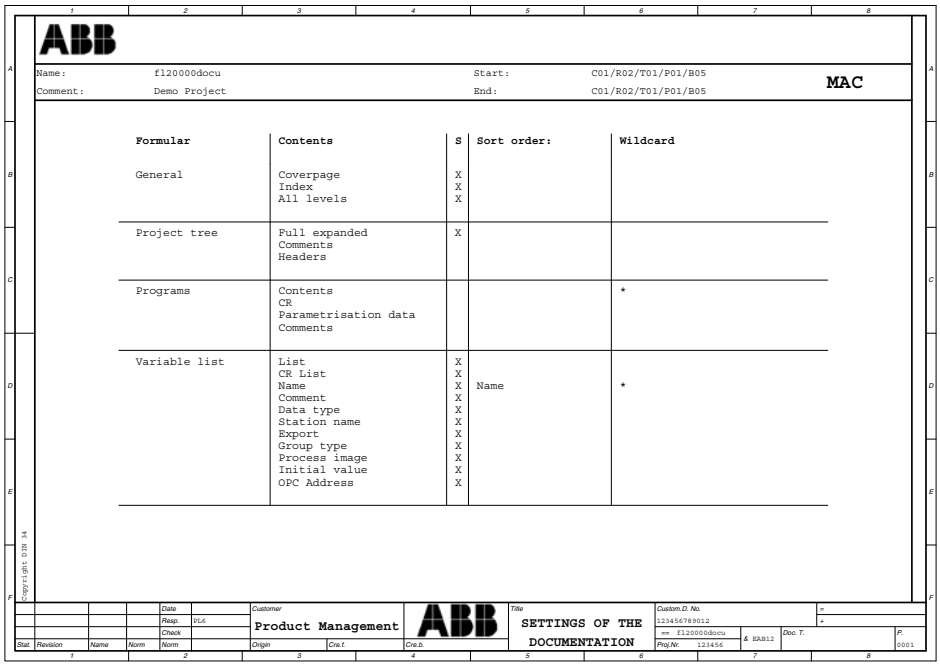
Copyright ABB 94

Customer: Product Management
Title: INDEX
Custom.D. No. 123456789012
Proj.Nr. 123456
Doc. T.
P. 0001

ti018us.eps

6.4.5 Documentation settings

If this field is selected, a sheet is generated featuring the settings selected for the print job.



Name: f120000docu Start: C01/R02/T01/P01/B05
Comment: Demo Project End: C01/R02/T01/P01/B05
MAC

Formular	Contents	S	Sort order:	Wildcard
General	Coverpage	X		
	Index	X		
	All levels	X		
Project tree	Full expanded	X		
	Comments			
	Headers			
Programs	Contents			*
	CR			
	Parametrisation data			
	Comments			
Variable list	List	X		
	CR List	X		
	Name	X	Name	*
	Comment	X		
	Data type	X		
	Station name	X		
	Export	X		
	Group type	X		
	Process image	X		
	Initial value	X		
	OPC Address	X		

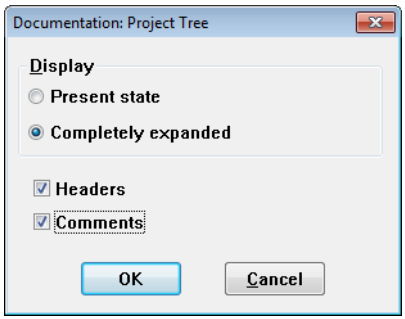
Copyright DIN 34

Customer: Product Management
Title: SETTINGS OF THE DOCUMENTATION
Custom.D. No.: 123456789012
Proj.Nr.: 123456
Doc. T.:
P.: 0001

ti019us.eps

6.4.6 Project tree

> **Project > Documentation > Detail! > Project tree > >>**

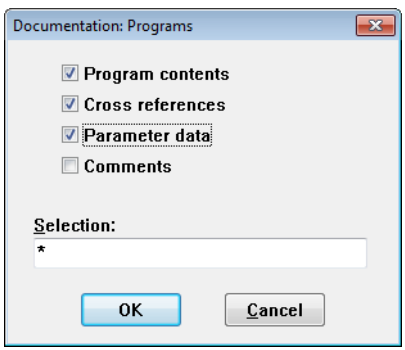


ti020_us.png

Documentation of the project tree can be effected in the present state (for example, parts blanked out) or as a whole. If required, the comments or header information can also be generated for the project tree elements to be documented.

6.4.7 Program Documentation

> **Project > Documentation > Detail! > Programs > >>**



ti021_us.png

The programs selected in the project tree are documented with their contents. A wide variety of sheet types can be selected in a documentation job:

Program contents

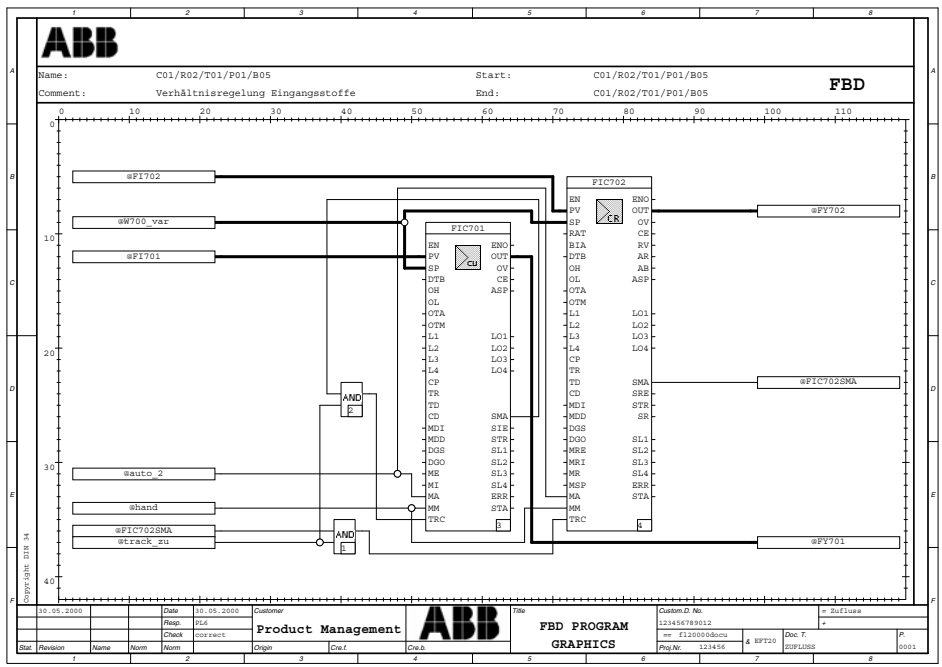
SFC/FBD programs/IL lists/graphic displays. The documentation form depends on the program type. It corresponds to the screen display in the corresponding program.

Cross references

The cross references of the variables of the associated program are documented.

Parameter data Parameter data of function blocks are generated in dialog box form. Since one sheet is generated per block, this should be performed only when required.

Comments Comment sheets are generated for the programs.



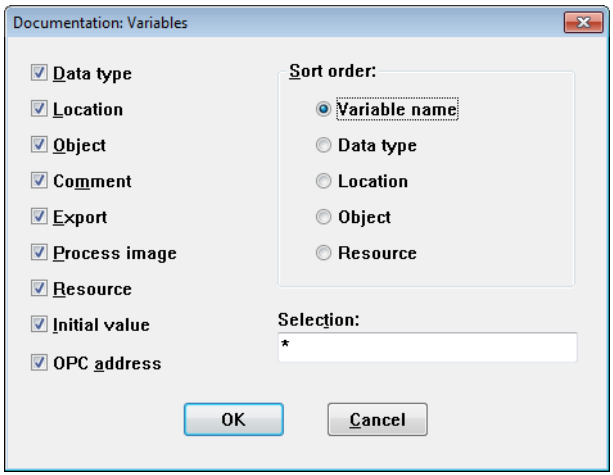
ti022us.eps

Cross references having exactly one source or one target are entered directly within an FBD program. If there are several sources or targets, they cannot be displayed. The cross-reference list is pointed out. Always this list also should be printed. If references are made to inputs and outputs, their slot and channel are specified.

For more information, refer to ***Engineering Manual IEC 61131-3 Programming, Function Block Diagram (FBD)***.

6.4.8 Variables list

> Project > Documentation > Detail! > Variables list > >>



ti023_us.png

Specify which parts of the variables lists are to be printed.

Select the required fields:

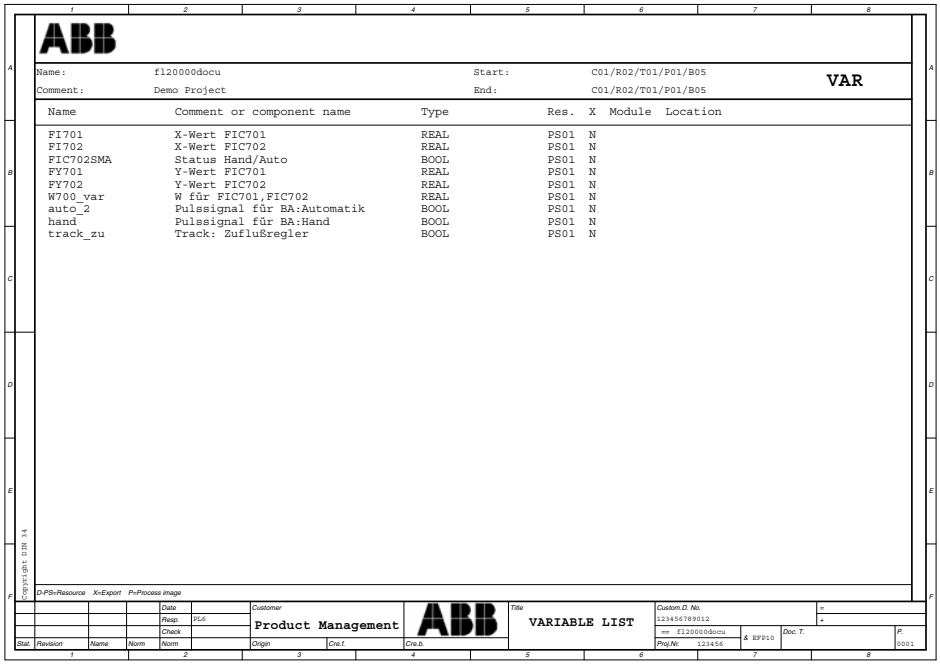
All the selected variables are printed as a list, featuring supplementary information such as comment, type, resource and so on.

It is also possible to define how the list is **Sorted**.

It can be sorted according to **Variable name**, **Data type**, **Location**, **Object**, or **Resource**.

The list size can be limited by a selection with wildcards.

Those variables are printed which have a cross reference into the selected part of the project tree.



Name: f120000docu
Comment: Demo Project
Start: C01/R02/T01/P01/B05
End: C01/R02/T01/P01/B05
VAR

Name	Comment or component name	Type	Res.	X	Module	Location
FI701	X-Wert FIC701	REAL	PS01	N		
FI702	X-Wert FIC702	REAL	PS01	N		
FIC702SMA	Status Hand/Auto	BOOL	PS01	N		
FY701	Y-Wert FIC701	REAL	PS01	N		
FY702	Y-Wert FIC702	REAL	PS01	N		
W700_var	W für FIC701,FIC702	REAL	PS01	N		
auto_2	Pulssignal für BA:Automatik	BOOL	PS01	N		
hand	Pulssignal für BA:Hand	BOOL	PS01	N		
track_zu	Track: Zuflußregler	BOOL	PS01	N		

D=PS=Resource X=Export P=Process image

Copyright ABB 34

Customer: Product Management
Title: VARIABLE LIST
Custom.D. No.: 123456789012
Proj.Nr.: 123456
Doc. T.
P.: 0001

ti024us.eps

Name Variable name

Comment or component name
Comment of the variable or component name of the structured variable

Type Data type as REAL, BOOL, UINT, UDINT, TIME, STRING, or DT

Res. Allocation of resource

X Enable variable for reading other resources

Object Object types (for example, DD01)

Location Designations of the object (for example, PS_1_0_2),

PS_1 Resource name

0 Unit (rack)

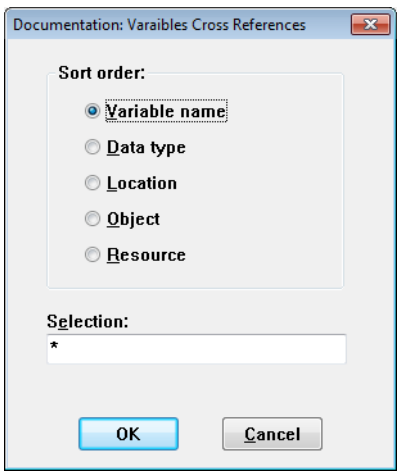
2 Location

P	The process variable is processed through the process image or directly
Initial value	The default value set at cold start of the process station

For more information, refer to ***Engineering Manual IEC 61131-3 Programming, Variables***.

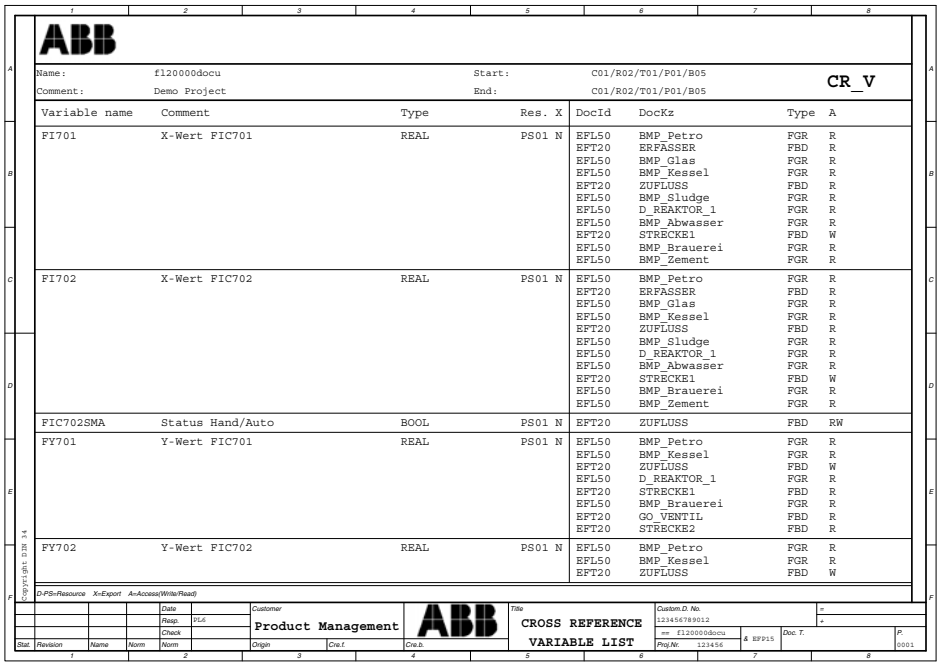
6.4.9 Variables cross reference list

> Project > Documentation > Detail! > Variables cross reference > >>



ti025_us.png

A cross reference list of the variables used in programs and displays can be generated. The Documentation: Variables cross reference dialog box appears for the variables list. For more information on selecting the sort order of variables cross references, refer to Variables list on page 279.



Name: fl20000docu Start: C01/R02/T01/P01/B05
Comment: Demo Project End: C01/R02/T01/P01/B05

CR_V

Variable name	Comment	Type	Res.	X	DocId	DocKz	Type	A
FI701	X-Wert FIC701	REAL	PS01	N	EFL50	BMP_Petro	FGR	R
					EFT20	ERFASSER	FBD	R
					EFL50	BMP_Glas	FGR	R
					EFL50	BMP_Kessel	FGR	R
					EFT20	ZUFLUSS	FBD	R
					EFL50	BMP_Sludge	FGR	R
					EFL50	D_REAKTOR_1	FGR	R
					EFL50	BMP_Abwasser	FGR	R
					EFT20	STRECKE1	FBD	W
					EFL50	BMP_Brauerei	FGR	R
					EFL50	BMP_Zement	FGR	R
FI702	X-Wert FIC702	REAL	PS01	N	EFL50	BMP_Petro	FGR	R
					EFT20	ERFASSER	FBD	R
					EFL50	BMP_Glas	FGR	R
					EFL50	BMP_Kessel	FGR	R
					EFT20	ZUFLUSS	FBD	R
					EFL50	BMP_Sludge	FGR	R
					EFL50	D_REAKTOR_1	FGR	R
					EFL50	BMP_Abwasser	FGR	R
					EFT20	STRECKE1	FBD	W
					EFL50	BMP_Brauerei	FGR	R
					EFL50	BMP_Zement	FGR	R
FIC702SMA	Status Hand/Auto	BOOL	PS01	N	EFT20	ZUFLUSS	FBD	RW
FY701	Y-Wert FIC701	REAL	PS01	N	EFL50	BMP_Petro	FGR	R
					EFL50	BMP_Kessel	FGR	R
					EFT20	ZUFLUSS	FBD	W
					EFL50	D_REAKTOR_1	FGR	R
					EFT20	STRECKE1	FBD	R
					EFL50	BMP_Brauerei	FGR	R
					EFT20	GO_VENTIL	FBD	R
					EFT20	STRECKE2	FBD	R
FY702	Y-Wert FIC702	REAL	PS01	N	EFL50	BMP_Petro	FGR	R
					EFL50	BMP_Kessel	FGR	R
					EFT20	ZUFLUSS	FBD	W

D=PS=Resource X=Export A=Access(Write/Read)

Copyright ABB 34

Customer: Product Management
Title: CROSS REFERENCE VARIABLE LIST
Custom.D. No.: 123456789012
Proj.Nr.: 123456
Doc. T.:
P.: 0001

ti026us.eps

Variable name Maximum 16 characters

Comment Maximum 34 characters

Type Data type as REAL, BOOL, UINT, UDINT, TIME, STRING or DT

Res. Allocation of resource

X Enable variable for reading other resources

DocId Document type

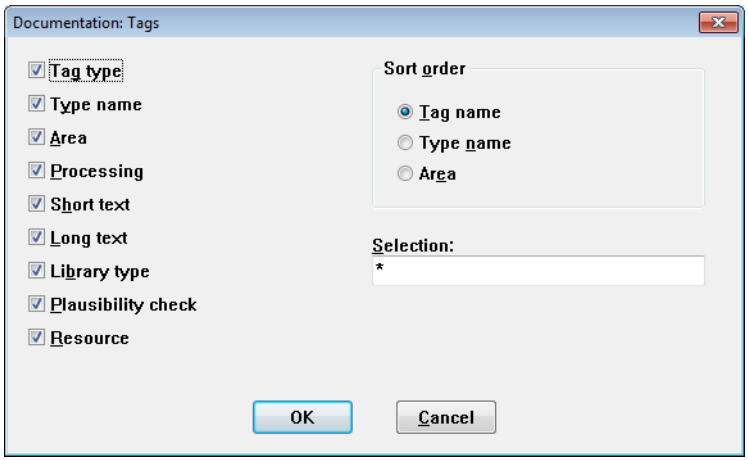
DocKz Document identification

Type Program or display type (IL, FBD, FGR)

A Source or sink of variables, R = Read, W = Write. For more information, refer to ***Engineering Manual IEC 61131-3 Programming, Variables***.

6.4.10 Tags list

> **Project > Documentation > Detail! > Tags list > >>**



ti027_us.png

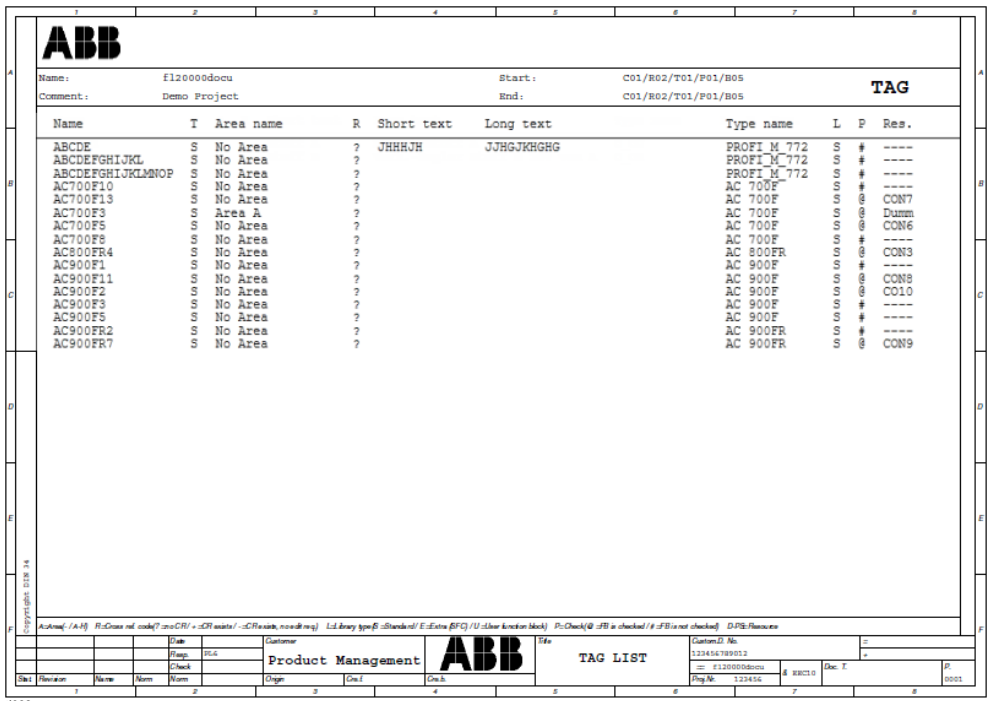
Specify which parts of the Tag list are to be printed.

Select the required fields.

All the selected modules are printed as a list, featuring supplementary information such as comment, type name, resource and so on.

It is also possible to define how the list is **sorted**. It can be sorted according to **Tag name**, **Type name** and **Area**.

The list size can be limited by a selection with wildcards.



Name: fl20000docu
Comment: Demo Project
Start: C01/R02/T01/P01/B05
End: C01/R02/T01/P01/B05

TAG

Name	T	Area name	R	Short text	Long text	Type name	L	P	Res.
ABCDE	S	No Area	?	JHHHJH	JJHGJKHGHG	PROFI_M_772	S	#	----
ABCDEFGHIJKL	S	No Area	?			PROFI_M_772	S	#	----
ABCDEFGHIJKLMNOP	S	No Area	?			PROFI_M_772	S	#	----
AC700F10	S	No Area	?			AC 700F	S	#	----
AC700F13	S	No Area	?			AC 700F	S	@	CON7
AC700F3	S	Area A	?			AC 700F	S	@	Dumm
AC700F5	S	No Area	?			AC 700F	S	@	CON6
AC700F8	S	No Area	?			AC 700F	S	#	----
AC800FR4	S	No Area	?			AC 800FR	S	@	CON3
AC900F1	S	No Area	?			AC 900F	S	#	----
AC900F11	S	No Area	?			AC 900F	S	@	CON8
AC900F2	S	No Area	?			AC 900F	S	@	CO10
AC900F3	S	No Area	?			AC 900F	S	#	----
AC900F5	S	No Area	?			AC 900F	S	#	----
AC900FR2	S	No Area	?			AC 900FR	S	#	----
AC900FR7	S	No Area	?			AC 900FR	S	@	CON9

A:Area(- / A-H) R:Cross ref. code(? :no CR/ + :CR exists / - :CR exists, no edit req) L:Library type(S :Standard/ E:Extra (SFC) / U :User function block) P:Check(@ :FB is checked / # :FB is not checked) D:PB:Resource

Copyright© ABB AG

Customer: Product Management
Title: TAG LIST
Custom.D. No.: 1234567899012
Proj.Nr.: 123456
Doc. T.:
P.: 0001

ti028_us.png

Name Tag name, maximum 12/16 characters

T Tag type

S = Standard name

F = Formal name

T = Template name

Area name Plant area of tag (A to O)

R Processing state, only information

\+ Function block is in processing (Processing ☑)

\- Function block is not in processing (Processing ☐)

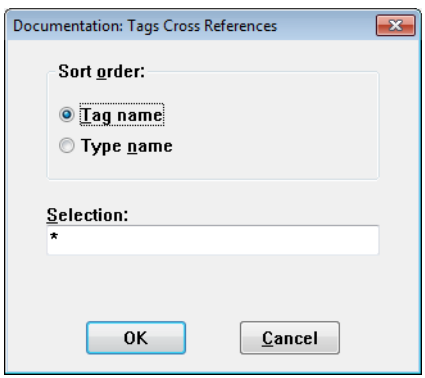
? Processing of function block not defined (Processing ▩)

Sequence chains and I/O blocks are always shown with “?”

Short/long text	
	Maximum 12/30 characters
Type name	Brief designations of the function block type
L	Library type
	S = Standard library type
	U = User defined function block library
	E = Extra Library type (SFC)
P	# Block not checked for plausibility
	@ Block checked for plausibility
Res.	Allocated resource

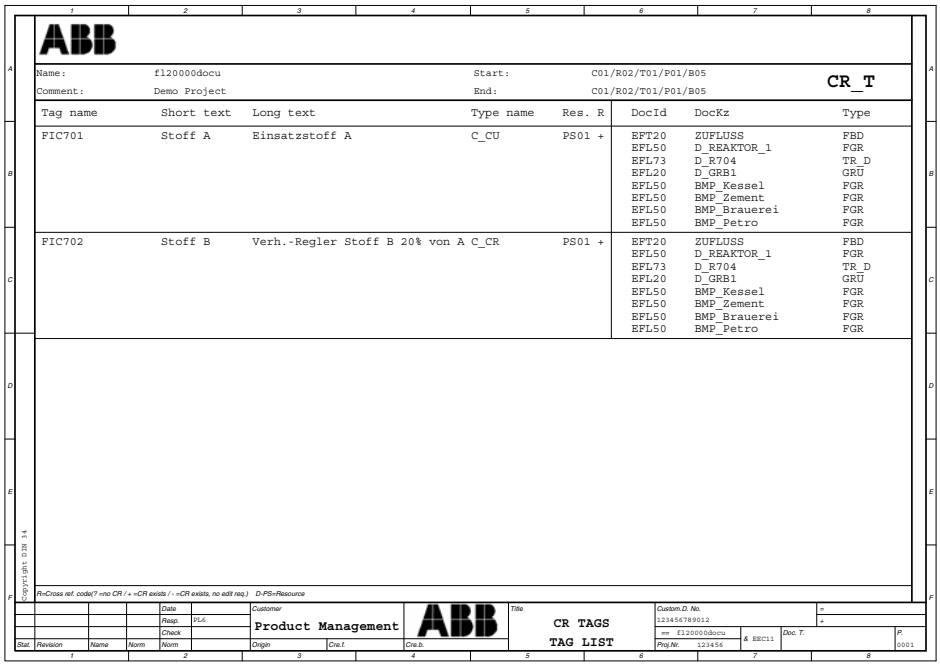
6.4.11 Tags cross references

> Project > Documentation > Detail! > Tags cross reference > >>



ti029_us.png

The dialog box is the same as for the tags list. For more information, refer to Tags list on page 283.



Name: f120000docu Start: C01/R02/T01/P01/B05 **CR_T**
Comment: Demo Project End: C01/R02/T01/P01/B05

Tag name	Short text	Long text	Type name	Res.	R	DocId	DocKz	Type
FIC701	Stoff A	Einsatzstoff A	C_CU	PS01	+	EFT20	ZUFLUSS	FBD
						EFL50	D_REAKTOR_1	FGR
						EFL73	D_R704	TR_D
						EFL20	D_GRB1	GRU
						EFL50	BMP_Kessel	FGR
						EFL50	BMP_Zement	FGR
						EFL50	BMP_Brauerei	FGR
						EFL50	BMP_Petro	FGR
FIC702	Stoff B	Verh.-Regler Stoff B 20% von A	C_CR	PS01	+	EFT20	ZUFLUSS	FBD
						EFL50	D_REAKTOR_1	FGR
						EFL73	D_R704	TR_D
						EFL20	D_GRB1	GRU
						EFL50	BMP_Kessel	FGR
						EFL50	BMP_Zement	FGR
						EFL50	BMP_Brauerei	FGR
						EFL50	BMP_Petro	FGR

R=Cross ref. code(? =no CR / + =CR exists / - =CR exists, no edit req.) D-PS=Resource

Copyright ABB 34

Customer: Product Management
Title: CR TAGS TAG LIST
Custom.D. No.: 123456789012
Proj.Nr.: 123456
Doc. T.:
P.: 0001

ti030us.eps

Tag name Name of the function block, maximum 12/16 characters

Short/long text

Maximum 12/30 characters

Type name Brief designations of the function block type

Res. Allocated resource

R Cross reference code

? = no cross reference

+ = cross reference exists

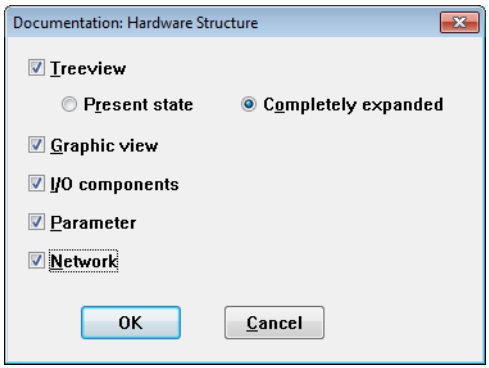
DocId Document type

DocKz Document identification

Type Program or display type (IL, LD, FBD, FGR)

6.4.12 Hardware structure

> **Project > Documentation > Detail! > Hardware structure > >>**



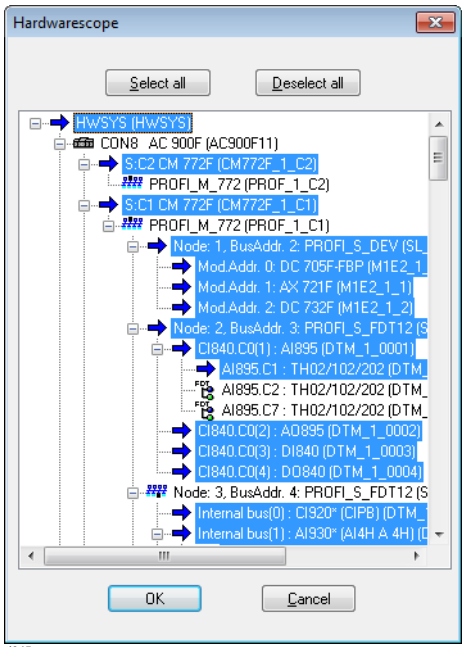
ti031_us.png

Specify which parts of the hardware structure are to be printed. Select the required fields.

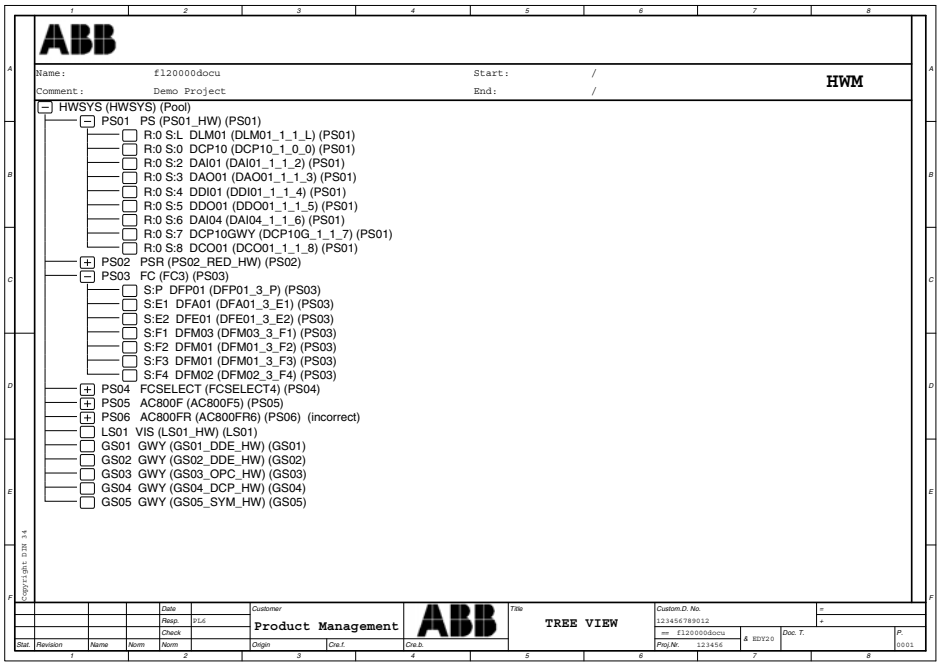
For selective documentation of the menu item **I/O components and parameter** a selection dialog box is available under **> Project > Documentation > Options > Hardware scope...**. For more information, refer to Selecting the hardware scope on page 299.

The menu items, **I/O components and Parameters**, should only be selected if an I/O component selection has first been made, which is useful for the project as described above. Otherwise, an expression for the I/O components and the parameters is created for each hardware component entered in the project, including all configured Profibus slaves and their modules.

If **modular Profibus slaves** are configured in the project, it is useful for a clearly viewable and compact display of the I/O components and parameters (user and DPV1 parameters), to mark only the modular Profibus slave without the associated modules, when setting up a print job for the hardware structure in the selection window under **Project > Documentation > Options > Hardware scope...**. The hardware tree must be opened under the slave which is to be documented. Otherwise, all the modules beneath it are also marked. Single modules for the hardware documentation can also be selected.

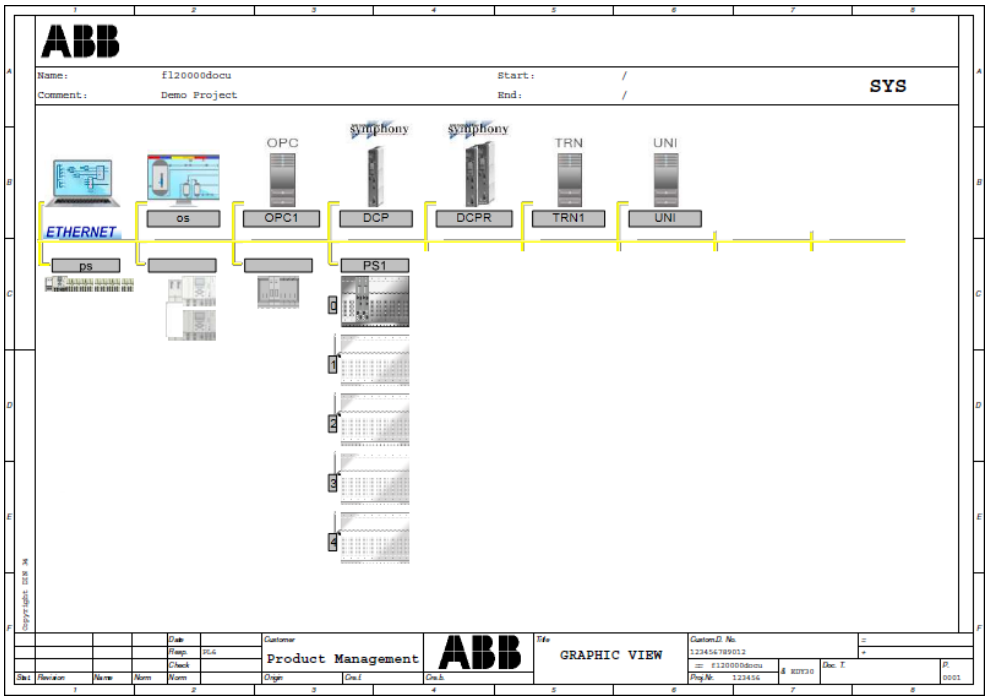

Hardwarescope
Select all
Deselect all
HWSYS [HWSYS]
CON8 AC 900F (AC900F11)
S:C2 CM 772F (CM772F_1_C2)
PROFI_M_772 (PROF_1_C2)
S:C1 CM 772F (CM772F_1_C1)
PROFI_M_772 (PROF_1_C1)
Node: 1, BusAddr. 2: PROFI_S_DEV (SL
Mod.Addr. 0: DC 705F-FBP (M1E2_1_
Mod.Addr. 1: AX 721F (M1E2_1_1)
Mod.Addr. 2: DC 732F (M1E2_1_2)
Node: 2, BusAddr. 3: PROFI_S_FDT12 (S
CI840.C0(1) : AI895 (DTM_1_0001)
AI895.C1 : TH02/102/202 (DTM_
AI895.C2 : TH02/102/202 (DTM_
AI895.C7 : TH02/102/202 (DTM_
CI840.C0(2) : AO895 (DTM_1_0002)
CI840.C0(3) : DI840 (DTM_1_0003)
CI840.C0(4) : DO840 (DTM_1_0004)
Node: 3, BusAddr. 4: PROFI_S_FDT12 (S
Internal bus(0) : CI920* (CIPB) (DTM_
Internal bus(1) : AI930* (AI4H A 4H) (D
OK
Cancel
ti045_us.png

6.4.13 Tree view

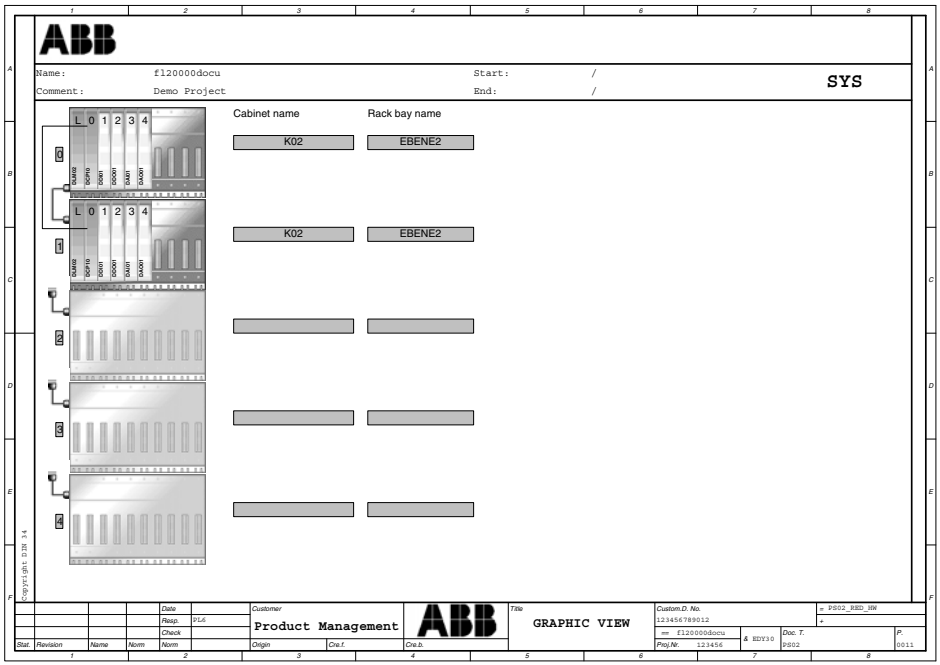


ti032us.eps

6.4.14 Graphic view

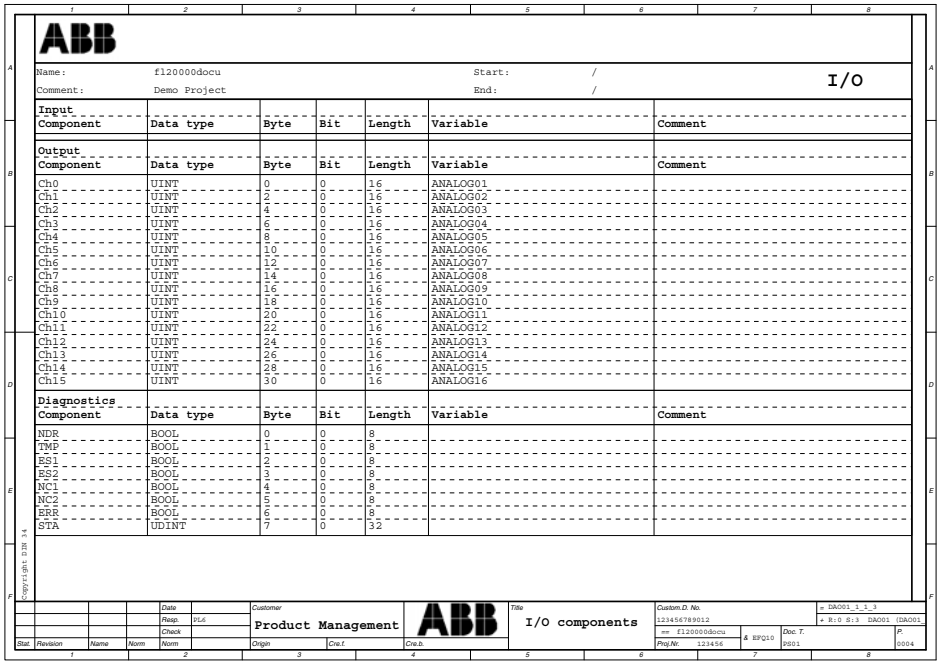


ti033_us.png


Name: fl20000docu
Start: /
Comment: Demo Project
End: /
SYS
Cabinet name
Rack bay name
K02
EBENE2
K02
EBENE2
GRAPHIC VIEW
Product Management

ti034us.eps

6.4.15 I/O components



Name: fl20000docu Start: /

Comment: Demo Project End: /

I/O

Input

Component	Data type	Byte	Bit	Length	Variable	Comment

Output

Component	Data type	Byte	Bit	Length	Variable	Comment
Ch0	UINT	0	0	16	ANALOG01	
Ch1	UINT	2	0	16	ANALOG02	
Ch2	UINT	4	0	16	ANALOG03	
Ch3	UINT	6	0	16	ANALOG04	
Ch4	UINT	8	0	16	ANALOG05	
Ch5	UINT	10	0	16	ANALOG06	
Ch6	UINT	12	0	16	ANALOG07	
Ch7	UINT	14	0	16	ANALOG08	
Ch8	UINT	16	0	16	ANALOG09	
Ch9	UINT	18	0	16	ANALOG10	
Ch10	UINT	20	0	16	ANALOG11	
Ch11	UINT	22	0	16	ANALOG12	
Ch12	UINT	24	0	16	ANALOG13	
Ch13	UINT	26	0	16	ANALOG14	
Ch14	UINT	28	0	16	ANALOG15	
Ch15	UINT	30	0	16	ANALOG16	

Diagnostics

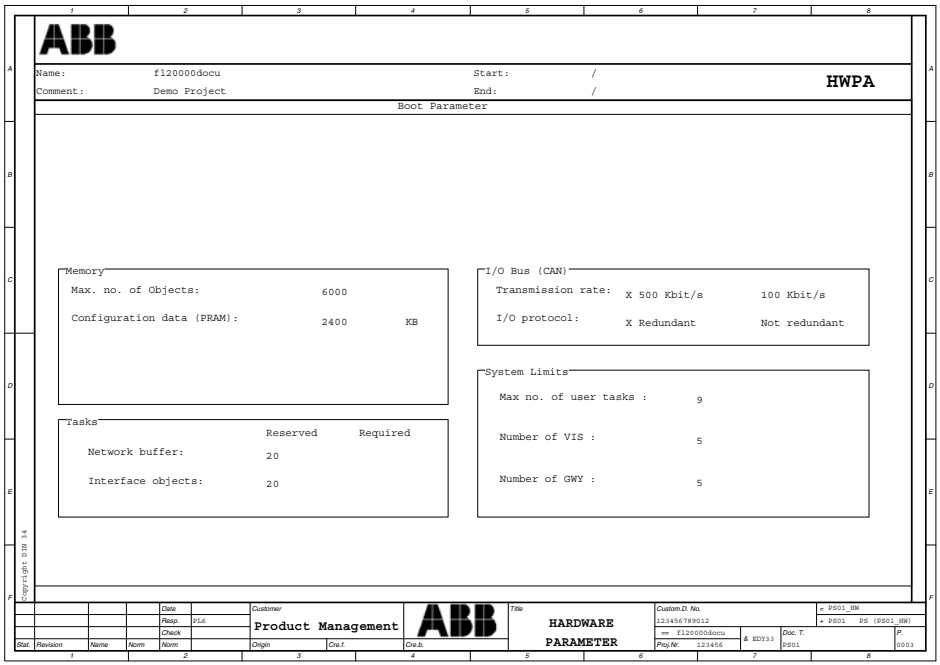
Component	Data type	Byte	Bit	Length	Variable	Comment
NDR	BOOL	0	0	8		
TMP	BOOL	1	0	8		
ES1	BOOL	2	0	8		
ES2	BOOL	3	0	8		
NC1	BOOL	4	0	8		
NC2	BOOL	5	0	8		
ERR	BOOL	6	0	8		
STA	UDINT	7	0	32		

Product Management

I/O components

ti035us.eps

6.4.16 Parameter



ti036us.eps

6.4.17 Network

ABB

Name: fl20000docu Start: /
Comment: Demo Project End: /

NET

Res. type	Res. name	Res. ID	IP address 1	IP address 2
D-ES		10	172.16.1.10	
D-GS/RED	GS05	70	172.16.1.70	172.16.1.71
D-GS	GS04	60	172.16.1.60	
D-GS	GS03	50	172.16.1.50	
D-GS	GS02	40	172.16.1.40	
D-GS	GS01	30	172.16.1.30	
D-OS	LS01	11	172.16.1.10	
D-PS/RED	PS06	6	172.16.1.6	
D-PS	PS05	5	172.16.1.5	
D-PS	PS04	4	172.16.1.4	
D-PS	PS03	3	172.16.1.3	
D-PS/RED	PS02	2	172.16.1.2	172.16.1.102
D-PS	PS01	1	172.16.1.1	

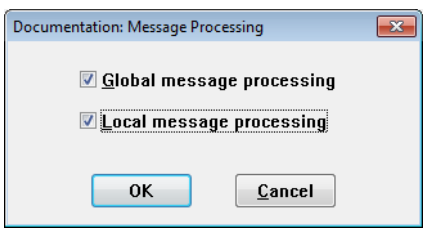
Copyright DIN 34

Customer: Product Management
Title: NETWORK
Custom.D. No.: 123456789012
Proj.Nr.: 123456
Doc. T.:

ti037us.eps

6.4.18 Message processing

> Project > Documentation > Detail! > Message processing > >>



ti038_us.png

Global message processing

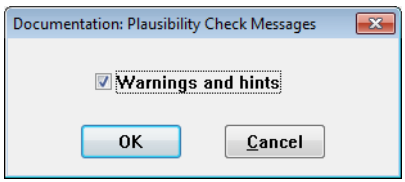
The output can be higher-level message processing.

Local message processing

The output can be message processing of the selected resource. For more information, refer to ***Engineering Manual Operator Station, Messages and Hints***.

6.4.19 Plausibility check errors

> **Project** > **Documentation** > **Detail!** > **Plausibility check errors** > **>>**

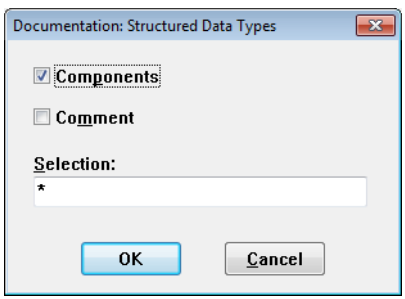


ti039_us.png

The output can be plausibility check errors. If the field *Warnings and hints* is selected, additionally messages are generated.

6.4.20 Structured data types

> **Project** > **Documentation** > **Detail!** > select **Data types** > **>>**



ti040_us.png

Select the components and comments of the user specific structured variables for the output.

6.4.21 Display access

> Project > Documentation > Detail! > Display access

The display allocations are documented as tables.

Name: f120000docu Start: /
Comment: Demo Project End: /

DSP

Tag name	Display name	Default
	Trend_10	N
L1120_G1	AK11	N
	Trend_11	N
L3020_G1	R30	N
	Trend_30	N
L3120_G1	R31	N
	Trend_31	N
L6001_G1	T60	N
L7001_G1	T70	N
L8001_G1	T80	N
L9001_G1	T90	N
LI3005	REAKTOR_3_G1	N
	REAKTOR_3	N
	REAKTOR_3_KB	N
	REAKTOR_3_AS	N
LI4005	REACTOR_4_G1	N
	REACTOR_4	N
	REACTOR_4_TR	N
	REACTOR_4_SF	N
LI700	D_GRB3	N
	D_REA_TANK	N
LI7005	REAKTOR_7_G1	N
	REAKTOR_7	N
	REAKTOR_7_TR	N
	REAKTOR_7_AS	N
LI720	D_R704	N
LI720_1	D_R704	N
LI750	D_FUELLEN_R1	N
	D_Rez_Reakt	N
LI752	D_Rez_Reakt	N
LI753	D_Rez_Reakt	N
LIC704	D_GRB1	N
	BMP_Abwasser	N
	D_R704	N
NI704	D_GRB1	N
	D_REAKTOR_1	N
NP2000A_M1	B20	N

Customer: Product Management
Title: DISPLAY ACCESS
Custom.D. No.: 123456789012
Proj.Nr.: 123456
Doc. T.:

ti041us.eps

Tag name Tag name of the allocated tag

Display name Display names of the displays which are allocated to the tag

Default **Y** if the display is defined as the default display

N if the display is not defined as the default display.

6.5 Documentation scope

6.5.1 Hard disk requirement

If output is in several sheets (for example, an entire project), ensure that there is enough space available on the temporary disk. Three sheets are always combined for one print job for the Windows print manager. This ensures that already at the time of creating the print job for Freelance Engineering, the first data can be transferred to the printer. Generation of the print data and output to the printer are effected with a slight delay and the first temporary data can be deleted again.

6.5.2 Viewing selected documentation scope

> Project > Documentation > Documentation > Preview...

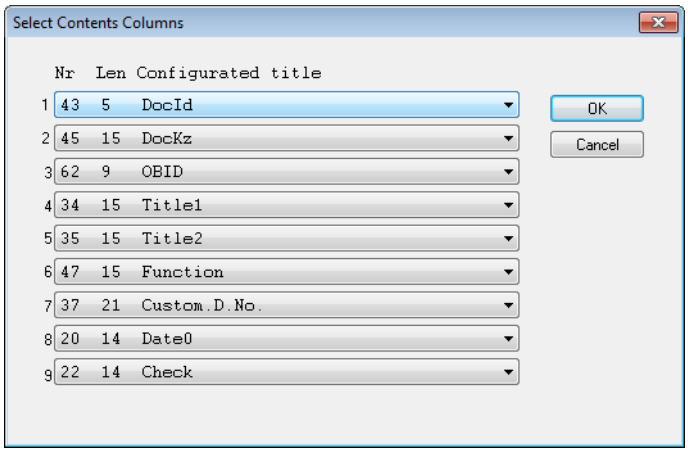
A table of contents of the sheets that have been selected in the previously selected documentation job is displayed. For example,

DocID	DocKz	OBjID	Title1	Title2	Function	Custom.D.No.	Date0	Check
EAA10		COV	COVERPAGE			123456789012	30.05.00	
EAB12		MAC	SETTINGS	DOCUMENTATION		123456789012	30.05.00	
EDY20		HWM	TREE VIEW	GRAPHICS		123456789012	30.05.00	
EEC10		TAG	TAG LIST			123456789012	30.05.00	
EFP15		CR_V	CROSS-REFERENCE	VARIABLE LIST		123456789012	30.05.00	
EFT20	INFLUX	FBD	FBD PROGRAM	GRAPHICS	INFLUX	123456789012	30.05.00	correct

6.5.3 Sorting the output

Start in Window “Configuration: Documentation”

> Project > Documentation > Options > Sort fields...



ti043_us.png

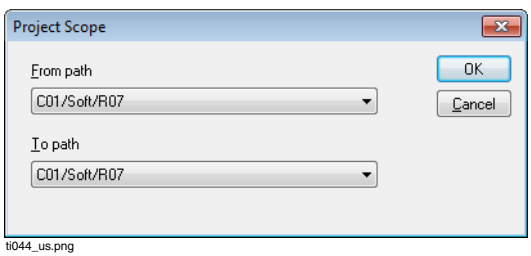
Sorting of the output can be defined. Documentation output can be sorted according to a maximum of five criteria (64, 20, 20, 12, 3 characters). The field contents of the drawing footer can be selected as sorting criteria. Refer to Field names in the drawing footer/header on page 312. The standard setting of Freelance Engineering is shown in the example.

Each sorting criterion set here automatically becomes a column in the table of contents when printing. Sorting to date is not possible.

6.5.4 Selecting the project tree objects

Start in Window “Configuration: Documentation”

> Project > Documentation > Options > Project scope...

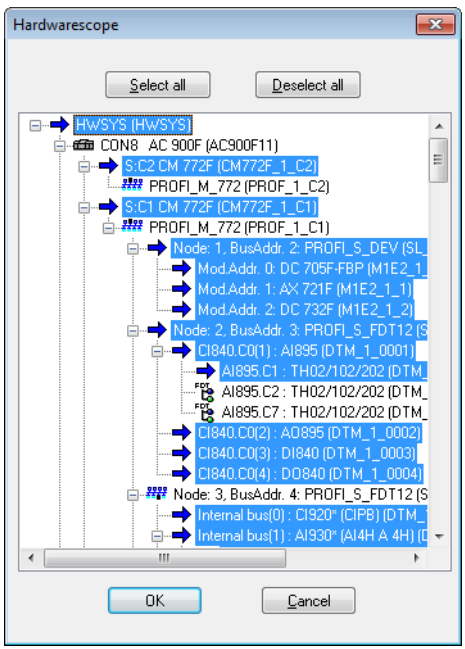


ti044_us.png

In the project scope dialog box a selection of the project tree objects **From path** and **To path** (only in the order of the project tree objects) can be made for the print, if this has not already been selected itself for print in the project tree.

6.5.5 Selecting the hardware scope

> Project > Documentation > Options > Hardware scope...



ti045_us.png

In the Hardwarescope dialog box, a selection can be made for the documentation of required I/O components.

This selection is only active as long as you are in the “Documentation” tab. After leaving this tab, the completely configured hardware is selected by default! This point is only important in connection with the selection under:

> Project > Documentation > Detail! > Hardware structure

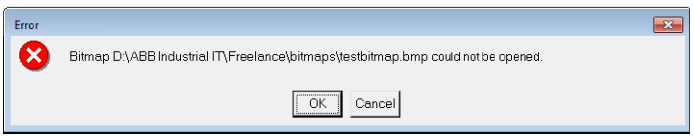
> I/O components > Parameter

6.5.6 Selecting notify print errors

> Project > Documentation > Options > Notify print errors

At this point, you determine whether errors appearing in connection with the printing of the project documentation, such as printing errors, missing objects (for example, bitmaps), and invalid or undefined field functions in drawing header/footer, are shown on the screen.

Example of a possible error message:



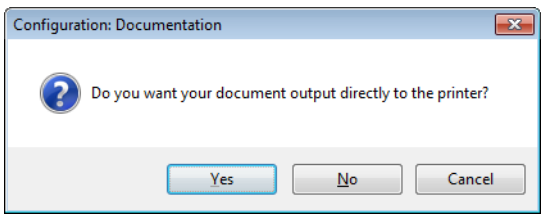
Printerror_us.png

6.6 Output documentation

6.6.1Print

Start in Window “Configuration: Documentation”

> Project > Documentation > Documentation > Print



ti047_us.png

Start printing to a file or directly to the printer.

Yes The output is effected to the standard printer entered in Windows. Depending on the printer option, the print quality can still be adapted and the number of required copies entered. Printing can be aborted with **Cancel**. Using the Setup button, another printer type other than the standard printer can be selected if necessary. For more information on setting the printer, refer to Printer setup on page 303. A message box appears as an indication that the printer is set to DIN A4 landscape format.

No It is possible to print to file. If this option is selected, a window opens and the path, file name of the target file can be edited here. Default setting for the file name: name of the print job with the extension “.DPD”; Default setting for the path is **<Freelance_installation_folder>\proj**. For more information on setting print file, refer to Print file on page 302.

Cancel Cancel printing

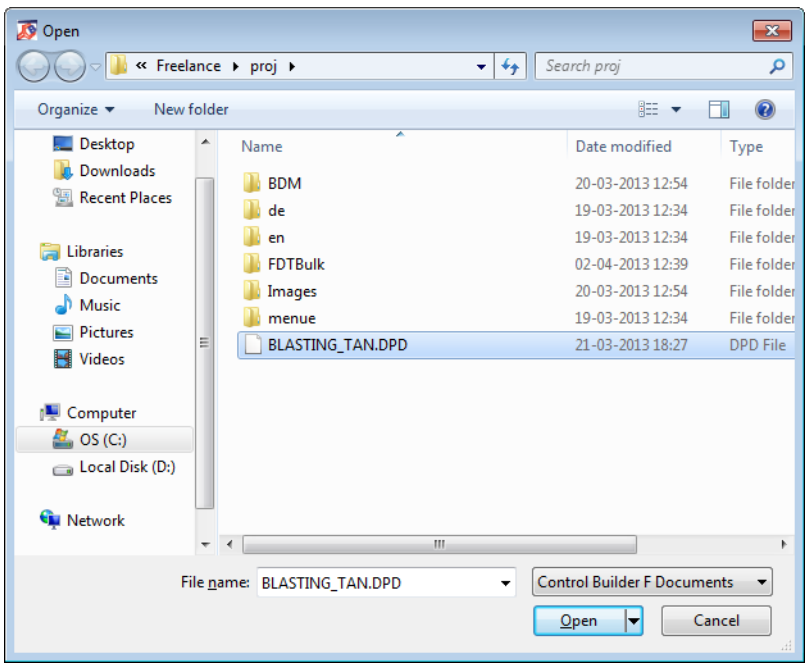
The maximum amount of hard disk space must be available.

Refer to Hard disk requirement on page 297. The preview function should be used to get a survey of the print scope. Refer to Viewing selected documentation scope on page 297.

6.6.2 Print file

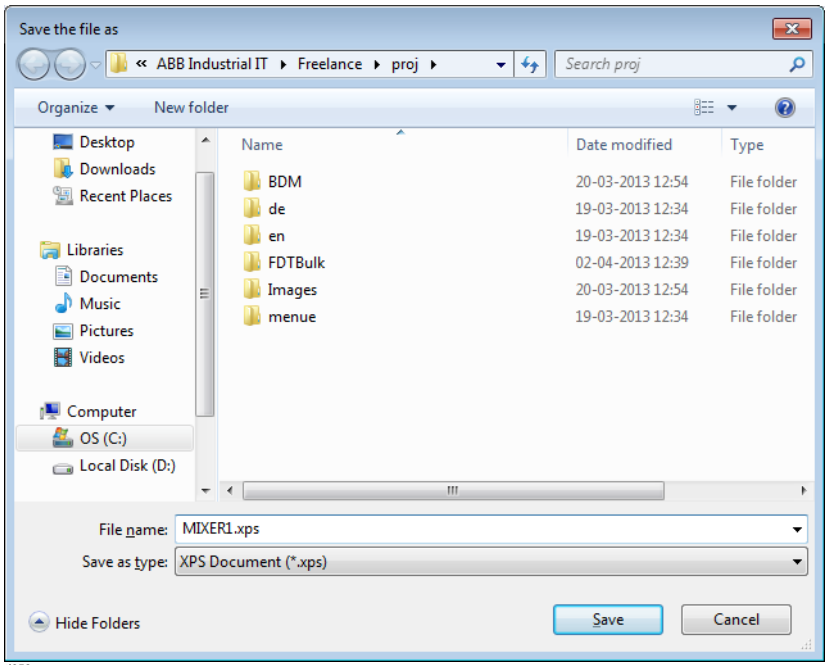
Start in Window “Configuration: Documentation”

> Project > Documentation > Documentation > Print file



ti048_us.png

Print a previously generated print file: The file to be printed is selected in a File name list box. The files will be saved with the extension .DPD.

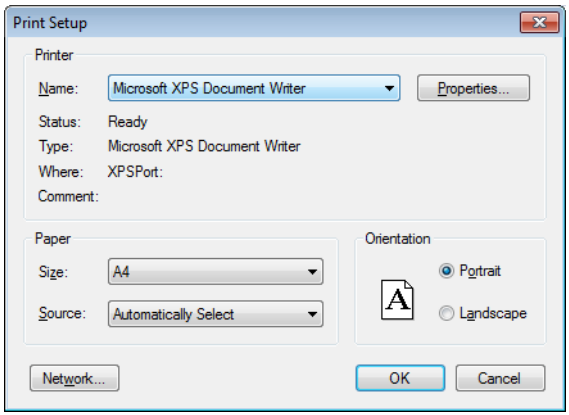


ti059_us.png

6.6.3 Printer setup

Start in Window “Configuration: Documentation”

> **Project > Documentation > Documentation > Printer setup**



ti049_us.png

Select the printer. If a printer other than the standard printer is to be used, **Special printer** is selected and the associated button is clicked. Now the required printer can be selected. Select **Properties** to adjust the selected printer, For more information on setting properties for a selected printer, refer to Windows documentation.

The selected paper format (DINA4, letter) is supported. Default: landscape format.

Adjustments for the page layout can be changed under **ABB > Freelance <version> > Settings**, page layout.

6.7 Comment field editing

6.7.1Edit comment field

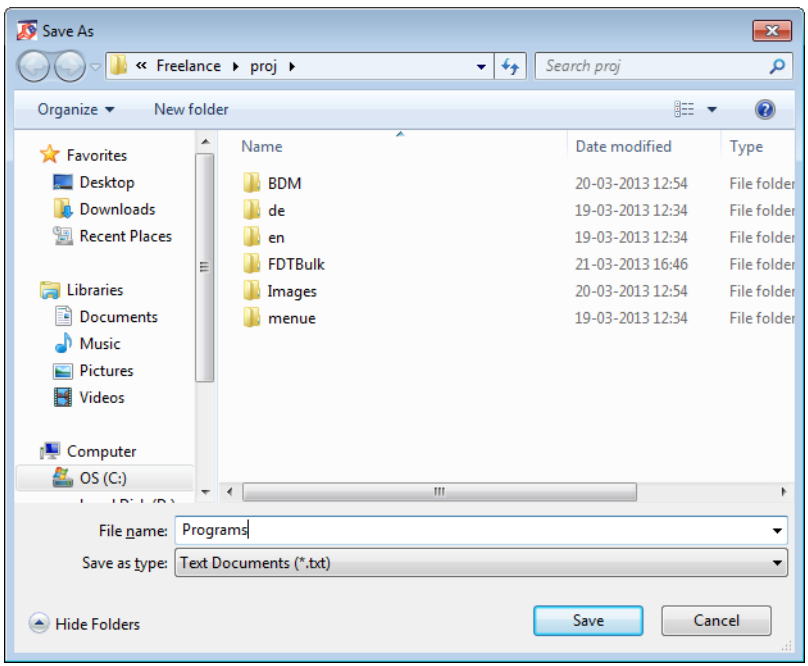
> Project > Documentation > Comment > Edit > Text input

The text contents of the comment field can be changed.

6.7.2 Export comment field

> Project > Documentation > Comment > Edit > Export

The text content of the comment field is saved as a text file *.txt, and serves as a basis for new print job comments. This text file is stored in Unicode format.

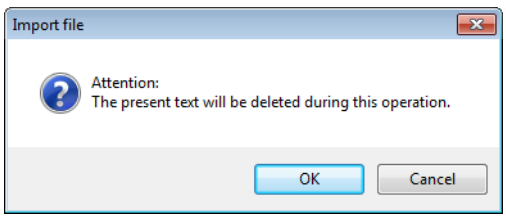


ti060_us.png

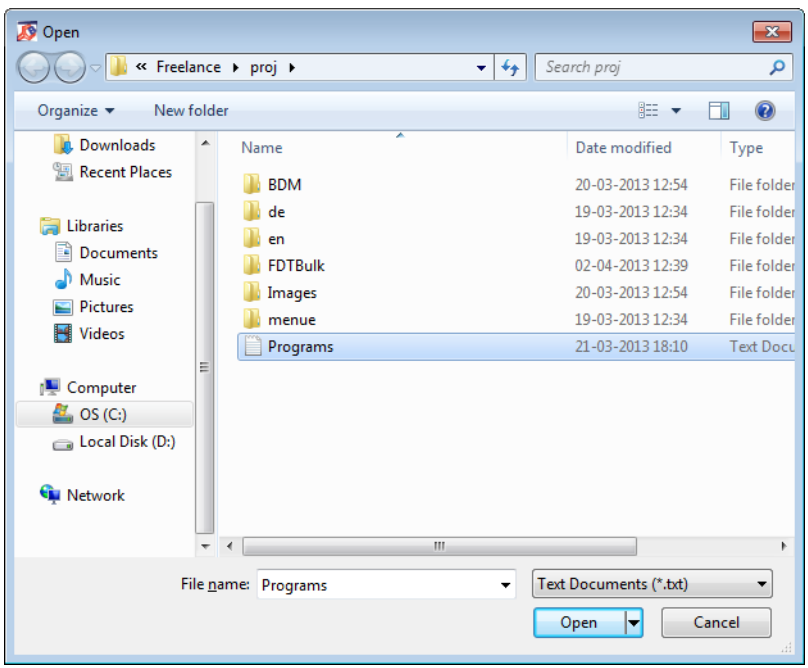
6.7.3 Import comment field

> Project > Documentation > Comment > Edit > Import

The text content of an originally exported comment field (saved as a text file *.txt) is inserted into the selected print job. Existing entries are completely overwritten. At this point, any required text can be inserted as long as it is changed into Unicode format.



ti061_us.png



ti062_us.png

6.8 Tables

6.8.1 Document types

Term	Type		Name
Documentation-description documents			
Title-/cover pages	EAA10	COV	Title page
Directories	EAB11	IND	Table of Contents
	EAB12	MAC	Setting the documentation job
General technical information documents			
Gen.techn. doc.	EDY10	MAN	Project tree
	EDY12	OBJ	Head confi./resource/task/program list
	EDY19	OBJ	Comment
	EDY20	HWM	Tree view
	EDY22	NET	Network
	EDY30	SYS	Graphic view
	EDY33	MOD	Module parameters
Technical requirements and dimension/design documents			
Tag lists, block schema	EEC10	TAG	Tag list
	EEC11	CR_T	Cross-reference tag list
	EEC20	FB_FBD	User defined function blocks FBD
	EEC40	FB_IL	User defined function blocks IL
Function-description documents			
Signal descriptions	EFP10	VAR	Variables list
	EFP15	CR_V	Cross-reference variable list
	EFP16	SVAR	Structured variables

Term	**Type**		**Name**
Freely available	EFQ10	I/O	I/O components
Freely available	EFL10	OV	Overview display
	EFL15	OV	Overview display cross references
	EFL19	OV	Overview display comment
	EFL20	GRU	Group display
	EFL25	GRU	Group display cross-reference
	EFL29	GRU	Group display comment
	EFL50	FGR	Graphic display (graphic)
	EFL53	FGR	Graphic display (parameters)
	EFL55	FGR	Cross-references graphic display
	EFL59	FGR	Comment graphic display
	EFL60	SFC_D	Sequential function chart
	EFL63	SFC_D	Sequential function chart parameters
	EFL65	SFC_D	Sequential function chart cross-references
	EFL69	SFC_D	Sequential function chart comment
	EFL70	TR_D	Trend display
	EFL73	TR_D	Trend display chart parameters
	EFL75	TR_D	Trend display cross-references
	EFL79	TR_D	Trend display comment
	EFL80	MSG	Global message processing
	EFL81	MSG	Local message processing
	EFL82	DSP	Display allocation
	EFL90	PG	Programmer display

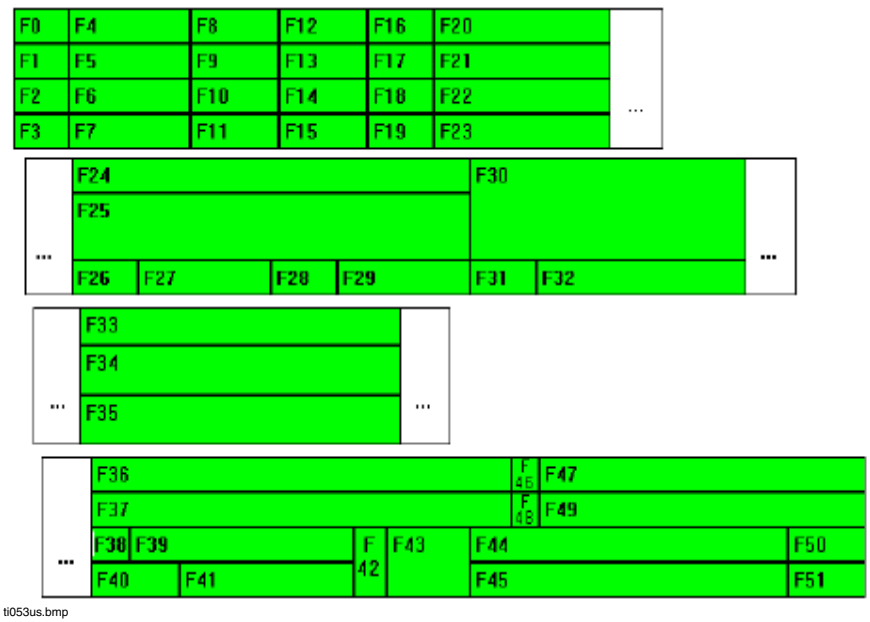
Term	Type		Name
	EFL93	PG	Programmer display parameters
	EFL95	PG	Programmer display cross-references
	EFL99	PG	Programmer display comment
Freely available	EFM10	ERR	Plausibility check error
Freely available	EFR10	SSL1	Signal sequence log automatic
Output	EFR13	SSL1	Signal sequence log parameter automatic
	EFR15	SSL1	Signal sequence log cross-reference automatic
	EFR19	SSL1	Signal sequence log comment automatic
	EFR20	SSLN	Signal sequence log, manual
	EFR23	SSLN	Signal sequence log parameter manual
	EFR25	SSLN	Signal sequence log cross-reference manual
	EFR29	SSLN	Signal sequence log comment manual
	EFR30	DCLD	Disturbance course log
	EFR33	DCLD	Disturbance course log parameter
	EFR35	DCLD	Disturbance course log cross-references
	EFR39	DCLD	Disturbance course log comment
	EFR40	OPL	Operational log
	EFR43	OPL	Operational log parameter
	EFR45	OPL	Operational log cross-references
	EFR49	OPL	Operational log comment
SW-specific Documents	EFT20	FBD	FBD program (graphic)

Term	Type		Name
	EFT23	FBD	FBD program (parameters)
	EFT25	FBD	FBD program (cross-references)
	EFT29	FBD	FBD program (comment)
	EFT40	IL	IL program (graphic)
	EFT43	IL	IL program (parameters)
	EFT45	IL	IL program (cross-references)
	EFT49	IL	IL program (comment)
	EFT50	SFC	SFC program (graphic)
	EFT53	SFC	SFC program (parameters)
	EFT55	SFC	SFC program (cross-references)
	EFT59	SFC	SFC program (comment)
	EFT60	LD	LD program (graphic)
	EFT63	LD	LD program (parameters)
	EFT65	LD	LD program (cross-references)
	EFT69	LD	LD program (comment)

6.8.2 Description of the fields or contents

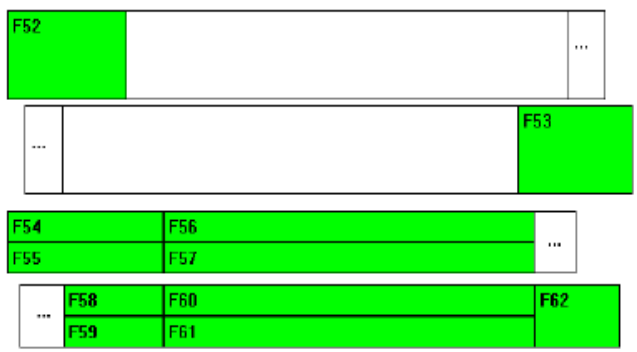
All fields used in the documentation must be unequivocal, that is, each field is given a serial number F1 to F62. This is used to describe the titles field, if no other entry has been made. For a definition of the fonts used, refer to Fonts, national languages and bitmaps in the drawing footer/header on page 317.

Drawing footer:



ti053us.bmp

Drawing header:



ti054us.bmp

6.8.3 Field names in the drawing footer/header

The following table shows the correlation between:

- Field designation **(FIELD)**
- Default variable **(CONTENTS)**
- Default title **(TITLE)**
- Description or contents of the field according to DIN **(COMMENT)**

FIELD	CONTENTS	TITLE	COMMENT
F0		Sta1	State 1 of printout
F1		Sta2	State 2 of printout
F2		Sta3	State 3 of printout
F3	Sta		Fixed text in frame
F4	$ObjDate	Revision1	Revision date1
F5		Revision2	Revision date2
F6		Revision3	Revision date3
F7	Revision		Fixed text in frame
F8		Date1	Date1 of revision
F9		Date2	Date2 of revision
F10		Date3	Date3 of revision
F11	Name		Fixed text in frame
F12		Norm1	
F13		Norm2	
F14		Norm3	
F15	Norm		Fixed text in frame
F16	Date		Fixed text in frame
F17	Resp		Fixed text in frame

FIELD	CONTENTS	TITLE	COMMENT
F18	Check		Fixed text in frame
F19	Norm0		Fixed text in frame
F20	$ObjCDat	Date0	Compilation date
F21	$PrjMan	Resp	
F22	$ObjS	Check	Checked by
F23		Norm0	
F24	Customer		Fixed text in frame
F25	#Logocust.bmp		Customer logo (bitmap) or text
F26	Origin		Fixed text in frame
F27		Origin	Original of
F28	Cre.f.		Fixed text in frame
F29		Cref	
F30	%LogoComp		Company logo (bitmap)
F31	Cre.b.		Fixed text in frame
F32		Creb	
F33	Title		Fixed text in frame
F34	$DocTypeName1	Title1	Drawing name1
F35	$DocTypeName2	Title2	Drawing name 2
F36	Custom.D.No.		Fixed text in frame
F37	$PrjOrdNr	Custom.D.No.	Customer drawing number
F38	==		Fixed text in frame
F39	$PrjName	Localization	Installation site
F40	Proj.No.		Fixed text in frame
F41	$PrjNr	Proj.No.	Project number

FIELD	CONTENTS	TITLE	COMMENT
F42	&		Fixed text in frame
F43	$DocT		Key to documentation type
F44	Doc.T.		Fixed text in frame
F45	$ObjKz		Object designation
F46	=		Fixed text in frame
F47	$ObjFunct		Function name of object
F48	+		Fixed text in frame
F49	$ObjLoc		Location of object
F50	P.		Fixed text in frame
F51	$PgNr		Sheet number
F52	%LogoLeft		Logo (bitmap) in upper left of header
F53	%LogoRight		Logo (bitmap) in upper right of header
F54	Name:		Fixed text in frame
F55	Comment:		Fixed text in frame
F56	$ObjName		Name of object (Path in the project tree)
F57	$ObjComm		Comment of object
F58	Start:		Fixed text in frame
F59	End:		Fixed text in frame
F60	$DocStart		Start of selected print scope
F61	$DocEnd		End of selected print scope
F62	$ObjId		Type of editor

6.8.4 Variables for drawing footer/header inscriptions

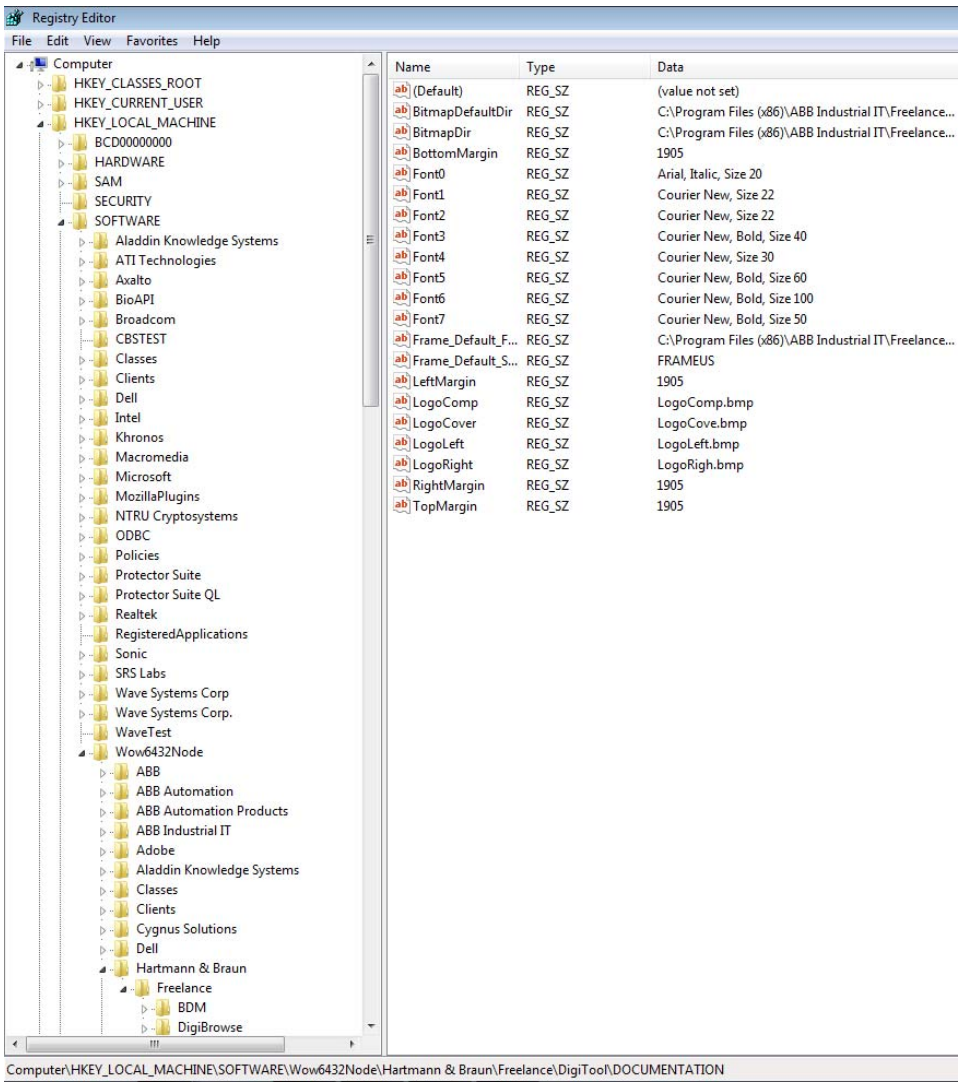
Command	Function
$	Avoids overriding through the project name
$DocEnd	End of the selection range in the project tree (depends on the documentation job)
$DocStart	Start of the selection range in the project tree (depends on the documentation job)
$DocT	DIN number of the current documentation type (hardcoded, language dependent).
$DocTypeName1	First part of the name of the current documentation type (hardcoded, language dependent).
$DocTypeName12	Name of the current documentation type (hardcoded, language dependent)
$DocTypeName2	Second part of the name of the current documentation type (hardcoded, language dependent).
$ObjCDat	Creation date of the current documentation object (if available).
$ObjComm	Short comment to the current documentation object (if available).
$ObjDate	Date of the current documentation object (if available).
$ObjFunct	Name of the current project tree object (if available).
$ObjId	Short Id to the current documentation object (hardcoded, language dependent, typically three capital letters).
$ObjKz	Documentation identifier of the current object.
$ObjLoc	Localization of the current object.
$ObjName	Name of the current documentation object (for example, name of the project tree object).
$ObjS	State of the current documentation object (CORRECT, INCORRECT).

Command	Function
$PgNr	Current page number of the print job.
$PrintDate	Date of the print issue.
$PrjComm	Comment of the current project (editable in the project header).
$PrjDate	Date of the current project (as shown in the project header).
$PrjMan	Manager of the current project (editable in the project header).
$PrjName	Name of the current project.
$PrjNr	Number of the current project (editable in the project header).
$PrjOrd	Orderer of the current project (editable in the project header).
$PrjOrdNr	Order number of the current project (editable in the project header).
%LogoComp	Default bitmap for your company.
%LogoCover	Default bitmap for your cover page.
%LogoCust	Bitmap for customer project
%LogoLeft	Default bitmap for the upper left corner.
%LogoRight	Default bitmap for the upper right corner.

Allocation of displays (only BMP files are permitted) is performed in the Windows Registry. Refer to Fonts, national languages and bitmaps in the drawing footer/header on page 317.

6.8.5 Fonts, national languages and bitmaps in the drawing footer/header

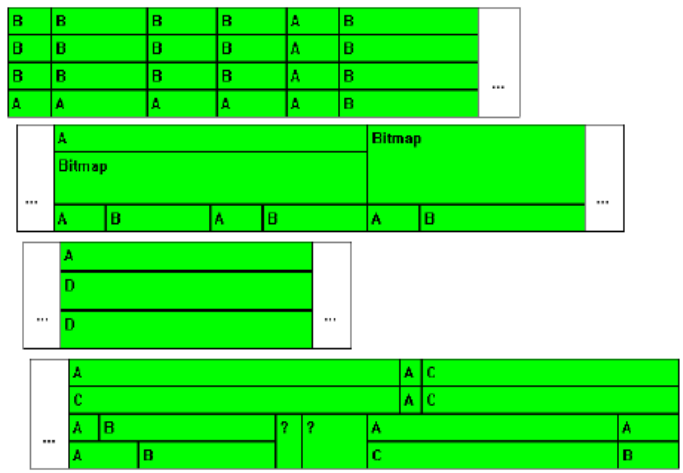
The fonts or the bitmaps used are defined in the Windows registry.



ti055_us.png

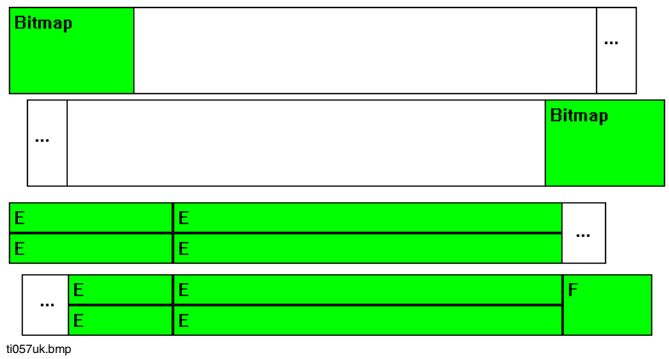
BitmapDefaultDir=\Program Files\ABB\Freelancebitmaps	Directory containing the bitmaps
BitmapDir=\Program Files\ABB\Freelance\bitmaps	Temporary directory
Font0=Arial, Italic,Size 20	in example A
Font1=Courier New, Size 22	in example B
Font2=Courier New, Size 22	in example C
Font3=Courier New Bold,Size 40	in example D
Font4= Courier New,Size 30	in example E
Font5=Courier New Bold, Size 60	in example F
Font6=Courier New Bold,Size 100	inscription of cover page
Font7=Courier New Bold,Size 50	Inscription of cover page
Frame_Default_File=\Program Files\ABB\Freelance\exe\frames.ini	path and file with the default field entries
Frame_Default_Sect=FRAMEUS	languages section in the file Frames.ini
LogoComp=LogoComp.bmp	Name of bitmap for company logo
LogoCover=LogoCove.bmp	Name of bitmap for cover logo
LogoLeft=LogoLeft.bmp	Name of bitmaps for the upper left corner
LogoRight=LogoRigh.bmp	Name of bitmap for the upper right corner

Using fonts in the drawing footer



ti056us.bmp

Using fonts in the drawing header



6.8.6 Presetting the field contents and titles

The presets for the field contents or titles are located in the file FRAMES.INI. In the English version the section FRAMESUS is used as default. Another section can also be selected.

Refer to Fonts, national languages and bitmaps in the drawing footer/header on page 317.

The file is located in the directory (**Freelance_installation_folder**) and can be edited with any ASCII editor, for example, EDIT of DOS or the EDITOR of WINDOWS.

The editor used must not append **any** control characters to the file!

The lines with the code FIELD_CONT_Pxx (xx=0 to 62) define the required field contents for the printout. The lines with the code FIELD_TITLE_Pxx (xx=0 to 62) specify a title for the fields. The title text can be superimposed at any time while entering the field contents to be printed, to show the user which entry is to be made in the fields.

Appendix A Glossary

Access	Parameters from function blocks and pre configured displays that either authorise or prevent certain sizes, values or actions for the user.
accumulator	The accumulator is a virtual storage location for storing the interim result of an operation in the instruction list.
Acknowledge	Operator action whereby one or more messages is/are acknowledged as seen and understood. Every action performed by the logged-in operator can be logged in the signal sequence log.
Acknowledge button	Button in the message line that enables the most recently received messages in the message line (VA, visual acknowledge) or message list (A, acknowledge) to be acknowledged without having to call up the message list.
Acknowledge level	The acknowledge level determines how events or messages on an operator station have to be acknowledged: 1) incoming and outgoing; 2) incoming only; 3) no acknowledgement.
Acknowledgement strategy	The Acknowledgement strategy is determined in the global message configuration and thus valid for all operator stations in a project. It specifies how messages and events must be acknowledged, therefore an acknowledge level is assigned to the message priorities.
Activate	Function of the hardware structure to take stations or parts of the hardware configuration into the current process. Hardware configuration that has been disabled with Deactivate are made active again. See also Deactivate

Active CPU module	See Primary
Actual time	The local time with any daylight saving time shift applied is the current time, that is, the time which the user sees on his or her watch. This time is used when setting the freelance time during commissioning is generally the time that is used on the Freelance user interface; for example timestamps of system and process messages are communicated with the system time and converted to actual time for display in Freelance Operations. See also Local time, System time, Daylight saving time.
Alphanumeric display	A dynamic symbol from the graphics editor for displaying analog values on the operator station.
Analog variable	Generic term for all non-digital variables with numeric formats such as real, integer, double integer or word.
Archive	An archive is the file in which the data from a trend display or log are stored. Depending on how data are logged, large volumes of data can be generated under certain circumstances.
Archive Browser	Freelance software program for a PC to visualize archives of trends and logs. Additionally archives can be exported as text files (csv format).
Area attributes	Area attributes include the color and fill pattern of the graphic objects like rectangle, polygon, ellipse and arc. These area attributes can be changed when making a display dynamic.
Attributes	An attribute is a key value associated with an object describing a particular characteristics of that object. In the case of graphic objects these are, for example, color, width and linear form.

Authentication key	The authentication key includes data corresponding to your licence rights for the Freelance software packages. To work with the Freelance software in the extent you have ordered, you will need your authentication key during installation. The Authentication key can be downloaded from ABB Software Factory (SOFA).
Auto/Automatic	Operating mode in which processing is carried out automatically through the program's control settings. The operator is not allowed to do any input in automatic mode. For operator input you must change to manual operating mode.
Automatic start-up	Autostart is a configuration parameter of the project element task. If Automatic start-up is activated, the task will start up automatically once it has been loaded into the process station. 'This causes all subsidiary program lists configured as ON to be run automatically.
Background	The background is the static part of a free graphic display.
Background color	A separate background color can be selected for the draw area for each graphic display and for the graphic pool.
Background objects	Function of the graphics editor. By default overlay dynamic objects the static graphic objects. The static components are drawn only once during call-up of the display, the dynamic components with each update. With the functions Foreground objects and background objects static elements are connected to a dynamic object and will be redraw with each update.
Band	The band is the value range of a variable that is displayed in a trend display. The limits of this range - the band start and band end - are configurable.
Band end	The band end defines the value for the end of the scale on the Y-axis in the single trend curve. See also Y scale

Band start	The band start defines the value for the start of the scale on the Y-axis in the single trend curve. See also Y scale
Bargraph	A dynamic symbol from the graphics editor. The bargraph is used to represent analog values by filling a rectangular column form either the top, bottom, left or right according to preference.
Block	The expression block is used in the documentation if it is not clear whether a function and/or a function block is meant.
Block	Summary of several similar entries in a list to perform an action on all selected elements, for example acknowledge of messages or export of data sets.
Block acknowledgement	Acknowledgement of several selected entries in the message list or hints list
Block class	A block class includes the functionality and appearance of the block. This information is defined with the interface, the program and the faceplate. With the installation of freelance software numerous standard block classes are provided. For specific process requirements, new block classes can be configured (user-defined function blocks). For the connection of OPC Servers new OPC block classes can be defined (OPC-FB-CLASS). A class itself cannot be processed in a process station. For execution, first an instance of the class must be created. See also Block instance.
Block instance	A block instance is the executable form of a block class. Different instances are identified by their tag names. Each instance works with values specific to that instance (local variables and parameters). See also Block class.
Block name	In contrast with functions, all function blocks are displayed with a block name. The block name is the same as the tag name in the tag list.
Block type	Identical to the type respectively class of a tag instance. As a name of a block type the short name of the type in the block library is used.

Blocked	SFC operating mode in which a transition blocks the advancement of the program run.
Bootstrap	The function Bootstrap is used to load the operating system into a process station.
Break point	A break point is a point in a program at which the user task is halted in the debugger.
Button	A dynamic symbol from the graphics editor. Buttons can be used to call up displays or tags, enter values for variables or display statuses.
Button field	A button field comprises one or more buttons. When one of the buttons is activated, the action must be confirmed.
Button type	Form of the button in the graphic display as a rectangle, in 3-D, as an ellipse or as a circle.
Carry out	Operator action performed on an SFC program whereby all active transitions whose transition criteria have been satisfied activate the steps next in sequence.
Channel	Input or output of an I/O module. The channels of an I/O module are numbered consecutively. See also Channel number
Channel number	Number of a channel in an I/O module
Check	Function of the project tree to check the configuration. While the function "check all" checks all selected elements regardless of their current check status, "check" covers only the elements of the selected part of the project that have not yet reached the state plausible. If the check was successful, the configuration can be loaded to the corresponding process, gateway and operator stations. The current check status is displayed with the configured colors for each element in the project tree. See also Check all.

Check all	Function of the project tree to check the configuration. All selected elements of the project are checked, regardless of their current check status, If the check was successful, the configuration can be loaded to the corresponding process, gateway and operator stations. The current check status is displayed with the configured colors for each element in the project tree. See also Check.
Check message	The result of the check function is displayed in the error list. The messages are divided into three categories: error, warning and hint. Errors must be removed in the configuration, so that the project can be loaded. For a warning the configurator must decide whether the project works also useful with this message. A hint indicates that inaccuracy was discovered by the system.
Checked	The state Correct is only assigned if no errors have been detected in the project tree or program when running plausibility checks on the selected project objects. Only project parts that have been checked without errors can be loaded into a station.
Class	Classes are data structures that contain variables and functions that can modify this data. A class is the description of an object, only by an instance of a class, a concrete object is created. See also Block class, Block instance.
Coil	The transmission of data with the Modbus protocol is done in registers (WORD) and coils (BOOL). A coil may be 1, 8 or 16 BOOLs in length.
Cold start	Restart of a resource, for example a process station. All current parameter data and internal cache values are deleted. The application programs are started new with the configured and corrected values. See also Warm start.
Cold start executed	The process station executed a cold start. The RUN/STOP switch of the controller is in state RUN, thus the controller goes into the state running.

Cold start stopped	The process shall execute a cold start. The RUN/STOP switch of the controller is in state STOP, thus the CPU will stop. After the switch is set to RUN, the cold start will be executed.
Cold start task	The cold start task is performed once when the process station is initialized or when a RESET is carried out in the controller. The cold start task is executed with the state change from cold start or cold start stopped to state running.
Colors	The colors that are not set through the Windows interface can be set individually in Freelance Engineering in the project tree and the program editors. The colors for trends in Freelance Operations can also be selected by the operator.
Comment	Detailed explanations to increase understanding can be added to the project and all project elements in the project tree, for example notes on how variables should be used, on a program's functionality or in general on the process field.
Commissioning	Operating mode of the engineering software in which there is an online connection to the process, gateway and operator stations; user programs can be loaded and operated. See also Configuration.
Common display pool P-CD	The common display pool P-CD is an element of the project tree in which those displays are configured which shall be accessible at all operator stations.
Configuration	Level in the project tree, under which the configuration of the software and hardware parts of a project are structured. The structuring of application follows the IEC 1131.3 and is divided into resources, tasks, program lists and programs.
Configuration	Operating mode of the engineering software in which user programs can be configured or modified. Configuration can be done without an online connection to any process station. See also Commissioning.

Connection	Online connection between Engineering station and process, gateway and operator stations and also between process stations via the Control Net (Ethernet TCP/IP)
Connectivity server	The data connection to the 800xA system is made by the connectivity servers (real-time data acquisition systems).
Control Aspect/Control aspect	The Control Aspect is an animated representation of the program, in which a tag or step or transition of an SFC program has been configured in Freelance Engineering, similar to the display of the program in the commissioning program
Control Builder F (CBF)	Software package for engineering station which enables overall configuration of the user program including the operation and observation level. Replaced by Freelance Engineering.
Control Net	All stations in a project, engineering, process, gateway and operator stations, are linked through the Control Net. The Control Net is based on the Ethernet standard with the TCP/IP transfer protocol.
Control room horn	A control room horn can be configured along with the field horn. This is done by entering a wave file (requiring a sound board) in the local event processing. When an event occurs with the configured priority, this wave file is executed.
Copy	The selected project objects are copied and can be inserted at any point.
Correct	With Correct a parameter of the selected tag is modified in commissioning mode. In addition, the new value is saved in the project, this means the configuration is changed. In opposite to this, when writing a parameter value, the project configuration remains unchanged.
CPU module	Main module of each process station in which the data processing and calculation of the user applications is performed.

Criteria window	Used for displaying the transition criteria and commands of an SFC program. Criteria windows can be configured both for steps (actions) and for transitions (conditions).
Cross communication	see Lateral communication
Cross reference	Cross references show which programs or displays the selected variable or tag is used in.
Crosshairs	In the graphics editor one of the display formats of the mouse cursor is termed crosshairs; here there is a horizontal and a vertical line each extending across the entire draw area. The point of intersection of the two lines corresponds to the cursor position.
CSV format	A standard ASCII format (comma-separated values) used as an export format for Freelance projects or parts of a project.
Cut	In contrast with deleted objects, the selected objects can be and actually are re-inserted into the program through Insert.
Data acquisition	The trends and logs are running on an operator station. This means that they are displayed and archived on the operator station. The acquisition of the necessary data can be carried out on an operator station or on a process station. On the operator station the data can be collected with a maximum cycle time of 1 second; for faster data acquisition an acquisition function block must be configured in a D-PS resource.
Data bank recovery	If the PC crashes in the course of a Freelance session, the latest position in the configuration can be recovered.
Data format	The data format specifies how a data type is displayed in the operation interface, like number of decimal characters (e.g. fff.ff for 340.05).

Data type	Data types are assigned to variables either directly in the program or through a variable declaration in the variable list. Along with the basic data types such as REAL or BOOL, user-defined data types can also be set up (structured variables).
Daylight saving time	In a Freelance system the Local time can be adjusted to daylight-saving-time automatically. For using Daylight saving time in user applications, a function block is available that converts a variable of data type DT (Date&Time) to daylight-saving-time. See also Actual time, Local time, System time
Deactivate	Function of the hardware structure to take stations or parts of the hardware configuration out of the current process. The configuration will be saved internally and can be made active again via the Activate command. With this function it is possible to take a plant gradually into operation. See also Activate.
Debugger	The debugger allows for convenient tracing of errors (bugs) in programs created with Structured text.
Default display	The default display is a display with a special significance for the display selection dialog in the context of display allocation. This allows a specific display to be called up for each tag when the F11-key is pressed.
Default display	The default display is a display with a special significance for the tag, step or transition. On the operator station this display can be called for a tag with the F11 key; in the context menu this display is shown at the first position.
Default task	The default task is executed as quickly as possible. This PLC mode has the lowest priority in comparison with the cyclical task. After one run-through the default task will start again automatically.
Default text	Text messages set up during configuration that are incorporated into the log, the alarm list or the hints list when particular events occur; these text messages may also be displayed in the dialog line.

Default value	Value, predefined from the system.
Delete	When program objects are deleted they are removed from the program permanently. When objects are deleted in the project tree they are removed permanently from the project.
Demo mode	If Freelance software is operated without a hard key, the application will automatically start in demo mode. In demo mode the software is fully functional and will run for 100 days. At the end of this 100-day period the software is no longer functional.
Design points	When a graphic object is selected, it is displayed with design points. Such an object can be resized by clicking on one of these points and dragging it while keeping the left mouse button pressed.
Detail	Function within configuration of the project documentation. For each documentation job the according scope and content is specified.
Detail view	View in the hardware structure. The selected object or module is displayed in the right part of the hardware structure with its details.
D-GS resource	If data from a Freelance project should be provided to other systems, a gateway station must be configured in the project. In the project tree an element of the type D-GS resource must be created.
Diagnostics	A number of diagnostic tools are running during operation, and these make it easier to trace a problem in the event of a malfunction. When an error occurs, a system error message results. There are also diagnostic options available through the diagnostics interface.
DigiVis	Software package for the operator station that enables operation and observation of the process with the aid of numerous standard functions. As from Freelance 2016 replaced by Freelance Operations.
Directory	During installation, directories are set up in advance for the storage of specific Freelance data.

Discrete	Dynamic graphic objects are output on defined x/y co-ordinates in conjunction with a maximum of 3 binary variables.
Display	Displays are pre-defined standard displays or can be free configured with the graphics editor. Displays are shown in Freelance Operations to visualize and operate the process. See also Graphic display; Overview display, Group display, Trend display, Sequential function chart display, Time scheduler display, WEB display, Log.
Display access	Through display access displays can be allocated to each tag as well as each transition and step of a signal function chart. These displays can then be called up quick and easily on the operation station for a selected tag, transition or step.
Display cycle time	The display cycle time determines the frequency with which the data in the message page, system display and the faceplates on the operator station is updated.
Display format	The display format determines the format in which a value of a variable or process signal is displayed in the operation interface.
Display object	Display objects can be simple or complex, static or dynamic symbols that can be created using the graphic editor and have parameters defined. A display object contains information for displaying the graphic symbol on the operator station.
Display selection	An operator action that alters the content of the display area of the operation interface.
Display selection dialog	Display of the operation line which enables displays to be called up rapidly. These displays are held in a fixed order which can only be changed using Freelance Engineering.
Disturbance course log	The disturbance course log is one of the state logs. It is used for logging chronological sequences of selected analog and binary variables.

Disturbance course log acquisition	Function block that records in the process station the values of up to 6 variables from a start point onwards and passes these values to the operator station.
Documentation	All configured objects of a Freelance project can be documented with a common layout with Freelance Engineering. The content, form and scope of the documentation can be freely selected.
Documentation job	A documentation job is a defining list specifying the user program areas that are to be documented.
Documentation management	Organization of all configured documentation jobs.
Documentation scope	All project objects positioned on the currently selected level of the project tree are documented.
D-OS resource	The displays and logs that are to be displayed on a Freelance operator station are configured in the project object D-OS resource.
D-PS resource	All the parts of a user program that are to run in a process station are configured in the project object D-PS resource.
Draw area	The draw area is the user's actual working area for creating and editing displays and programs
Drawing header / footer	The documentation of a Freelance project is printed out with a drawing footer and draw. header. This makes it possible for the draw. footer to be defined for the project and for each element of the project; system variables and bitmaps can be assigned to the individual fields
Duplicate	A selected graphic object or symbol is duplicated on the drawing surface without being saved in the clipboard as with copying. The duplicate copy is displayed on top of the original but in a slightly shifted position; it is also ready-selected.

Dynamic	When a display is made dynamic the static display, created using the graphics editor, has variables and tags from the user program introduced into it. This is achieved by accessing a shared database, thus helping to avoid errors in data entry.
Edit	Menu in Freelance Engineering through which fields, objects or programs are edited. It includes functions like calling programs or displays, general parameters, inserting, copying or deleting fields or objects.
Emergency mode	If the hard key cannot be detected during program start up or during normal operation (hard key is missing or defective) the system will enter the emergency mode. In this mode the software is fully operational for 100 days. During this time you have to exchange the defective hard key.
Emulator	The controller emulator allows testing and simulation of user programs even without connected hardware.
Enable	For an Signal function chart to run, an Enabled state must be detected. The enable command could be issued automatically by the user program or the operator.
Engineering station	PC or laptop with Microsoft Windows operation system and the software Freelance Engineering. Used by the system operator for configuration, commissioning, and documentation.
Error	Errors are reported in Freelance Engineering if a program is found that cannot satisfy the plausibility check due to syntactical errors. Errors can also occur when executing a task (e.g. division by zero); such errors are reported from the process station.

Error handling	Each runtime error detected during execution of a task command results in an entry regarding the cause and location of the error in system variables (NoMathErrCorrection) specially predefined for error handling. If automatic error handling is active and a recoverable error is detected, the affected task remains in the state running. In this case the error is ascertained only by evaluation of the system variables.
Error list	All plausibility errors are added to an error list. This error list contains all the errors for the currently selected level in the project tree.
Error task	The error task is the task with the highest priority (100); its function is the handling of errors in sections of the user program that form part of the common D-PS resource.
Ethernet	Serial bus system to DIN/ISO 8802, Part 3 (IEEE 802.3). Used for the Control Net.
Event	An event triggers a message or a control action.
Event function block	The name of the event function block. With this function block a logging process can be triggered based on specific process states.
Event log	Event logs are used for logging events such as messages, faults, switching actions and/or operator actions. Logging is performed automatically when the appropriate event occurs.
Event-controlled	A logging process or the transfer of a file is initiated by an event.
Exit	The menu choice Exit is used for leaving the current editor.
Expand	The project tree can be displayed either in expanded or compressed format by selecting the junctions. This has the effect of either opening up or closing down individual levels thus enabling the individual program parts to be displayed in a manageable way.

Export	Any parts of a project or program can be exported for use in other projects or programs. The export format used is CSV.
Export variables	Variables can be read from other process stations without any further configuration if they are declared with the export attribute.
Extension	All Freelance files are saved in the directory specified during installation; the default name is “Freelance”. The different types of files can be distinguished from one another by their extensions. For example, .pro, .csv, .arc and so on.
Extended User Management	A method to manage the user access rights on the operator stations and also for configuration and commissioning on the engineering station. It includes Use Local Account and Use Domain Account to log in Freelance system.
External parameters	External parameters for function blocks are parameters or values that are passed over from the user program by connecting a signal flow line or variable.
Faceplate	A faceplate is a graphic display or observation and operation of a tag, adjusted to the block class definition A faceplate is part of the block class. Faceplates for the standard modules are provided with the freelance software; for user-defined blocks own faceplates can be created. See also Block class.
Failure indication/Failure LED	Indicates at the process station the status of the power supply.
Fault message	Fault messages have the priority levels 1 to 3. These messages are used, for example, to signal that a limit value has been exceeded.
Favorites/Favorites menu	Any function and function block can be added to the favorites list for the configuration of application programs. Thus, these blocks can be accessed faster during configuration.

FBD program	An FBD program is a program created using the Function Block Diagram.
Field	In lists such as the tag lists, individual fields can be edited. Any change to particular fields such as Name or Type will result, after confirmation, in automatic system-wide changes to preserve consistency.
Field horn	Along with the control room horn, the field horn is one of the components of event processing. A horn block configured in a process station converts a signal from the field (process) into an event, with the effect that a message is then sent to the operator station.
Fill area	A dynamic symbol from the graphics editor. The fill level symbol is used to represent analog values with a polygon being filled either (optionally) from the bottom or top, from the left or right.
Fill pattern	Area attribute of a graphic object or symbol which allows the hatching style of the area to be modified.
Firmware update	Update of the controller firmware. The update allows error corrections and providing of new features.
Flash	Flashing refers to the cyclical pulsing of display objects.
Font type	Text attribute in the graphics editor.
Force	SFC operating mode whereby a transition forces an advancement when the program activates the transition but the transition criterion is not yet satisfied.

Force value	In order to allow program response and system behavior to be tested during commissioning, it is possible to force specific variables. To simulate a particular system state in the programs to be tested, variables on the I/O modules are assigned specific values and transferred to the user programs, that use these values instead of the actual values.
Foreground objects	Function of the graphics editor. By default overlay dynamic objects, the static graphic objects. The static components are drawn only once during call-up of the display, the dynamic components with each update. With the functions Foreground objects and background objects static elements are connected to a dynamic object and will be redraw with each update.
Format	Formats for logs: Depending on the log type the format of the log and its data can be freely configured or selected from a pre-defined set of formats. Formats for data: Data formats are used in graphic displays and logs to control the representation of different data types.
Framing error	In serial communications (e.g. Modbus, Telecontrol protocol), a framing error is the result of reading a string of symbols which are grouped in blocks at the wrong starting point.
Free colors	Name of that color group in the graphics editor which contains several colors in different intensities.
Freelance Engineering	Software package for the engineering station which enables overall configuration of the user program including the operation and observation level

Freelance Operations	Software package for the operator station that enables operation and observation of the process with the aid of numerous standard functions.
Freelance system	The sum of all the Freelance components: Freelance Engineering, Freelance Operations, add-on packages, hardware components. See also Process station, Engineering station, Operator station.
FTP	When trend archives or log files are transferred from an operator station to another subscriber on Ethernet, the transfer protocol used is FTP (File Transfer Protocol). The transfer may be manual, cyclical or event-controlled.
Function	On being executed, a function returns exactly one single data item. Functions do not hold any state information. Each time a function is called with the same arguments (input parameters) it therefore always returns the same result (output parameter).
Function block	More complex unit for performing open- or closed-loop control tasks. When a function block is used in a user program, a project-wide unique tag name must be given. After loading with this tag name a preconfigured operation is available for this block in the operator station.
Function block class	See Block class.
Function block diagram	A graphically-oriented user program for control tasks in which symbols representing functions and function blocks are arranged in a draw area; its pins are linked with variables from other programs via signal flow lines.
Function block instance	See Block instance.
Function key	The function keys (F1 - F12) form a standard part of the keyboard, and some of them have standard functions assigned to them.

Global variable	Global variables are the variables at all process stations of a Freelance project. In order to release a variable for system-wide use, the attribute Export is set to Yes.
Graphic area	The graphic area is the region in the editor in which the program itself is configured.
Graphic display	Additional to the standard displays and logs, free graphic displays can be configured with the graphics editor in Freelance Engineering to observe and operate the process on an operator station. Different static and dynamic graphic objects can be used to create graphical views of the plant and function and processes.
Graphic macro	A graphic macro consists of any combination of static and dynamic graphic objects. All attributes of the objects contained in the macro can be defined as macro parameters. Each time a macro is used, these attributes, which are part of the macro definition, are painted with their default values; all values specified as parameters can be individually adjusted at each instance of the macro's use.
Graphic object	Elements of the graphics editor to create free graphic displays to visualize and operate the processes in the plant. Examples of static objects are line, rectangle, circle; dynamic objects are bargraph, alphanumeric display, trend window etc.
Graphic pool	Buffer display in the graphics editor, used to exchange parts of graphic displays within a project.
Graphic symbol	Object in the graphics editor; made up one or more graphic elements and enables flexible dynamics.

Graphics editor	Editor in the Freelance Engineering software to create free graphic displays for the operator station.
Grid	In order to allow more accurate positioning during configuration, a background grid can be shown in the graphic area of an editor.
Group	Function of the graphics editor enabling several objects to be configured jointly. See Ungroup.
Group display	A group display is a combination of several faceplates. It provides the user with the opportunity to display related tags in the same display. After a displayed faceplate has been selected, the corresponding tag can then be operated.
Hard disk requirement	The amount of hard disk space needed for a trend archive or a log file is calculated, during the configuration. If it is established during loading the project to the operator station that the free storage capacity on the hard disk is not sufficient, an appropriate message is issued.
Hard disk space	During installation of the Freelance software hints are shown about the required space. Additional space will be required to work with the software. The actual amount of space required is determined by the size and content of your user program, like the amount and size of trends and archives. For details please refer also to manual **Getting Started.**
Hard key	Running Freelance software requires a hard key (USB port) and an authentication key. These enable the licensed version of the program to be used along with the licensed options.
Hardware structure	The part of Freelance Engineering in which the real hardware setup is configured for the project. In this part of the configuration the user programs are assigned to the process and operator stations, input and output values of the modules and devices are assigned to the parameters of the user applications and also the addresses are allocated on the Control Net.

Header	Parameter window for a project or project tree object in which universal details like name, short comment, and particular detail like cycle time for a task are entered. The drawing header and footer for documenting the parts of the user program are also configured in the header. During commissioning the D-PS resource, task and program lists should be started and stopped through the header.
Help	The Freelance help system enables operators to quickly call up relevant information onto the screen while they are working with the software.
Hint	The option is provided to configure a hint for the operator for every message of type Fault Message or Switch Message. These hints should be configured to inform the operator of the cause of the message, options for remedying the process anomaly and, where necessary, further operating hints. All hints are saved in the hint list.
Hint field	Display field in the message line showing whether or not there are any available hints relating to a message or event. Through this hint field one reaches a hint list showing any unacknowledged hints.
Hint list	The hint list contains a list of all hints that have arisen and not yet been acknowledged.
Hint message	Hint messages have the level 5. They appear only in the hint list and are purely for the operator's information.
Horn	A field horn or control room horn can be controlled by an event. A horn function is also provided that allows you to attach any control signals you wish to a horn.
Host name	A name identifying the computer. The host name must be unique within the network.

I/O channel	The variables which are to be linked with the process via the I/O modules are entered in the I/O channel allocation. The I/O module slot and channel resulting from the allocation is shown in the variable list.
I/O editor	The I/O editor is used in the hardware structure to assign variables of the project (variable list) to channel of a hardware module.
I/O module	Analog Output, Analog Input, Digital Output, Digital Input
I/O unit	Component of the process station. An I/O unit accommodates the I/O modules.
Icon	Windows terminology. Symbol for a program or a link.
Idle	Task state; the task is loaded on the process station but has not yet been started.
Import	In order to re-use parts of other programs or projects, such previously-exported sections can be imported. When importing on the project level, these units are deposited in the project pool. The format used for import is the CSV format.
Initial step	Every SFC (Sequential Function Chart) program starts with an initial step. A reset command always returns to the initial step.
Initial value	The value, which is taken during an initialization of a process station or after loading of new parts of the project.
Initialize	Deleting the section of the user program residing in the process station and re-starting the D-PS resource. The operating system is not affected.

Initialize all	With Initialize all everything is deleted, even the operating system.
Insert	One or more program objects that have been previously copied or cut are inserted at another position in the program.
Instance	See Block instance.
Instruction	Instructions are constructs in structured text which declare executable actions in process stations.
Instruction list (IL)	User program for control tasks similar to an assembler programming language. Its functional scope comprises operators that make a logical link between the contents of the accumulator and the operand. In addition, all function block types can be used.
Interface	The term interface refers to the point where two systems meet. The different features of the two systems are made compatible with one another through the interface (e.g. Modbus interface).
Internal parameters	Internal parameters for function blocks are parameters or values which are configured with a parameter dialog. Examples are tag name and limit values; they cannot be determined externally via a pin of the function block.
Interpolation	The trend display and the representation therein between two captured values can appear in three varying forms: with no interpolation (whereby only the data point is shown), as a line connecting two points, or as a staircase.
IP address	Address of an Ethernet node according to TCP / IP. Each Ethernet device gets an IP address. In the network dialog in the hardware structure the IP addresses are shown and can be edited.
Jump	A jump is brought about through a conditional jump instruction. Jumps are possible as operators in the Instruction List or as elements of the Sequential Function Chart.

Label	Jump markers from L001 to L999 (labels) can be entered in the Instruction List (IL); these serve as target addresses for jump operators. These labels may be entered in any order. In a ladder program implicit networks are defined by labels.
LatCRcv task	Lateral Communication Receive task. Global variables can automatically be read in other process stations if they are enabled for lateral communication through the attribute Export = Yes in the resource to be sent. The cross-communication necessary for receiving is performed automatically by the LatCRcv task.
LatCSnd task	Lateral Communication Send task. Global variables can automatically be read in other process stations if they are enabled for lateral communication through the attribute Export = Yes in the resource to be sent. The cross-communication necessary for sending is performed automatically by the LatCSnd task.
Lateral communication	Name for the cross-communication between the D-PS resources and thus between the process stations. For each resource up to 1400 bytes can be specified for lateral communication. The communication is handled automatically through the lateral tasks. See also Export variables.
Level	The project's structure is provided by the levels in the project tree. Project objects of the same type are all held on the same level.
Library	Classes of function blocks are grouped in a block library. In the standard library the blocks provided by Freelance are included. All user-defined function blocks are managed in a separate library. For the connection of OPC servers different OPC function block libraries can be configured and assigned to the OPC servers. The graphic symbols which can be used as macros for the creation of graphics displays are summarized into the macro-library.
License options	The license options indicate the functional scope authorised for the currently-installed Freelance software. See also Authentication key and Hard key.

Limit value	Limit values form the basis for generating events. Limit values are thus declared in the operation interface and/or in a log, e.g. when a particular value is exceeded. Particular events are specified depending on the type of limit value. The message is generated with, amongst other attributes, a specific priority and a message text.
Line attribute	The attributes of a line are its color, weight, line type, line start and end and the corner style. The border color of graphic symbols can also be changed in the context of making them dynamic.
Line break	Monitoring of an the input signal for voltage or current interruption. In general, a cable break detection triggers a system message.
Line number	Line numbers are assigned automatically in Instruction Lists, running sequentially from 1 to 1000. If empty lines or command lines are inserted, the command lines further down the list are automatically incremented by the number of lines inserted.
Line start / end	Attribute of graphic object line. It is displayed either with or without an arrow (optional).
Line style	Attribute of the graphic object line. The line or border color can be displayed either as dashes, semicolons or a continuous line.
Line type	Attribute of graphic objects. The line or border color can be displayed either as dashes, semicolons or a continuous line.
Load	During the commissioning mode programs or sections of programs that have passed the plausibility check without revealing any errors can be loaded into the according process, gateway or operator station. Therefore the elements or levels are selected in the project tree and the load procedure is started from the menu.
Local time	The local time is the system time with the time zone taken into account, but not the daylight saving time. In the process stations the local time is available in the system variable <name>.DateTime. See also Actual time, System time, Daylight saving time.

Log	Logs are used for documenting events, states and sequences from the process. The data they capture can be saved on the hard disk in the operator station, printed out on the printer or displayed on the monitor.
Log type	Three different types of logs are provided: Signal sequence log (SSL), Operation log (OP), Disturbance course log (DCL), Excel reports (REPORT).
Long text	A text entry up to 30 characters in length for providing a brief description of a function block or preconfigured display.
Loop	A loop is a repetition of one section of a program that is self-contained and can be performed more than once. Every loop must have a correct entry point and exit point; if this is not the case the result is an endless loop, which is not allowed. There must also be a condition to determine whether or not the loop is entered. This is implemented by means of a conditional jump.
Macro	A graphic symbol composed of one or more graphic objects or symbols that share a common name. This unit can be used over and over again and can be retrieved from the macro pool and saved in a macro library.
Macro library	Project-independent collection of graphic macros, created with the graphics editor and copied from the macro pool into a library.
Macro pool	The collection of all the macros in the project. The macros in the pool can be copied into a macro library from where they can be used in other projects.
Manual	Operating mode in which the tag can be operated from the operator station.

Measuring range	The measuring range defines the valid range for a value; measured values within the measuring range have a defined accuracy; at controller blocks the measuring range is used for internal normalization of the control algorithm. Within the faceplates the process values of the controller function blocks are shown with their measuring range. See also Scale range.
Measuring range end	The measuring range end is the physical value assigned to the transmitter's end point (e.g. 20 mA = 1000 l/h).
Measuring range start	The measuring range start is the physical value assigned to the transmitter's 0-point (e.g. 0/4 mA = 200 l/h).
Media colors	Name of a color group in the graphics editor; contains flashing colors.
Menu line	A screen region in the operation interface. The menu line remains visible at all times. Because of this, actions linked in the background with the menu can be performed in any operating situation.
Message	Process states and events can be configured as messages. When that event occurs a message is then sent to the operator station. Such messages are configured through the function blocks.
Message color	Colors for message display.
Message configuration	Message configuration is subdivided into a local and a global section. Locally, specific for each operator station, is selected which priorities and which sources will be reported. Additionally details for Message and hint list and also control horn handling are configured. In the global message configuration the acknowledgement strategy, determining which messages with a particular priority should be acknowledged and how this should be done.
Message filter	Definition of which message priorities are displayed in the message list.

Message line	The message line is part of the operation interface and is shown above every display on the operator station. The message line allows the operator to view any time the most recently received messages without calling-up the message list. The kind of the representation of the messages in the message line (list, tags, areas) can be firmly configured or be changed by the operator on the operator station.
Message list	In the message list the messages sent from the process stations to the operator station are displayed and managed. These messages may relate either to the Freelance system or to the automated process. In the message list a message can be selected and/or acknowledged, and the display assigned to a message can also be called up. In this way one can quickly obtain detailed information relating to the selected message.
Message list	In the message list at the operator station the messages sent from the process stations to this operator station and current system messages are visualized sorted and managed.
Message order	Definition of the position of the newest message in the message list. Thus, when set to 'Display newest message at top', any newly-arrived message will appear at the top of the message list.
Message priority	Messages are distinguished by different priority levels; priority S1, S2 and S3 are reserved for system messages, 1 to 3 for fault messages, 4 for switch messages and 5 for hint messages.
Message text	Additional output text of a message in the message box and in the message list.
Message types	In Freelance messages are subdivided into the following message types based on their significance for the process: system errors, fault messages, switch messages, hints and hint messages.

Mirror	Function of the graphics editor whereby one or more graphic objects can be displayed as either a horizontal or vertical mirror image.
Modbus	Using the Modbus protocol, process data can be exchanged with other systems through a serial or TCP interfaces of Freelance. This protocol allows data to be transmitted in either master or slave mode.
Module type	The modules of a process station are divided and named respect to their properties. As example, DI 724F stands for Digital Input and 724 defines the style. AI 723F means Analog Input.
Monitoring time	Parameter of an SFC (Signal Function Chart) program. The length of time after which the transitions/advancing criteria that follow a step but have not been satisfied will be reported.
Mounting	Configuration element of the hardware structure whereby a module is assigned to a slot in the process station.
Name	Names are textual labels for objects within the project. A name should be unique within the project, and every object must be given a name.
Network	Part of the hardware structure allowing station numbers and IP addresses to be entered and configured on the Control Net (Ethernet / TCP/IP).
Network address	See IP address.
Network board	Plug-in Ethernet card for a PC through which connection to the Control Net is effected.

Node	Graphical representation of a project object in the project tree. The color of the nodes is used to visualize their processing status.
Normal	Operating mode in the Sequential Function Chart in which a transition or step is processed normally. This means that the transition is neither blocked nor forced and steps are not switched permanently on or off.
Not executable	Task state in a process station, it is not possible to execute the task. The task is loaded into the process station and the automatic error detection system has detected an error such as division by zero; the task cannot be executed.
Offset mode	This operating mode is used to determine which offset value is being used in the time scheduler.
Offset value	Value added to the time scheduler's set point.
Online	To commission a Freelance project the process, gateway and operator stations must have an online connection to the engineering station.
OPC block class	Block class for connection of external systems via an OPC interface. The instances of an OPC block class are formed with the OPC items.
OPC block library	In an OPC block library OPC block classes are managed together. Each OPC block library can be assigned to specify OPC servers in the project.
Operand	In an Instruction List program the accumulator combines the contents of the accumulator with the operand in accordance with the operator command and save s the result in the accumulator. The operand here may be a constant or a variable.
Operate and observe (man machine interface)	A general term describing the process-control approach in an industrial process.

Operating mode	Status (on, off, manual, automatic) of the tags in the process station, respectively enabling particular operator actions or commands from the user program.
Operating system	Basic software in a resource. In the process stations the operating system (firmware) enables the communication with other process stations and enables the calculation of the user programs. Further functions of the operating system are program management, test routines etc.
Operation log	Cyclical, manual or event-related logging of up to 200 variables within a configurable text.
Operator	The person who holds responsibility at any given point in time for controlling process events through the control system.
Operator	The operator is a standard function in the Instruction List IL such as Load LD or Save ST. The operator combines the accumulator with the operand. The result of this operation is saved back in the accumulator again.
Operator action	An operator action is an action on the part of the operator through which a process state or process value is changed from the operator station.
Operator action	Intervention in the process by the operator. An operator intervention can only be carried out if the access parameter has been set to ‘Yes’ for this task. Furthermore, the operating rights for the operator currently logged in must be allocated.
Operator interface	The operator interface is the sum of all the display objects and operating objects at the operator station.
Operator station	An operator station based on a PC with a Windows operating system and the software Freelance Operations. Used for observation and operation and alarm messages, trends, archives and logs. See also D-OS resource

Optional parameters	Optional parameters are function block attributes such as short text, long text, dimension, conductivity and limit values. These function block parameters are set to default values and need not necessarily be entered.
Options	Operating mode in SFC programs whereby the progression through the following steps is controlled through actions on the part of the operator.
Overlap	Function of the graphics editor for moving graphic objects either to the foreground or the background if they are overlapping one another.
Overview display	Preconfigured display for quick selection of displays and/or logs.
Page layout	Specification of the output format of a log.
Parameter	Parameters are configurable attributes effecting the processing and display of function blocks, displays and logs.
Parameter dialog	Dialog window through which parameters can be entered and modified.
Parenthesis depth	Parenthesis depth of logical operators in the Instruction List from level 1...8.
Parity	Simple error detection system using the parity bit (none, odd or even).
Permanent	Scheduler operating mode whereby all sections of the set curve are run through systematically.
Permanent off	SFC operating mode whereby a step is switched off permanently. Activating it has no effect.

Permanent on	SFC operating mode whereby a step is switched on permanently. The step is not necessarily active but will nonetheless be kept in the processing sequence.
Pin name/Pin identifier	Each pin of a function block is displayed with a short identifier that indicates the function of this connection, e.g. EN for enable.
PLC mode	A task with this operating mode is processed as quick as possible in the process station. This gives rise to variations in processing time according to system loading.In order that the cyclical tasks are processed correctly, the SPS mode task is assigned the lowest possible priority (50) within a D-PS resource.
Points	Design points for the graphic objects line and polygon can be repositioned, deleted or added.
Pool	Area in the project tree containing incorrect or currently not used project parts that may be introduced later into the process again.
Position of a graphic symbol	The position of graphic symbols can be shifted either continuously according to an analog variable or in discrete steps according to binary values within the display area.
PRAM	Memory in the Freelance process stations that is used for the configuration data. This memory is cold-start resistant.
Primary	The currently-active CPU module or the active controller of a redundant process station is named as the primary.
Print job	The output of documentation is always controlled by a so-called print job, i.e. there must always be a job selected before printing. The contents of the print task is defined in this job.

Printer	Printers can be connected to the operator and to the engineering station. Different printer types can be used depending on the requirements; for example color printer for hard copies, line printer for message logs.
Printer setup	Printer selection, setting of options.
Priority	The multitasking of several tasks with the same cycle time is controlled through priority (not to be confused with message priority). The lowest priority that can be assigned to a user task is 51 and the highest is 99.
Priority levels	Messages received from the process stations may have differing levels of importance in terms of effect on the smooth operation of the process. The Freelance system provides six different priority levels for messages. S1, S2, S3 = System messages, 1 to 3 = Fault messages, 4 = Switch messages and 5 = Hint messages.
Process image	A temporary storage location for the task where the task variables are read before the task is run and where those variables are written back again after the task has finished. These variables are flagged with the @ character.
Process image variable	A variable where the access is via the process image. These variables are flagged with the @ character.
Process message	Process messages inform the user about disturbances and specific status changes in the Freelance system. A distinction is made between system messages, fault messages, switch messages, hints and hint messages. The message types are further subdivided according to their importance for the process and according to the priority levels.
Process station	The process station contains the CPU module, which performs the processing of the configurated programs.

Processing	Control over which function blocks are included in processing is provided by the parameter Processing On/Off. If the task is configured for autostart and the program list already activated, the block will be executed if set to Processing On.
Processing number	In the function block diagram, FBD, the key numbers displayed in the bottom-right hand corner of the functions and function blocks indicate the processing sequence within this FBD program.
Processing sequence	The processing sequence shows the sequence in time in which the project objects (e.g. program list, program or function block) contained in a task are processed.
Program	The structural unit of a project within which the open- and closed-loop control tasks can be configured. The Function Block Diagram (FBD), Instruction List (IL), Ladder Diagram (LD) and Structured Text (ST) are provided for defining programs.
Program element	A program element is the smallest unit of a program that can be displayed, namely functions and function blocks, signal flow lines and, sometimes, sections of such lines along with variables to be read or written.
Program list	Structural element of the project, which controls the assignment of user program sections to the task. In the task the cycle time is specified, in the program list, the order of execution of the various programs is controlled.
Project	The configuration of a system is created and stored with the software and hardware parts in a project.
Project documentation	All the components of a project can be documented directly. This is achieved by assembling the required documentation types in a documentation job.

Project name	A single project name is assigned to all the various parts of the user program. This project name will be used later for documentation purposes and also forms the file name for configuration.
Project object	Part of the project configuration. Some of these project objects are visible, such as, resource, task, program list, program, function and function block, and also hidden objects such as process image, variables and messages.
Project tree	An organization tool used for structuring a project in the form of a tree.
Project version	The project version is made up of three parts as follows: Part 1 (major) states the number of bootstraps; Part 2 (minor) is incremented by 1 every time program sections are loaded; Part 3 (patch) states how often parameters have been corrected. All three parts can be read through system variables. A version control system allows a comparison to be made between the engineering station and process station.
Quad Monitor	The Freelance system supports connecting to up to four monitors with matched licenses, and user is able to customize the preferred display types on each monitor.
Rack	Framework for the process stations modules; used for central unit and the I/O unit(s).
Rack ID	Identification code of the rack. Set with the coding switch at the link module.
Radio button field	A dynamic symbol from the graphics editor allowing up to 25 values to be preset by selecting a radio button.
RAM	Random Access Memory. Identifies the memory of the CPU. The RAM can be described with data and read again. The RAM of the Freelance process stations is battery buffered.

Raster/Snap	If a raster or snap is active, objects can be placed at fixed positions only; thus a better alignment of the objects is possible.
Redundancy connection	The physical connection of the two redundant partner that is used for the synchronization of the redundant data.
Redundancy link	Via the redundancy link, the synchronization of the redundancy partner, Primary and Secondary, is carried out.
Redundancy synchronization	The first synchronization between the redundant controllers runs in parallel to “Load whole station” or “Load changed objects”. First the operating system is loaded into the secondary, then the user program and finally the current process data. Once the first synchronization has been successfully achieved, both redundant stations pass to state SYNC. From now on data is adjusted cyclically between the two controllers.
Redundancy toggle	A redundancy toggle takes place when on the active controller (Primary) detected a problem or the toggle has been actively triggered, for example, by an operator action on Prim/Sec-switch of the controller. The previous Secondary is the new Primary and takes over the further execution of the programs. A redundancy toggle is bumpless, all states of the user programs remain unchanged.
Redundant process station	A process station that is constructed and configured redundantly. A redundant process station can be built only with dedicated controller types.
Redundant task	For a redundant resource redundant tasks, but also non-redundant tasks can be configured. All programs of a redundant tasks are running redundant.At the end of the tasks, the newly calculated values are transferred to the secondary. All variables for a redundant task must be written via the process image.

Reference point	Coordinates of a static/dynamic display object in a graphic display with particular related functions such as mirroring.
Refresh cycle	General term for the update time of data; for I/O signals of the refresh cycle describes the data exchange rate between the I/O module and the CPU, for fieldbus communication the data exchange rate between two devices.
Register	The transmission of data with the Modbus protocol is done in registers (WORD) and coils (BOOL). A register may be 1, 8 or 16 WORDs in length.
Repeat time	The time at which an SFC program is restarted.
Report printer	The report printer is the printer connected to the operating station. A maximum of two printers can be connected to one operator station.
Reset	Control action on a function block or SFC program to reset the counter in an internal register or reset the SFC program to initial step. Resetting of a process station via the reset button of the controllers triggers a warm- or cold start of the process station.
Resource	Structure element of the projects to control the assignment of the user application s to the process and operator stations. In the hardware structure the software resource are assigned to the operator and process stations to specify the communications for the load process.
Restart time	The restart time is the point in time at which the SFC is to be started. Unlike repeat time, restart time represents a single point in time for starting the SFC.
Ring memory	Fixed-size memory area which is overwritten after a specified length of time.

Rotate	graphics editor function whereby one or more graphic objects are turned through 90 degrees counter-clockwise about the point central to the design points.
Rounded corner	A linear attribute in the graphics editor. Corner rounding allows the corners of polygons, rectangles or continuous lines to be displayed with a rounded form.
Run task	The run task of a process station is executed once when the controller is started.
Run/Stop indication	Indication at the controller, for example a LED, that visualizes the current processing state of the process station.
Run/Stop switch	With the RUN / STOP switch on the process station the function processing in the process station can be stopped and then started again. The RUN/STOP indication of the controller shows the processing status - stopped or running. A process station in state Stop state is still able to communicate over the Ethernet to the operator and engineering station. For safety reasons it is not possible to set a process station to state "running" via software from the engineering or operator station when the switch on the controller is in position "Stop".
Running	State of the task in the D-PS resource, the task is executed. The operating system is loaded and the RUN/STOP switch on the CPU module is set to RUN.
Scale end	Defines the scale end for values of a tag.
Scale range	With the scale range the conversion of a process value is determined to a different value range. Signals of different sources must be scaled to a common area in order to compare them with each other. Within the faceplates the process values of the according function blocks are shown with their scale range.
Scale start	Defines the scale start for values of a tag.

SD card	Secure Digital (SD) card is a nonvolatile memory card. It is a family of solid-state storage media. The SD card allows to backup and restore the application or run the firmware update.
Secondary	The currently-passive CPU module or the passive controller of a redundant process station is named as the secondary.
Security Lock	It enables access right settings for specific user groups on the operator stations and also for configuration and commissioning on the engineering station.
Selection	Operating technique from the operation interface. Selection is used to activate displays or to select tags for operation.
Selection area	A dynamic symbol from the graphics editor. The selection area can be used to configure display calls or actions. By selecting a selection area in Freelance Operations, the tag or display entered is called up or a write or acknowledge action is initiated.
Selection list	Selection lists provide a way of accessing the project database in order to insert variables, tags or text into a program, display or function block.
Sequence selection	An object of the Sequential Function Chart for formulating alternative program paths. A sequence selection divergence represents the start of an optional branch in the sequence.
Sequence selection convergence	An object of the Sequential Function Chart for formulating alternative program paths. A sequence selection convergence represents the end of an optional branch in the sequence.
Sequential function chart	The Sequential function chart (SFC) is used for programming sequence control functions. It can be used to break down complex tasks into manageable units and to describe the flow of control between these units. See also SFC program.

Sequential function chart display	Standard display that allows a program sequence set up using the Sequential Function Chart to be operated on a Freelance operator station.
SFC program	An SFC program (Sequential function chart) is a project object within a task which, unlike the Instruction List, processes the allocated programs with the aid of a structure (Grafcet).
Short comment	A text entry in the header of a project element which may be up to 159 characters in length. Unlike the fuller comment for a project object, only short comments can be entered here.
Short term archive	When a trend display is selected, the short-term archive for the individual signals is displayed first. The short-term archive holds 200 values, i.e. max. 6 x 200 values per trend display. For older values the system will then automatically return to access the archive file.
Short text	A text entry up to 12 characters in length as a brief description of a function block or pre configured display.
Side effect	A configuration change that has an effect on the processing of the higher-level task or D-PS resource. Any tasks or D-PS resources flagged in this way must be re-loaded in order that the configuration change can come into effect.
Signal colors	Name of a color group in the graphics editor; contains deep signal colors to be used to visualize event states consistently in the operation interface.
Signal flow	Identifies the direction of data processing in an FBD program. The signal flow in an FBD program generally runs from left to right.
Signal flow line	Signal flow lines are the graphic representation of the signal flow in an FBD program. A signal flow line shows the connection between a variable and a function block or between different function blocks.
Signal range	Specifies the value range of an input or output signal. Usual signal ranges are 0...10 V, 0...20 mA and 4...20 mA.

Signal sequence log	Configurable logging of system errors, fault messages, switch messages, operator hints and operator actions.
Signal sequence log 1	Logging of system errors, fault messages, switch messages, operator hints and operator actions with continuous printing.
Signal sequence log N	Logging of system errors, fault messages, switch messages, operator hints and operator actions, recording these in a log file.
Simultaneous sequence convergence	An element of the SFC for formulating parallel program sequences. A simultaneous sequence convergence represents the end of parallel sequence branches.
Simultaneous sequence divergence	An element of the SFC for formulating parallel program sequences. A simultaneous sequence divergence represents the start of parallel sequence branches.
Sort	Lists can be re-ordered through sorting.
Sort criteria	The sorting order is defined by the sort criterion.
Sound board	Plug-in card for a PC that plays back acoustic files (wave files).
Standard display	Along with the free graphic displays created with the graphics editor there are also several standard displays available; to display these in Freelance Operations you only need to configure a tag and a few parameters. The standard displays in Freelance are the system display, the overview display, group displays, trend displays, SFC displays, time scheduler display, WEB display and logs. Additionally for each tag a faceplate is available without any configuration.
Standard value	Same as default value.
Start address	In the Modbus configuration the address of the first variable to read or write.

Start characteristics	Definition of the response of the system software at system startup.
Start time	The start time is the time at which the SFC program is to be/was started for the first time.
State	See Operating mode.
State log	The function of state logs is to record process states cyclically. This includes the cyclical logging of the state of a tag or the logging of sequences in the process.
Static colors	Name of a color group in the graphics editor; contains the general colors of the graphics editor.
status line	The status line shows information about the current processing, such as the name of the selected program, the logged-in user, the license status or information about active filter in tag and variable list.
Step	An element of the SFC controlling actions in conjunction with a transition. In Freelance a string of FBD and/or IL programs continues to be processed until the next transition is satisfied.
Step list	List of all the steps in an SFC program.
Step name	Free-form text with up to 8 characters, displayed and documented in the graphic object of a step in an SFC program).
Stop Task	The stop task is executed once when the state of a process station state changes from Running to Stopped. The stop task is not activated until all other tasks have been stopped. A restart cannot be performed until the stop task has finished executing. If the stop task does not contain any programs, it will not be started.

Stopped	Task state in the process station, the task is not executed. The operating system and configuration are loaded, the RUN/STOP switch of the controller is set to STOP.
Structured variable	Structured variables are variables with a data type of a user-defined, so called structured data type. The data type consists of a combination of named components of differing data types.
Structured data type	A user defined data type that can be composed of several basic data types such as BOOL or REAL.
Structured text (ST)	User program for control tasks in a text-oriented programming language. An ST program consists of instructions. All the Freelance functions and function blocks can be used
Subnet mask	Masking in the main network for a possible lower-level network.
Substitute value	Configurable fixed default value which is, in case of overrun or underrun of a process signal, used for further calculation instead of the real value.
Switch message	Switch messages have the priority level 4. Messages of this type are used to report switch events, e.g. "Valve open" or "Valve closed".
Syntax error	Syntax comprises a set of formal rules needed by a programming language in order to function correctly. Syntax errors occur when these rules are disobeyed, and they lead to a message through a dialog box or to a plausibility error.
System bus	All stations are connected through the system bus. This system bus is based on the Ethernet standard with the TCP/IP transfer protocol.
System log	Archive file in which all actions on the engineering station are logged, if Security Lock is installed.
System message	System errors have the top priority of 0. Messages with this priority cannot be either configured or changed by the user. System errors generate messages about error states in the system itself.

System message	System messages have the highest priority level and are subdivided into three message groups S1 to S3. These messages cannot be configured or changed by the user. System messages are used to indicate fault states of the system (hardware) itself.
System structure	Main view of the hardware structure in which all configured process, gateway and operator stations are displayed.
System task	General term describing all the preconfigured tasks in a process station resource.
System time	Current time on the stations, used for internal transfer of time stamps. For time synchronization of the stations of a project, the system time is used. System time is equal to Greenwich Mean Time GMT (UTC). See also Actual time, Local time, Daylight saving time.
Tag	A tag is an instance of a pre configured function block. A tag name must be assigned to each configured function block. All the tags in a project are listed in the tag list.
Tag list	This list holds the names of all function blocks and elements of the hardware structure configured in the project.
Tag type	Brief description of the function block in the function library.
Task	The tasks control the processing of the user program parts in a process station. A distinction is made between cyclical tasks and those which are processed as quick as possible.
Task cycle time	The task cycle time determines the intervals at which the functions and function blocks of the programs will be executed. Interval time is entered in TIME format, for example T#2s500ms for 2.5 seconds.
Taskbar	Windows taskbar, that is used to call the Windows start menu.

TCP/IP	Transmission protocol for Ethernet bus system. Corresponds to layers 2-4 of the ISO/OSI seven layer model.
Terminal	A terminal is the physical connection point on an I / O module.
Terminal emulation	For diagnostic purposes, a debug terminal can be connected to a Freelance process station. When installing the engineering software, a terminal emulation program is installed in the folder <Freelance_installation_folder>\exe\Terminal.exe.
Terminal pin	The terminal pins are the inputs and outputs of the blocks. A distinction is made between mandatory and optional pins. Mandatory pins require supply via a signal flow line in order that the module can operate correctly, while optional pins do not. This difference is represented in the programs of the Function block diagram by making the optional pins shorter. Through the parameter definition of constants certain optional pins become completely redundant. As with the signal flow lines, the color and line width of the pins contain information about the data type required or set.
Text attributes	Describes the format in which text appears. Font size, font, text orientation and colors can all be selected.
Text edit	A single-line text entry as a graphic object, text.
Time master	In general, the process station with the lowest station number is the time master. This station transmits the time once per minute, and so synchronizes the clocks of all other stations. Small deviations are compensated in the received stations by adjusting the system clock, for larger deviations the time is set by a time jump.
Time scheduler	With a time scheduler, analog values defined as time-dependent are output; these values can be transferred to other functions, e.g. to a controller as a setpoint input.

Time scheduler display	The time scheduler display displays the time scheduler's state and enables it to be controlled by the operator. This display consists of a trend region for tracking set points and process values, a state field and the associated faceplate.
Time synchronisation	All Freelance stations are equipped with a real-time clock. When the system is running, the clock times must correspond throughout the system, so that for example entries can be made in logs and trends in the correct time sequence. For this purpose, the system time on all stations via the Ethernet bus is synchronized. For the PCs in the Freelance system time synchronization can be enabled or disabled individually. If the time synchronization for a Freelance Operations station is disabled, trends can not be displayed correctly.
Time Zone	The time zone (TZ) is calculated from the difference between the Greenwich Mean Time (GMT) and the local time (TZ = GMT - Local Time).
Timeout	Timeout means a period which may last an operation before it is terminated with an error. In the Freelance configuration maximum waiting times (timeout values) are generally configured as INT values in the range of 0..10000 ms.
Transition	An element of the Sequential Function Chart that activates the following step (altern. situation) or steps (parallel situation). Within a transition in Freelance, an FBD or IL program will continue to be processed until the transition criterion is satisfied.
Transition criteria	Condition that must be satisfied in order for a step to activate subsequent steps. See also Transition.
Trend	The function of a trend is to provide a graphic representation of values using a time axis. In Freelance this is done by capturing values in the process station and transferring them to the operator station in blocks; alternatively directly in the operator station variables can be collected and visualized as trends.

Trend data acquisition	Function block that captures the signal values of the linked variables and transfers them in blocks to the operator station.
Trend display	The trend display is used to display values graphically using a time axis. A maximum of 6 trends can be shown in one trend display.
Trend server	The Trend server is used for the acquisition of variables. The Trend server makes these variables available for the trend displays in the operator stations.
Trend window	Trend display on the engineering station that, like the value window, can be called up when in commissioning mode. The variables are shown as a trend curve.
Unchecked / Incorrect	Each configuration and configuration change first generates elements with the status not checked = incorrect. With the function plausibility check the configurations will be checked and, if no errors were found, transferred to the status Checked or Correct.
Undo	This function, which appears under Edit, can be used to reverse the last change, e.g. deletion of an object.
Ungroup	Function of the graphics that resolves the combination of elements that were done via the group command. See also Group
User FB variables	The pins and parameters of a user-defined function block are declared in the interface editor of the user function block as user FB variables. Within the class description of the block - the block program and faceplate - this user FB variables are used in the same way as global variables in a project.
User function block	A user function block is a function block created by the user that can be reused in other programs in the same way as other function blocks. Also called User-defined function block.

User function block pool (P-FB)	Within this pool, the user-defined function blocks are configured in the project tree in the form of programs. The pool P-FB contains the class definitions of all user function blocks in a project.
User function blocks menu	From this menu the operator can select the checked user function blocks that are configured in the user function block pool (P-FB).
User Management	Used for user access rights management. It includes Security Lock and Extended User Management.
User program	The sum of all the open- and closed-loop control tasks in the process, which have been configured with Freelance Engineering for the process stations.
Value	Numeric value of a variable that must be entered in a specific format depending on data type.
Value region	General term for the allowed values of a variable. In the description of graphical user interfaces also part of a representation or display where values are shown, for example the trend display of Freelance Operations.
Value window	Display in the commissioning phase allowing several variables in a variety of numeric formats to be displayed together.
Variable	A variable is the data carrier for a value or piece of information. The variables represent the link between the programs and displays; information based on a data type is transported from one location to another through a variable.
Variable list	The variable list contains all the variables configured in the project as inputs, outputs and flags from the programs and modules.
Variable window	A dialog within commissioning that enables one or more variables to be allocated to the value display or trend display.

Version	Along with the version number of the installed software there is also a project version number which can detect and report whether the version of the user program loaded matches the version in the engineering station.
Version control	A version control is performed to check which project is loaded on the selected process station and whether this project is the same as that currently open in Freelance Engineering.
Version error	State announcement of a resource during commissioning. Significant differences exist between the programmes running in the station and the actual configuration. This error can be removed only by a loading the whole station.
Vertical line	An element of the Sequential Function Chart enabling a vertical connection to be established between steps and transitions.
Visual acknowledgement	Only the messages in the message line are acknowledged. The message itself remains in the message list and must also be acknowledged there.
Waiting time	The waiting time TWA is the minimum duration of time the SFC program will remain in a step.
Warm start	New start of a resource, for example, a process station. In the Freelance process stations the configuration stored in the PRAM is preserved. With intact battery the working data are also preserved, so that the application programs are further calculated on the actual values. See also Cold start.
Warm start executed	The process station executed a warm start. The RUN/STOP switch of the controller is in state RUN, thus the controller goes into the state running.

Warm start stopped	The process shall execute a warm start. The RUN/STOP switch of the controller is in state STOP, thus the CPU will stop. After the switch is set to RUN, the warm start will be executed.
Warning	During the plausibility check of project parts errors were detected that are not serious enough to prevent execution.
Watchdog	Component for monitoring functions. When a failure is detected, an appropriate response is triggered. For example, a connection monitoring may trigger a redundancy toggle.
WEB display	After calling a WEB display on the operator station, the local WEB browser is launched and the configured WEB site is activated.
Write	Transaction from the engineering or operator station whereby a modified parameter from a tag is written to the process station. Unlike correction, such changes are not saved in the project file.
Y scale	With the Y-scaling the range of a variable is specified that is displayed in a trend display. The limits of this range - the band start and band end - are configurable and can be modified by the user on an operator station. See also Band start, Band end.
Zoom	4 zoom levels are provided in the graphics editor, each of these levels displaying the selected zoom region with greater magnification than the preceding level.
Zoom region	The zoom region is used to select the screen region to be displayed in the graphics editor's display area.

Index

E

F

G

H

I

L

M

N

O

P

Q

R

S

T

U

V

W

—

www.abb.com/freelance
www.abb.com/controlsystems

—

We reserve the right to make technical changes to the products or modify the contents of this document without prior notice. With regard to purchase orders, the agreed particulars shall prevail. ABB does not assume any responsibility for any errors or incomplete information in this document.

3BDD012503-111 A