

Menu Items

- [Text list, page 55](#)
- [Transition, page 405,](#)
- [Trace, page 431,](#)
- [Visualization, page 445](#)
- [Visualization manager, page 496.](#)

To **rename** an object in the Project Explorer, click on the object entry to open an editing field or use the <F2> shortcut.



A device object has to be added using the special command: [Add device, page 331,](#).

3.10.3 Opening an Object

Icon:

This command can be used if an object located in the Project Explorer is to be opened for examination or editing.

When the object is selected in the Project Explorer, the "Open" command can be accessed in the context menu.

Alternatively, the object can also be opened by double-clicking it.

The object opens in the corresponding editor.

In online mode, a dialog opens which prompts again about the view to be used to display the object (e.g. implementation/instance 1...).

3.10.4 Implement Interface...

This command is used to update the [implemented interfaces, page 38](#), for a function block.

This means that if the interface definition was modified, e.g. another method was added, this change can be moved into the function block by selecting the function block object in the POU or Project Explorer window and executing the 'Implement interfaces...!' command.

Example:

1. Interface_1 contains the GetName interface method of type STRING.

Upon creation of the FB1 and FB2 function blocks, Interface_1 was in each case implemented and thus, GetName is automatically inserted as POU method also below FB1 and FB2.

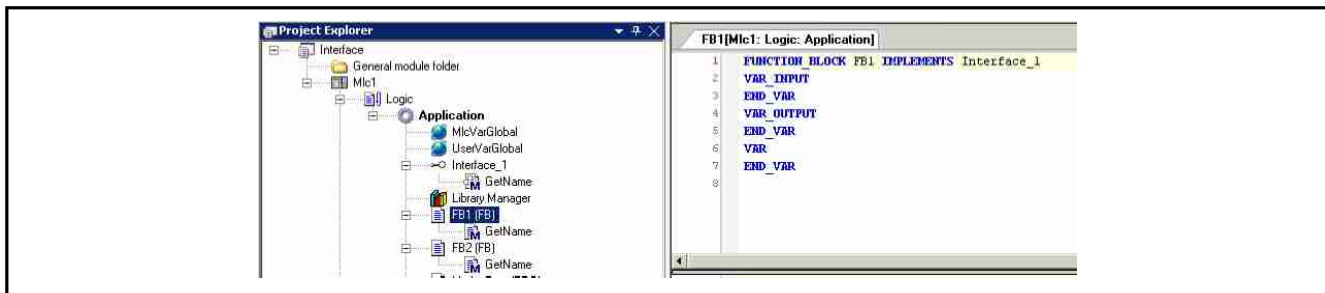


Fig.3-201: FB1 and FB2 implement ITF1

2. Interface_1 is now extended by method GetID:

Menu Items



Fig.3-202: Interface modification

3. FB1 and interface ITF1 are updated with the "Implement interface" command (context menu):

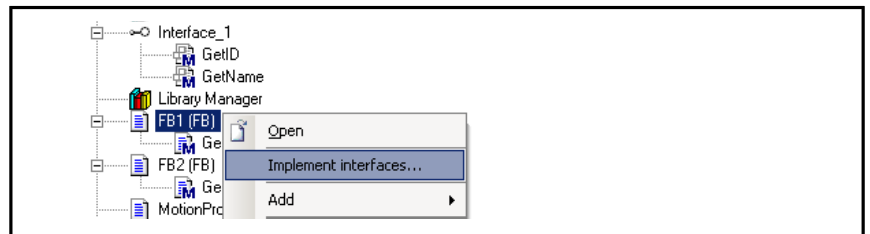


Fig.3-203: Updating the interface in FB1

4. The selection window for the implementation language of the new GetID method appears.

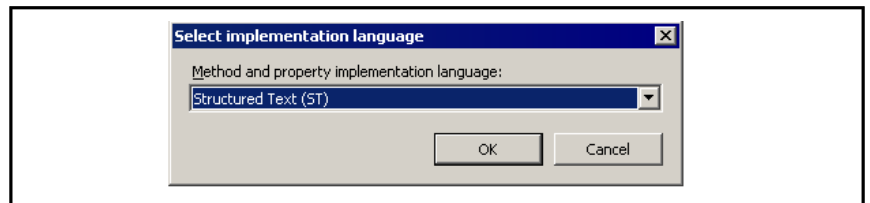


Fig.3-204: Selecting the implementation language

The GetID interface method becomes the GetID POU method of the FB1 function block.



Fig.3-205: Interface in FB1 is updated

3.10.5 Set Active Application

This command is used to define the [applications](#), page 66 located in the **Project Explorer** that are to be active.

If an application is currently selected in the Project Explorer the command can be selected in the context menu and by default in the project menu. After it is executed, the object entry of the application is displayed in bold in the tree.

Menu Items



Note that in all cases online actions apply only to the "active application"!

3.10.6 Object Properties

Object Properties, General Information

If objects (e.g. devices, applications or visualization objects) are selected in the Project Explorer, information on these objects can be found in the "Properties" window.

Examples:

- [Device properties, page 238](#)
- [Application properties, page 240](#)
- [Object properties, general information, page 242](#)
- [File properties, page 245](#)
- [Visualization object properties, general information, page 245](#)
- [GVL properties, network variables, page 246, sender](#)
- [GVL properties, network variables, page 248, receiver](#)

Device Properties

Icon:

Context menu: **Properties**.

The properties of the device currently selected in the Project Explorer are displayed on tabs. A few settings can also be changed in these dialogs:

Highlight the device node and activate the command to open the Properties dialog.

"General" tab

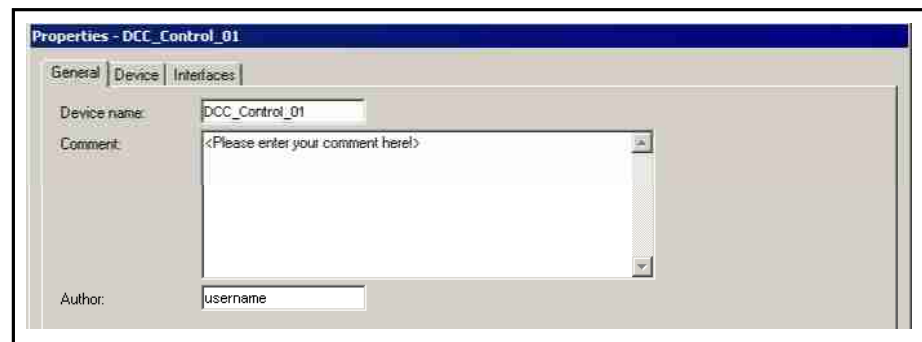


Fig.3-206: General device properties

- **Device name:** Name of the device in the project tree, cannot be edited. The device name can only be changed in the project tree.
- **Comment:** Insert a comment here that describes the device in detail. This entry is optional.
- **Creator:** Enter the name of the project creator here. This entry is optional.

"Device" tab

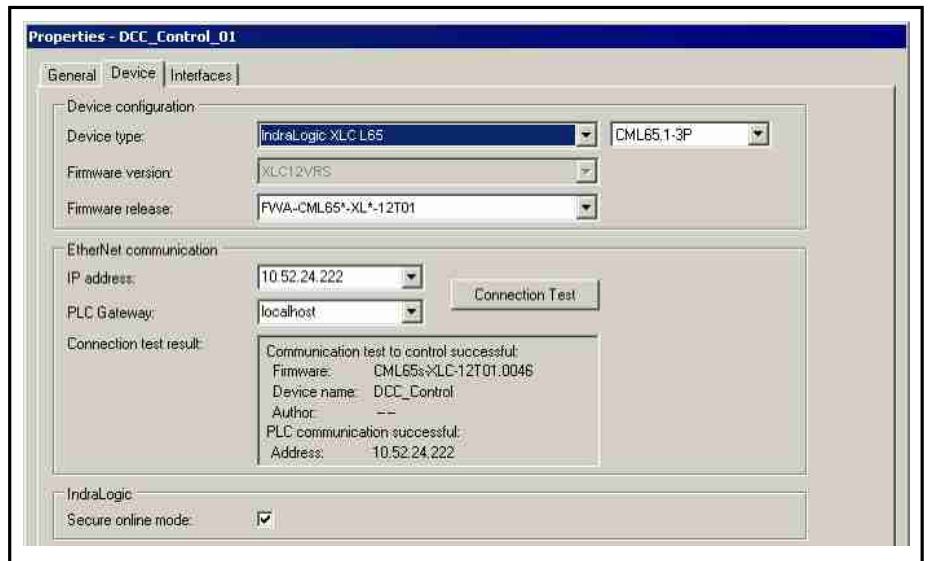


Fig.3-207: Device properties Device

With this dialog, the device IP address can be modified, the PLC Gateway can be set and a connection test (ping) to the device addressed can be carried out.

The **Secure online mode** option adds an additional prompt for activities that are relevant to security. See also [Debugging, Overview, page 126](#).

"Interfaces" tab

The entries in this dialog are for information purposes only and cannot be edited.

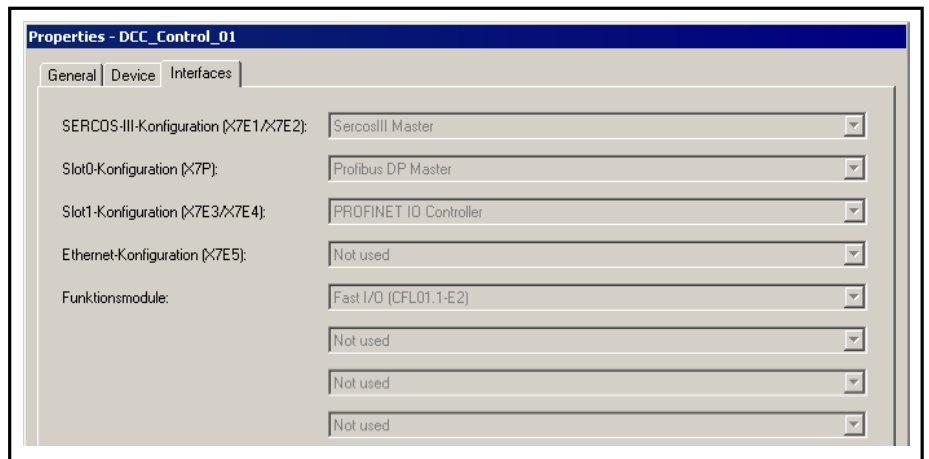


Fig.3-208: Interfaces tab

- **SERCOS III configuration (X7E1/X7E2):** Current SERCOS III configuration that can be selected when the device is added. The selection includes "Sercos III Master" or "Not used".
- **Slot0 configuration (X7P):** Current PROFIBUS configuration that can be selected when adding a device. "PROFIBUS DP Master", "PROFIBUS DP Slave" or "Not Used" can be selected.
- **Slot1 configuration (X7E3/X7E4):** The following communications protocols can be configured using this interface:
PROFINET IO controller, PROFINET IO device, EtherNet/IP scanner, EtherNet/IP adapter, Ethernet interface or "Not used".

Menu Items

- **Ethernet configuration(X7E5):** Engineering interface that can be configured as EtherNet/IP adapter or "Not used" for exclusive engineering interface.
- **Function modules:** Depending on the control type, different function modules are available for extension.
Here note the numbering, "function module 1" to "function module 4" which have to correspond to the switch positions for the slot numbers and the position of the function modules.

Application Properties

Icon: 

Context menu: **Properties**.

Highlight the "Application" node and activate the command to open the "Properties" dialog.

"Common (general)" tab

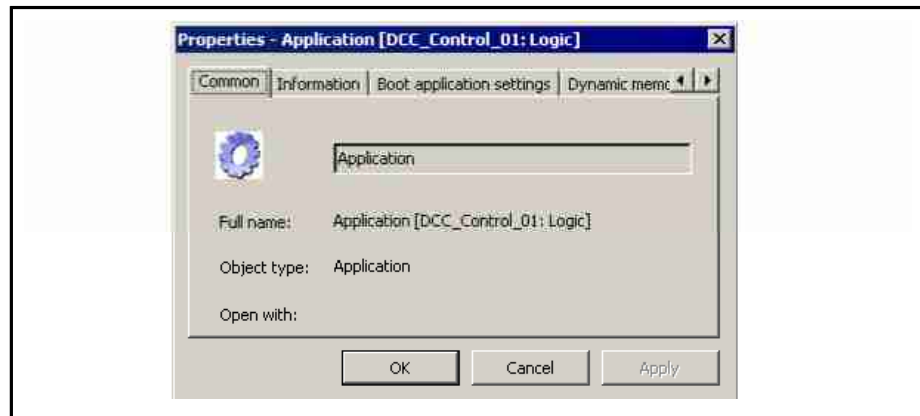


Fig.3-209: General object properties

- **Full name:** Name of the object as it appears in the Project Explorer.
- **Object type:** Type of object (e.g. "POU", "Application", "Interface" etc.),
- **Open with:** Type of editor in which the object is to be displayed or edited.

"Information" tab

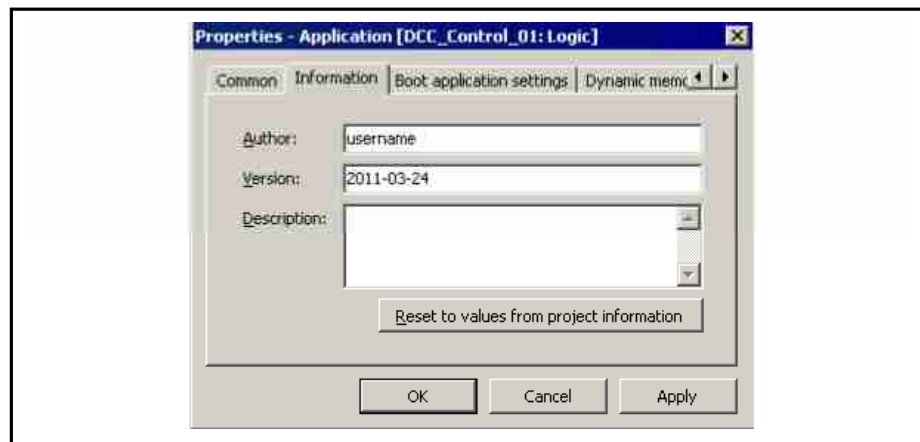


Fig.3-210: Information tab

The user can use this tab to store information about the application: author, version and description of the application.

"Reset to values from project information" deletes this information.

"Boot application settings" tab

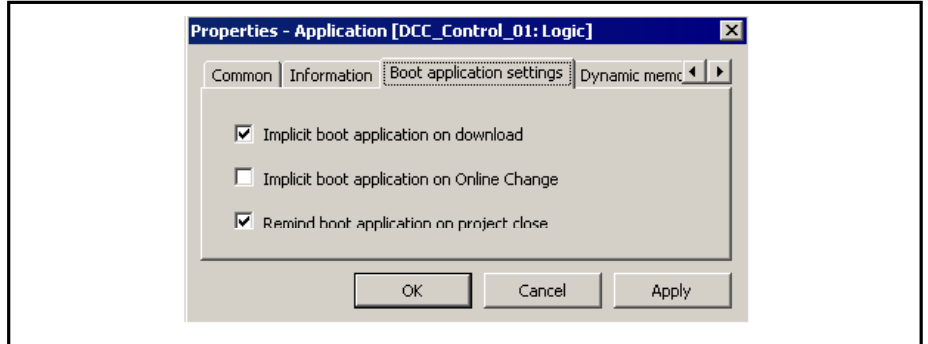


Fig.3-211: Boot application settings tab

During every download the active application is automatically saved as a file called <application>.app in the target system directory.

A boot application is started automatically when the control is started.

To do this, the application project on the control has to be available in a file <ProjectName>.app. To create the file, use **Debug** ► **Generating a boot application**.

In this tab, the user can specify how to handle the boot application file:

- Transfer the boot application implicitly into the control with each download.
- Transfer the boot application implicitly into the control with each online change.
- When the project is closed, ask the user if the file is to be transferred into the control.

"Dynamic memory setting" tab

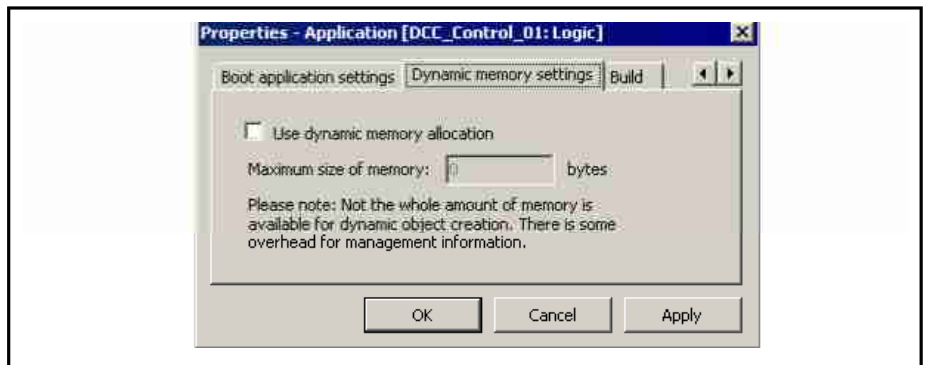


Fig.3-212: Dynamic memory setting tab

"Use dynamic memory allocation":

Select this option if memory is to be reserved for the application automatically. In this case, specify the desired "Maximum memory size" (bytes).



Note that not the entire memory is available for the dynamic object generation but one part is used for management information.

The current settings are transferred to the control in every download or online change.

"Compile" tab

In the "Compile" tab, options regarding object compilation can be selected.

Menu Items

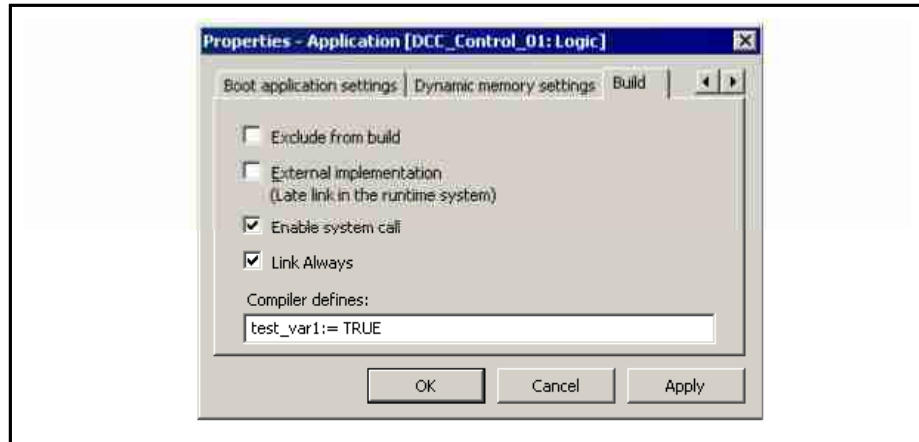


Fig.3-213: Compile tab

- **Exclude from compilation:** The object is not considered in the next compilation run.
- **External implementation (late link in the runtime system):** When the project is compiled, no code is generated for this object. The object is only considered if the project is running on the target system, providing the object is present there (e. g. in a library).

- **Enable system calls:** The reason is that in contrast to IndraLogic 1. x, the ADR operator can now be used with function, program, function block and method names and it replaces the INSTANCE_OF operator.

For more information here, see [Function pointers](#), page 557 in Data types.

However, it is not possible to call function pointers within "IndraLogic 2G". To enable a system call (runtime system), the "Enable system calls" option has to be selected for the function object.

- **Always link:** The object is highlighted in the compiler. Thus, it is always included in the compile information. This means that it is always compiled and loaded to the PLC. This option is relevant if the object is located in an application or is referenced by libraries that are also located in an application. The compile information is also used as the basis for the selectable variables of the symbol configuration.
- **Compiler definitions:** "Defines" (see {define} instructions) and conditions for object compilation can be entered here (conditional compilation). In the "Declaration" section is a description of the available ["conditional Pragmas"](#), page 546.

The "expr" expression, which is used in such Pragmas, can also be entered here and several entries are possible in a list of items separated by commas.

For example, it might be desired to compile an application depending on a value of a certain variable. (In the example above: test_var1.)

Object Properties

Icon:

Context menu: **Properties.**

Various objects in the Project Explorer have a "Properties" dialog in the context menu.

Highlight the desired object in the Project Explorer and activate the command to open the Properties dialog.

"Common (general)" tab

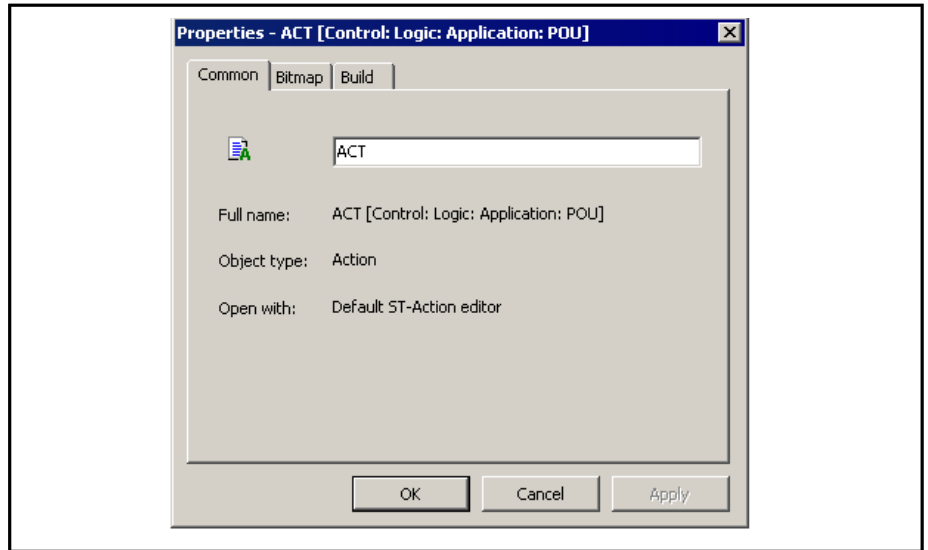


Fig.3-214: General object properties

- **Full name:** Name of the object as it appears in the Project Explorer. The object can be renamed here.
- **Object type:** Type of object (e.g. "Logic", "Program", "Task configuration" etc.),
- **Open with:** Type of editor in which the object is to be displayed or edited.

"Bitmap" tab

The "Bitmap" tab allows a bitmap to be assigned to the associated object, e.g. a symbol assigned to the graphical symbol in the function block diagram for a function block, an action or a program.

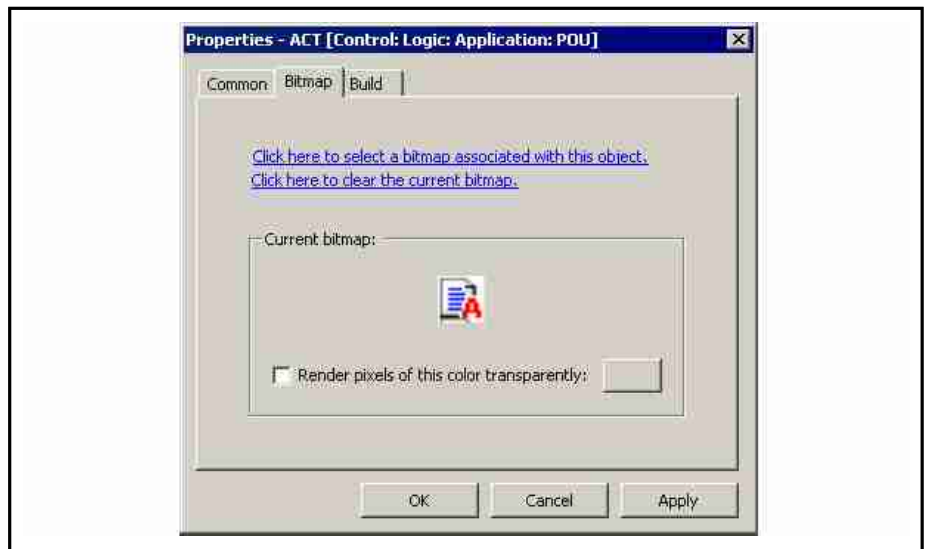


Fig.3-215: Object properties, bitmap

Here, assign to or remove an image file from by means of which it is displayed in the graphical view of the [library manager, page 367](#), and in the [toolbox of the FBD/LD/IL editor, page 355](#).. Transparency of the figure can be obtained by selecting a color that is then displayed transparently. For that purpose, the "Transparency color" option has to be selected. Then, the default dialog for selecting a color can be opened using the rectangular button on the right side.

"Compile" tab

In the "Compile" tab, options on the object compilation can be selected.

Menu Items

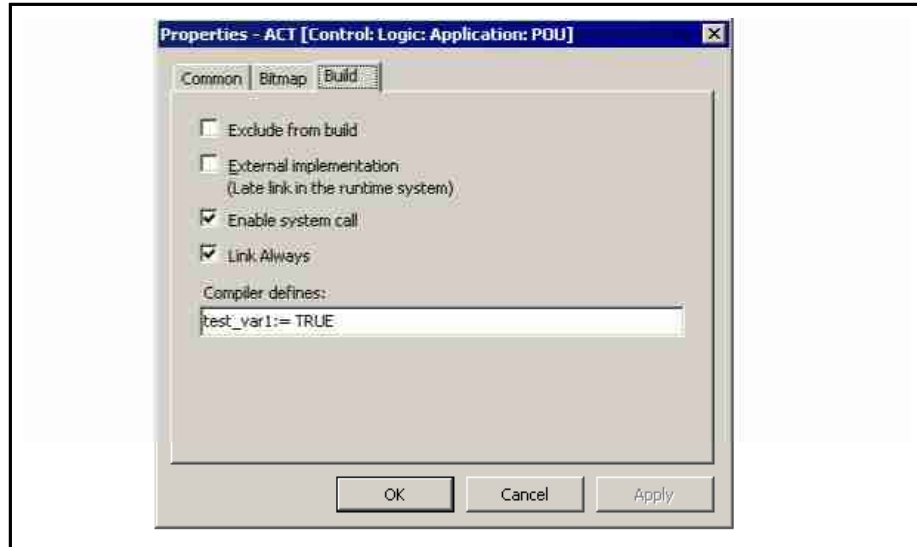


Fig.3-216: Compile tab

- **Exclude from compilation:** The object is not considered in the next compilation run.
- **External implementation (late link in the runtime system):** When the project is compiled, no code is generated for this object. The object is only considered if the project is running on the target system, providing the object is present there (e. g. in a library).
- **Enable system calls:** The reason is that in contrast to IndraLogic 1. x, the ADR operator can now be used with function, program, function block and method names and it replaces the INSTANCE_OF operator.

For more information here, see [Function pointers](#), page 557 in Data types.

However, it is not possible to call function pointers within "IndraLogic 2G". To enable a system call (runtime system), the "Enable system calls" option has to be selected for the function object.

- **Always link:** The object is highlighted in the compiler. Thus, it is always included in the compile information. This means that it is always compiled and loaded to the PLC. This option is relevant if the object is located in an application or is referenced by libraries that are also located in an application. The compile information is also used as the basis for the selectable variables of the symbol configuration.
- **Compiler definitions:** "Defines" (see {define} instructions) and conditions for object compilation can be entered here (conditional compilation). In the "Declaration" section is a description of the available ["conditional Pragmas"](#), page 546.

The "expr" expression, which is used in such Pragmas, can also be entered here and several entries are possible in a list of items separated by commas.

For example, it might be desired to compile a function block depending on a value of a certain variable. (In the example above: test_var1.)

File Properties

Menu Items

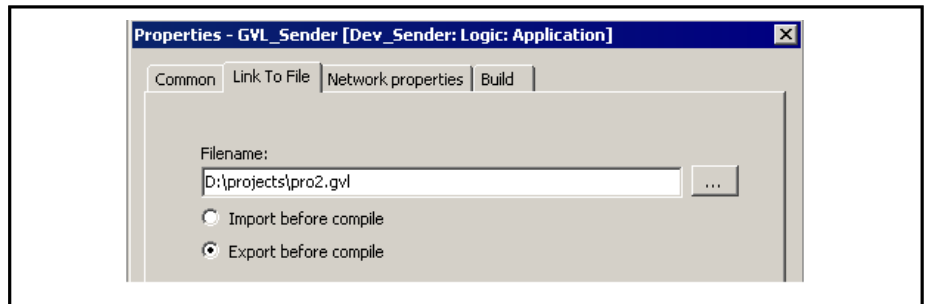


Fig.3-217: Properties dialog, linking to file

Global variable lists, page 52, can be defined by means of an external file in text format. Such file can be generated using the export functionality that is available in the Properties dialog of the relevant variable list.

- If the **Export before compilation** option is selected, a file with the extension "gvl" is stored automatically in every project compilation (e.g. via <F11>) in the path contained in the File name field.
- If the **Import before compilation** option is selected, an existing export file can be imported in every project compilation. This allows for the import of a GVL that has been defined in another projection in order to set-up communication via [network variables](#), page 248,.

Properties of Visualization Objects

Icon:

Context menu: **Properties.**

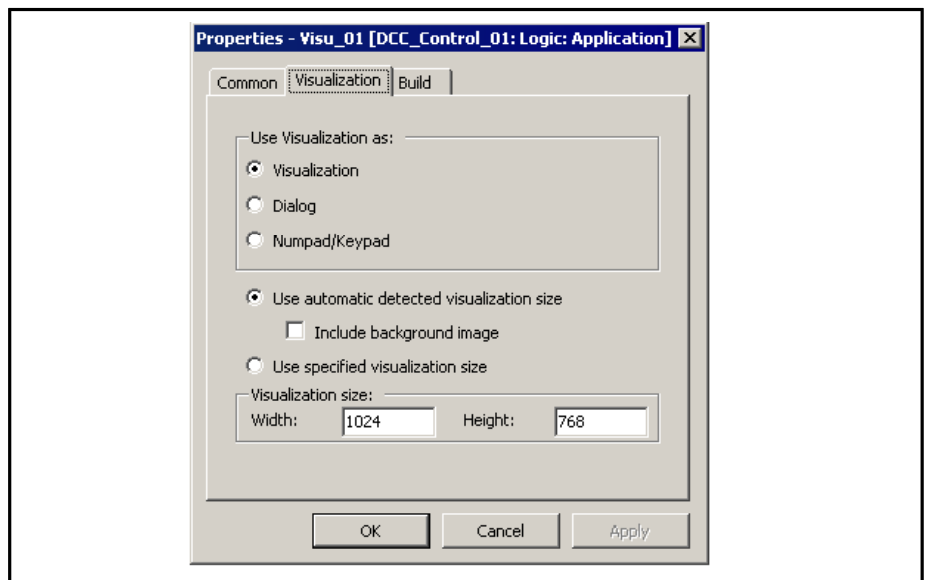


Fig.3-218: Visualization tab

For the currently selected visualization, the following settings are displayed, which can also be changed:

- **Use the automatically determined visualization size:** The size of the visualization which shows all currently contained visualization elements and the background image is determined. If the "Include background image" option is, however, deactivated, the background image is cut off if it exceeds the area including all visualization elements.

Menu Items

- **Use specified visualization size:** The visualization size is defined by the values displayed above (width, height) whereas it is not considered whether all visualization elements and the background image fit into this area, i.e. are completely visible.
- **Visualization size:** current "width" and "height" of the visualization (number of pixel)

Please note that you can define in the [visualization manager, page 496](#), whether the visualization size that is defined here is to be adjusted to the image size of the visualization client automatically.

Network Variable Properties

Network Variable Properties, Sender End

If the network functionality is supported by the device, the network properties of a global variable list (GVL) can be displayed and edited in the Properties dialog.

Defining network properties for a GVL means making the variables declared in the GVL available as network variables. For that purpose, a GVL has to be provided by the "sender" of the network variables. The receiver as counterpart has to define a corresponding global network variable list containing the same variable declarations. See [information on network variables, page 71](#).

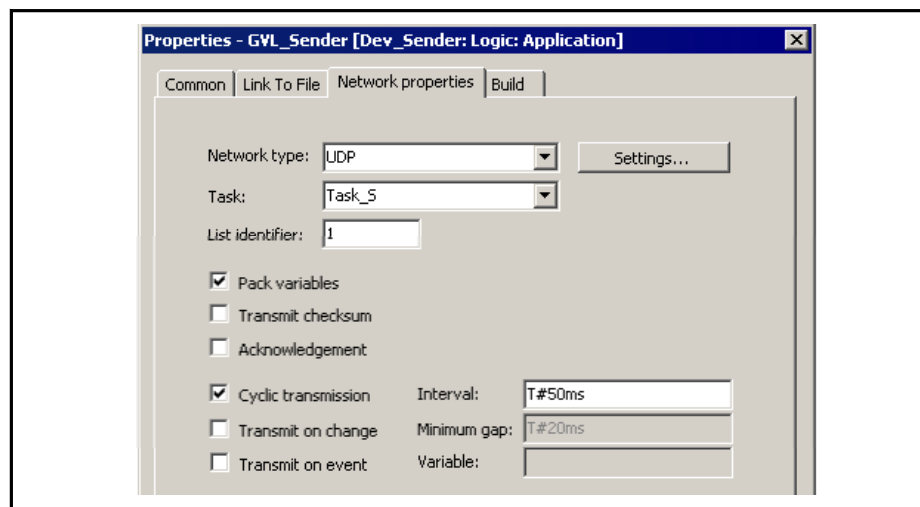


Fig.3-219: Setting GVL network properties, sender

- **Network type:** Select the desired type from the target system-dependent list. For example "UDP" for a UDP system.
- **Task:** From the selection list, choose the task of the current application that controls the variables to be sent. The variables are always sent at the end of one task cycle.
- **Variable list code:** Number of the list that is to be sent first (default=1). More lists are numbered consecutively.
- **Settings:** Protocol-specific settings; the possible entries depend on the relevant network library.

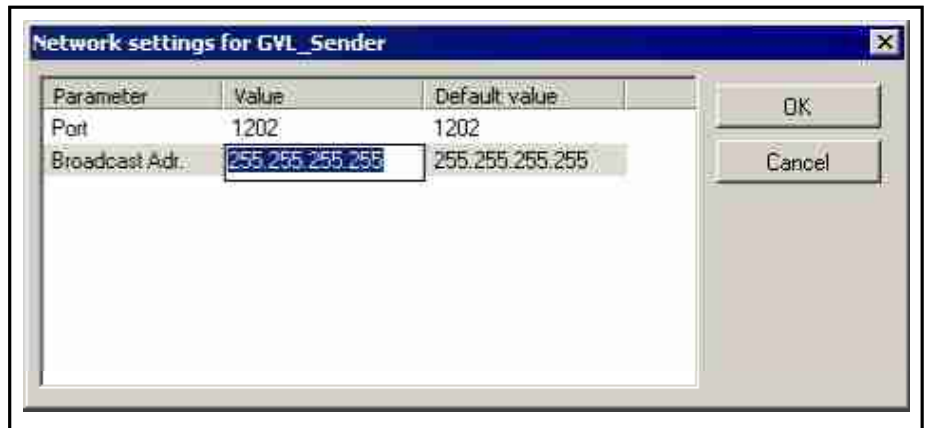


Fig.3-220: Setting GVL network properties, settings
For "UDP" networks, the following parameters have to be set:

- **Port:** Number of the port that is used for the data exchange with other network participants; the **preset value** is "1202"; the current value can be changed in the Value field at any time (select the field and press <space> in order to access the entry mode); ensure that the other nodes in the network define the same port! If more than one UDP connection is defined in the project, the port number is automatically adjusted in all configurations to the value set here.
- **Broadcast addr.:** The **preset value** is "255 . 255 . 255 . 255" means that data is exchanged with all network participants. The current value can be changed in the "Value" field (select the field and press <space> in order to access the entry mode) where to enter the address or the address range of a subnetwork (e.g. "197 . 200 . 100 . 255" in case of communication with all nodes having IP addresses in the range from "197 . 200 . 100 . x").



Note for Win32 systems that the broadcast addresses have to comply with the subnet mask of the TCP/IP configuration of the computer!

The following options can be enabled or disabled for configuring the transfer behavior of the variables:

- **Zip variables:** For the transfer, the variables are bundled in packages (telegrams) the size of which depends on the network type. If the option is disabled, one package per variable is generated.
- **Transfer checksum:** A checksum is attached to every variable package. The checksum is verified by the receiver in order to ensure that the variable definitions of sender and receiver comply with each other. A package with incorrect checksum is not accepted.
- **Cyclic transfer:** The variables are sent within the specified interval. (time specification e.g. "T#70ms").
- **Confirmation:** A confirmation message is sent for every data package received. If the sender has not received a confirmation before resending, an error is written into the diagnostic structure.
- **Transfer in case of change:** The variables are only sent if their values have changed. The **minimum distance** can be used to specify how much time has to pass at least between two transfers.
- **Event-controlled transfer:** The variables are sent as soon as the specified variable becomes TRUE.

Menu Items



Note that the network variables are automatically sent upon each system start. That means that the current variable values are also transferred if this were not the case at that time due to the configured transfer triggers (change, result).

Network Variable Properties, Receiver End

If the network functionality is supported by the device, the current network settings for a GNVL ([global network variable list, page 53](#)) can be displayed and changed in the Properties dialog. These are the settings that have been made upon insertion of the GNVL in the 'Add object' dialog. (See [more information on network variables, page 71](#)).

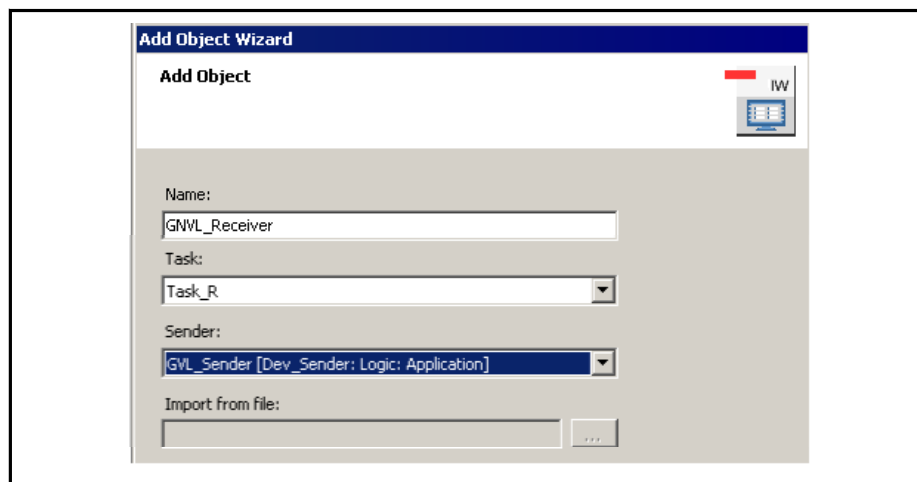


Fig.3-221: Properties dialog, network settings, receiver

- **Sender:** Name of the global variable list of the sender device that is referenced in the present GNVL. The square brackets contain the name of the device and the application or "import from file".
- **Task:** Name of the task of the current device controlling the network variables.
- **Import from file:** Enables the transfer of data as XML list e.g. from an IndraLogic-1.x system.

See also [file properties, page 245](#).

3.11 FBD/LD/IL - Commands

3.11.1 FBD/LD/IL - Commands, Overview

The commands described below are used to work in the [FBD/LD/IL editor, page 339](#).

This editor is the common editor for the three programming languages [function diagram \(FBD\), page 339](#), [ladder diagram \(LD\), page 340](#), and [instruction list \(IL\), page 340](#).

The menu item appears in the main menu when the FBD/LD/IL editor is active.

The menu item commands are also available in the context menu, depending on position.

If required, the menu structure can be reconfigured via the **IndraWorks ▶ Tools ▶ Customize ▶ Commands** dialog.



The display of the elements in FBD/LD/IL networks is configured in the "Options, FBD, LD and IL", page 207,. To do this, in the main menu click on **Tools ▶ Options ▶ IndraLogic 2G ▶ FBD, LD and IL**.



Note that in contrast to IndraLogic 1.x, networks are also used in the IL editor as structural units.

The commands:

- [Add network, page 250](#)
- [Add network below, page 250](#)
- [Commenting on/off, page 250,](#)
- [Add assignment, page 250](#)
- [Add FB call, page 251](#)
- [Add empty block, page 254](#)
- [Add jump, page 255](#)
- [Add jump label, page 256](#)
- [Add return, page 256](#)
- [Add function block input, page 257](#)
- [Only LD: Add coil, page 257](#)
- [Only LD: Add set coil, page 257](#)
- [Only LD: Add set coil, page 257](#)
- [Only LD: Add contact, page 257](#)
- [Only LD: Add negated contact, page 259](#)
- [Only LD: Add contact \(right\), page 259](#)
- [Only LD: Add parallel contact \(below\), page 259](#)
- [Only LD: Add parallel negated contact \(below\), page 259](#)
- [Only LD: Add parallel contact above, page 259](#)
- [Only LD: Add contacts: Add below, page 260](#)
- [Only LD: Add contacts: Add right \(after\), page 260](#)
- [Only LD: Add contacts: Add above page 261](#)
- [Only IL: Add IL line after, page 261](#)
- [Only IL: Delete IL line, page 261](#)
- [Negation, page 261,](#)
- [Edge detection, page 262](#)
- [Set/Reset, page 263](#)
- [Determine further connection, page 264](#)
- [Insert line branching](#)
- [Insert line branching above](#)
- [Insert line branching below](#)
- [Update parameters, page 265](#)
- [Remove unused FB call parameters, page 266](#)
- [View: Show as function block diagram, page 266](#)
- [View: Show as ladder diagram, page 267](#)

Menu Items

3.11.2 Add Network • [View: Show as instruction list, page 267](#)

Icon:

Default shortcut: <Ctrl>+<Shift>+

Menu: **FBD/LD/IL ▶ Adding a network**

Use this command to add a [network, page 356](#), **above** the highlighted network in the editor.

If the cursor is currently positioned in the editor window but not in a network, the new network is attached at the end of the existing network list. The network numbering is automatically updated.

Note that in contrast to IndraLogic 1.x, networks are also used in the IL editor [networks, page 346](#), as structural units.

3.11.3 Add Network Below

Icon:

Default shortcut: <Ctrl>+<Shift>+<A>

Menu: **FBD/LD/IL ▶ Insert Network below**

Use this command to add a [network, page 356](#), in the FBD/LD/IL editor. If the cursor is positioned in an existing network, the new network is added **below** the existing network.

If the cursor is currently positioned in the editor window but not in a network, the new network is attached at the end of the existing network list. The network numbering is automatically updated.

Note that in contrast to IndraLogic 1.x, networks are also used in the IL editor [networks, page 346](#), as structural units.

3.11.4 Commenting On/Off

Icon:

Default shortcut: <Ctrl>+<Shift>+<C>

Menu: **FBD/LD/IL ▶ Commenting on/off**

Use this command in the FBD/LD/IL editor to "comment out" a network or to return it to normal status again. The "Commenting on/off" command affects the network in which the cursor is located.

A network that is "commented out" is displayed and treated as a comment, i.e. the network is not processed.

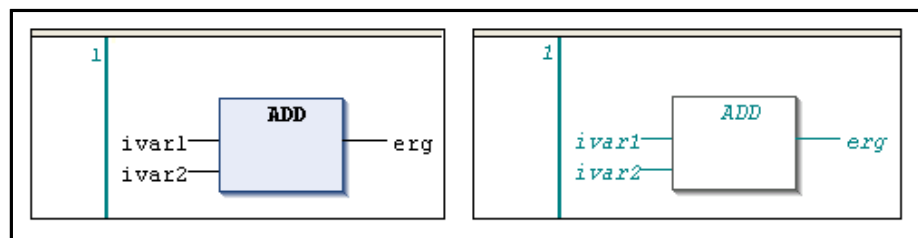


Fig.3-222: Network in normal status and in comment status

3.11.5 Add Assignment

Icon:

Default shortcut: <Ctrl>+<W>

Menu: **FBD/LD/IL ▶ Insert Assignment**

Menu Items

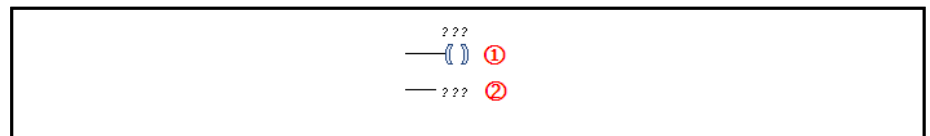
Use this command to insert an assignment in the FBD or LD editor. It is not available in the IL view.



In the LD view, an assignment is inserted as a coil using the command "Add coil", page 257.

Depending on the current [cursor position, page 351](#), the assignment is inserted directly before a selected input (cursor position 2), directly after a selected output (cursor position 4) or, if the entire network or subnetwork is highlighted, at the end of the respective network (cursor position 6 or 11).

In the FBD editor an assignment is inserted as a line followed by three question marks; in the LD editor it is inserted as a [coil, page 361](#) and three question marks.



- ① Assignment in LD
- ② Assignment in FBD

Fig.3-223: Assignments in the LD and FBD editors

To define an assignment, select the placeholder text "???" and replace it with the name of the variable that is to be assigned the signal (value) coming from the left. The input assistance can be accessed with the button.

In the IL editor, an assignment is inserted as an [ST operator, page 341](#); see the following example.

```

1 | PROGRAM ILInsertAssignment
2 | VAR
3 |     iVar: INT;
4 |     bResult: BOOL;
5 | END_VAR

1 | LD          iVar
  | ADD        1,
  |           4
  | GT         12
  | ST         bResult
    
```

Fig.3-224: Assignment as ST operator in the IL editor

3.11.6 Add FB Call

Icon:

Default shortcut: <Ctrl>+

Menu: **FBD/LD/IL** ► **Call FB call**

Use this command to add a function block element into an FBD, LD or IL network. This applies to calling an operator function block, a program, a function block, a function or an interface.

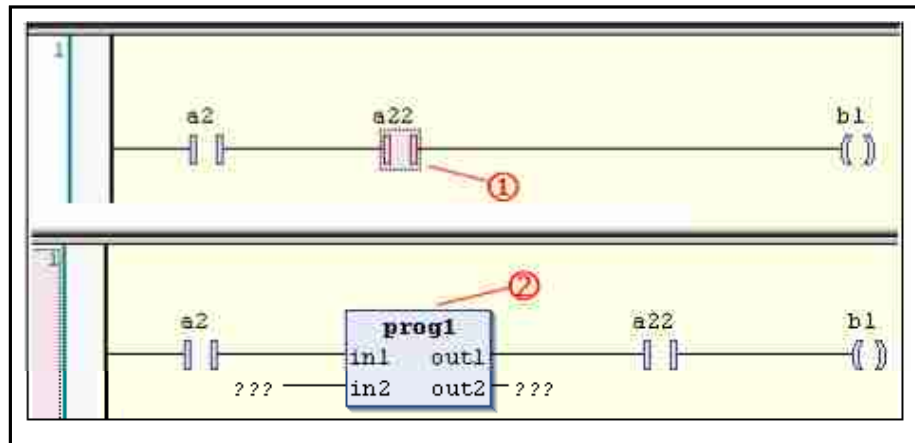
The corresponding instructions are added in the IL editor.

After selecting the "Add FB call" command, the [Input assistance, page 98](#) dialog opens, which lists the available function block categories. When selecting a function block and confirming it with "OK", a corresponding box with inputs and outputs or the corresponding IL instructions is inserted at the current cursor position in the network.

Menu Items

Alternatively, standard blocks can be dragged out of the [Tools, page 355](#), tab directly into the editor window. See the following information on editor-specific details:

- FBD/LD • "Program" or "function block" type function blocks are always added in a series; that means that the processing line is connected with the top-most input and the topmost output for the function block inserted.



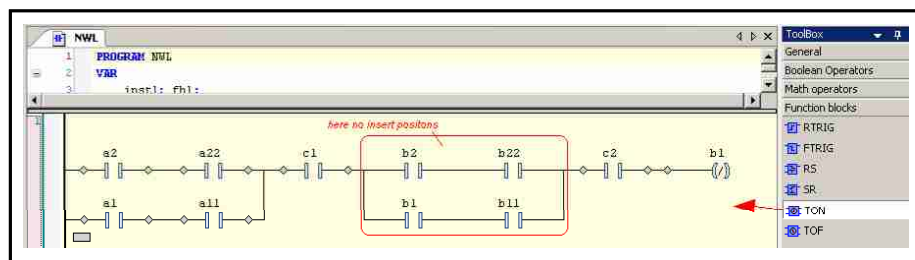
- ① Select contact
- ② prog1 program function block added

Fig.3-225: Adding a function block in LD

- The text in the inserted function block element indicates the **function block type** (e.g. "F_TRIG") and [can be edited, page 351](#).

By replacing this text with the type name of another valid function block, **replace** this function block call with a different one. A replacement can also be made by highlighting the existing function block element and inserted a different one in this position. Please note that the inputs and outputs **already assigned** for a function block remain in their previous order from top to bottom, unless the greatest number of inputs for the new function block is lower. In this case, the inputs and outputs that can no longer be linked are deleted, starting from the bottom.

- If it is available for the relevant function block and if the [Show function block symbol, page 207](#), option is selected, a symbol is displayed within the function block element.
- Within the parallel connections in an LD network, there is no insertion position for a function block from the "ToolBox" tab, since a function block call requires a direct connection to the electrical power supply (vertical lines at left and right in the LD network list).



- ① Insertion not possible

Fig.3-226: Insertion positions for FB call in an LD network

- **Function block calls with EN/ENO:** There is a special command for the insertion of a FB call with EN input and ENO output: [Add FB call with EN/ENO, page 254](#).

Menu Items



Consider the following when structuring the program: At the inputs of an EN/ENO FB, no FB call can be inserted! If the output of a function block is to be used as input of an EN/ENO function block, it has to be written to a variable in advance and the latter has then to be used as input of the EN/ENO function block.

- **VAR_IN_OUT parameters** of an inserted POU FB call are marked with an arrow pointing in both directions.
- **For function blocks**, the text field above the box in which the local instance variable has to be entered can also be edited. If a function block instance element of this type is replaced by inserting a different function block type, you have to enter the instance definition again.
- The **formal names of the inputs and outputs** are displayed in the boxes of function calls and function block calls. The output of a function (return value) is, however, shown without name.
- If the **interface of a function block** has changed (e.g. modified number of outputs), the function block parameters can be updated with the "Update parameters" command; see ["Update parameters, page 265"](#).



The function block parameters are not updated automatically as with IndraLogic 1.x!

- Insertion positions: A new FB call is inserted at the current cursor position as follows:
 - If an input is selected ([cursor position 2, page 351](#)), the block box is inserted before it and connected into the existing line branch with its first input and output.
 - If an output is selected ([cursor position 4, page 351](#)), the block box is inserted after it and connected into the existing line branch with its first input and output.
 - If a block box is selected ([cursor position 3, page 351](#)), this function block is replaced by the new one. As far as possible, the assignments of the inputs and outputs remain in their order from top to bottom. If the old function block had more inputs or outputs than the new one, those that are not linked are deleted (starting at the bottom).
 - If a jump or return element is selected ([cursor position 3, page 351](#)), the block box is inserted before it and connected into the existing line branch with its first input and output.
 - If an entire network or subnetwork is selected, ([cursor positions 6 and 11, page 351](#)), the function block box is inserted after the last element and connected with its first input. The output has no connection.
- All function block inputs that do not automatically receive assignments during insertion are given "???" as text. Select this text and replace it with the name of variable or with a constant.
- If you insert a function block at the end of a network, its output is not linked.

IL editor

In the IL editor, "Add FB call" can be used to insert a function block. After selecting the "Add FB call" command, the [Input assistance, page 98](#) dialog opens, which lists the available function block categories. If the "Insert with arguments" option is selected in the "Input Assistance" dialog, a [CAL instruc-](#)

Menu Items

tion, page 341, is inserted for example along with the corresponding input and output parameters; see the following example.

Example:

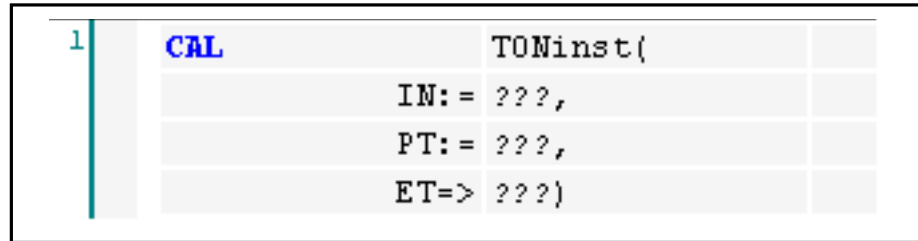


Fig.3-227: FB call in IL

The "TON" function block was selected with input assistance. The input parameters are automatically specified in the following lines and can be defined. In this example, the user replaced the "???" in the CAL line with "TONinst" (local instance of TON).

FB calls with EN/ENO cannot be inserted in the IL editor.

3.11.7 Add FB Call with EN/ENO

Symbol:

Shortcut: <Ctrl>+<Shift>+<E>

Menu: FBD/LD/IL ► Insert FB call with EN/ENO

This command is only available in the FBD editor. It can be used in order to add, page 251 a FB call with an EN input and an ENO output to a network.

Processing EN/ENO function blocks:

If at the time of the FB call EN has the value FALSE (0), the operations defined in the function block are not executed. Otherwise, i.e. if EN has the value TRUE (1), these operations are carried out. The ENO output acts as repeater of the EN input.

3.11.8 Add Empty Block

Symbol:

Menu: FBD/LD/IL ► Add empty block

This command is used to add an empty block element into an FBD, LD or IL network.

See Add FB call, page 251, for detailed information on adding FBs in the FBD/LD/IL editor. In contrast to the "Add FB call" command, the input assistance is not opened automatically, but the instance filed above the FB element.

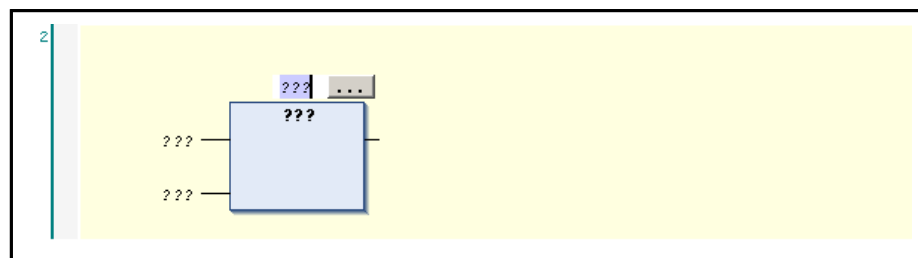


Fig.3-228: Empty function block

Decide then which function type is necessary:

- If there is to be a **function block**, enter the desired instance variable name and close the field by means of "Enter". The input assistance can

Menu Items

be used to select the name of an already existing instance variable. After entry of an already declared instance variable, the block is shown accordingly. If an instance name has been entered that is still unknown, you have to specify the name of the desired function block as well. After closing the input field, the focus is for that purpose automatically set into the input field within the box. In this case, the input assistance only provides function blocks.

- If the block is to represent an **operator**, a **program**, a **function** or an **interface**, press the <down arrow key> when the mouse cursor is located in the instance input field. The input focus changes to the function block type field (within the function block element) where the desired operator, program, function or interface name can be entered directly or via the input assistance. After completing this entry, the block is displayed in the network accordingly.

The corresponding instructions are added in the IL editor.

3.11.9 Add Jump

Icon: →

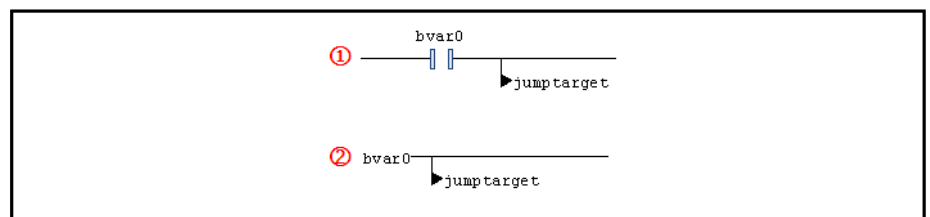
Default shortcut: <Ctrl>+<J>

Menu: **FBD/LD/IL ▶ Add jump**

Use this command to add a jump. The jump target is another network that is indicated here by its [label, page 357](#), (jump label).

In the FBD and LD editors the jump element is inserted based on the current cursor position, either directly before a selected input ([cursor position 2, page 351](#)), directly after a selected output ([cursor position 4, page 351](#),) or if the entire network or subnetwork is highlighted, at the end ([cursor position 6 or 11, page 351](#)).

A new jump element contains the text "???" that can be selected via click and replaced with the name of label for the network that is the jump target; see the following figure.



- ① Jump label in LD
- ② Jump label in FBD

Fig.3-229: Jump with jump label

In the IL editor, a jump is programmed with the [JMP operator, page 341](#),.

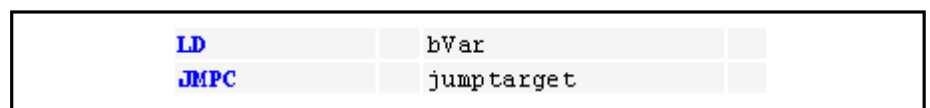
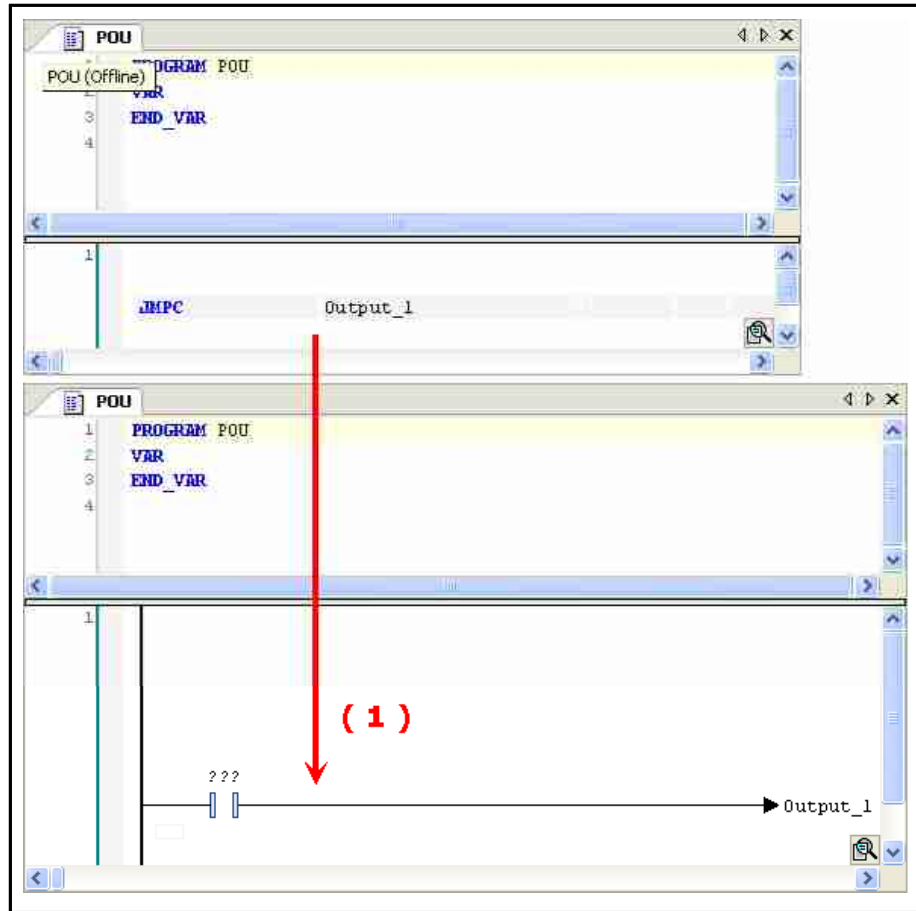


Fig.3-230: Jump with JMP operator

If a JMP operator that was previously inserted within the IL editor without a preceding LD is later converted in the LD editor, a dummy operator, "???", is inserted.

Menu Items



(1) Conversion from IL to LD
Fig.3-231: Example: Conversion of a JMP operator

3.11.10 Add Jump Label

Icon:

Menu: **FBD/LD/IL ▶ Add jump label**

Use this command to add a [jump label](#), [page 357](#) to the currently selected network. The jump label can be used as the target for a [jump](#), [page 357](#).

When calling the "Add jump label" command, by default the text "Label:" is entered in the associated text field which can be changed as desired.

3.11.11 Add Return

Icon:


Menu: **FBD/LD/IL ▶ Add return**

Use this command to insert a RETURN instruction at the current cursor position.

FBD/LD Based on the current cursor position the return element is inserted directly before a selected input ([cursor-position 2](#), [page 351](#)), directly after a selected output ([cursor position 4](#), [page 351](#)), directly before a branch ([cursor position 5](#), [page 351](#)), or at the end of a network or subnetwork ([cursor position 6 or 11](#), [page 351](#)).

IL In the IL editor an instruction is added with the "RETURN", [page 341](#), operator.

3.11.12 Add Block Input

Icon: 

Default shortcut: <Ctrl>+<Q>


Menu: **FBD/LD/IL ► Call FB call**

This command adds another input to an already added extendable block (AND, OR, ADD, MUL, SEL) in the FBD or LD editor. The command is not available in the IL editor.


The maximum number of inputs depends on the block type (ADD e.g. may have 2 or more inputs).

To insert the input at a certain position relative to the existing input, highlight the input above which the new one is to be inserted ([cursor position 1, page 351](#)).

To insert another input for a function block that is way at the bottom, highlight the function block body ([cursor position 3, page 351](#)).

The new input is initially assigned the text "???". Replace this text with the name of a variable or with a constant. The input assistance can be accessed with the  button.

3.11.13 Add Coil


Icon: 

Default shortcut: <Ctrl>+<W>

Menu: **FBD/LD/IL ► Add coil**

Use this command to insert a [coil, page 361](#) parallel to the existing coils.

If the [highlighted position, page 351](#) is a connection between [contacts, page 360](#) and coils, the new coil is inserted at the end. If the highlighted position is a coil, the new coil is inserted parallelly and directly above it.

By default, the coil is assigned the text "???". To change this text, click on it to edit it or use the [Input assistant, page 98](#). The input assistance can be accessed with the  button.

3.11.14 Add Set Coil

Icon: Menu: **FBD/LD/IL ► Add set coil**

Use this command to insert a [set coil, page 362](#).

The "[Set/Reset, page 263](#)", command can convert a coil into a "set" coil.


3.11.15 Add Reset Coil

Icon: Menu: **FBD/LD/IL ► Add reset coil**

Use this command to insert a [reset coil, page 362](#).

The "[Set/Reset, page 263](#)", command can convert a coil into a "reset" coil.

3.11.16 Add Contact

Icon: 

Default shortcut: <Ctrl>+<K>

Menu: **FBD/LD/IL ► Add contact**

Menu Items

Use this command to add a contact element to a ladder diagram network.



The "Add contact" command is only available in the LD editor. When switching to the FBD or IL view, a contact is converted accordingly.

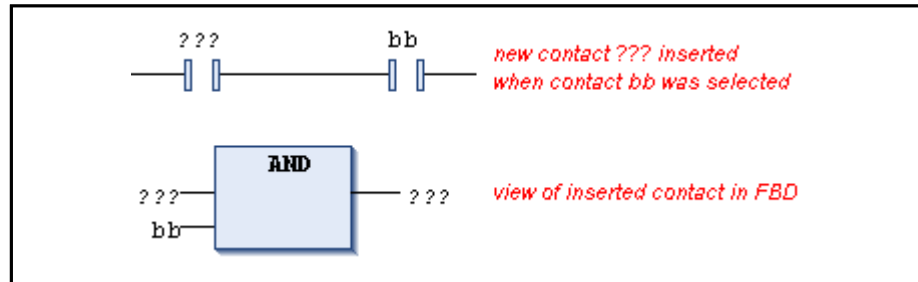


Fig.3-232: Switching from LD to FBD

The new contact is inserted at the current cursor position in the sequence **before** the existing contact or a function block box. If the cursor position is in the branch of an existing parallel connection, the new contact is inserted into it.

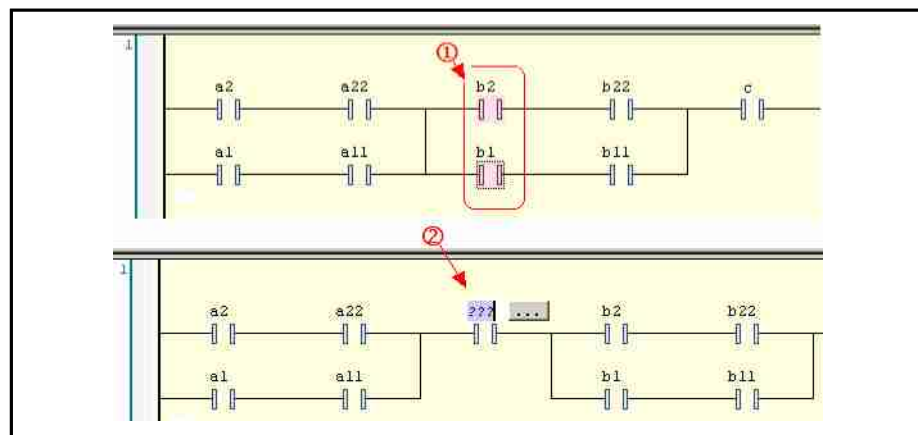


Alternatively, insert a contact element from the [Tools, page 355](#), tab. However, if it is to be inserted not within, but before, after or between existing parallel connections, only the "Add contact" command can be used.

To do this, one of the existing contact elements in each branch of the parallel connection has to be highlighted (multiple selection with <Ctrl> key) and the "Add contact" command used; see the following example "Add contact with parallel connection".

Example:

Adding a contact with parallel connection




- ① Highlight b1 and b2 while pressing and holding the <Ctrl> key.
- ② Use the "Add contact" command.


Fig.3-233: Adding a contact with parallel connection

Note also the following commands:

- [Add contact \(right\), page 259](#)
- [Add parallel contact \(above\), page 259](#)
- [Add parallel contact \(below\), page 259.](#)

By default, a new contact element contains the text "???". To modify this text, highlight the three question marks and enter the desired variable name or an address (depending on the current settings in the "FBD, LD and IL options", [page 207](#)) or a constant. In addition, the input assistance can be accessed with the  button.

3.11.17 Add Contact (Right)


Icon: 

Default shortcut: <Ctrl>+<R>

Menu: **FBD/LD/IL ► Add contact (right)**

Use this command to add a contact element into a ladder diagram network. The same features apply as for the "Add contact" command, except in this case the new element is not inserted to the left, but instead, to the right of the current cursor position; see [Add contact, page 257](#).

3.11.18 Add Parallel Contact (Below)

Icon: 

Default shortcut: <Ctrl>+<D>

Menu: **FBD/LD/IL ► Add parallel contact (below)**

Use this command to insert a parallel contact at the current cursor position in an LD network. The same features apply as for the "Add parallel contact (above)", except in this case the new contact appears below the selected position; see [Add parallel contact \(above\), page 259](#).

3.11.19 Add Negated Contact

Icon: 

Menu: **FBD/LD/IL ► Add negated contact**

Use this command to insert a negated [contact, page 360](#).



An existing contact can also be converted into a negated contact with the [Negation, page 261](#), command.

3.11.20 Add Parallel Negated Contact (Below)

Icon: 


Menu: **FBD/LD/IL ► Add parallel negated contact (below)**

Use this command to add a parallel negated [contact, page 360](#), below the current cursor position.



An existing contact can also be converted into a negated contact with the [Negation, page 261](#), command.

3.11.21 Add Parallel Contact (Above)

Icon: 

Default shortcut: <Ctrl>+<E>

Menu: **FBD/LD/IL ► Add parallel contact (above)**


Use this command to insert a parallel contact (parallel connection) at the current cursor position in an LD network. Several existing elements can also be highlighted in order to insert a new contact parallel to them and above the current cursor position.

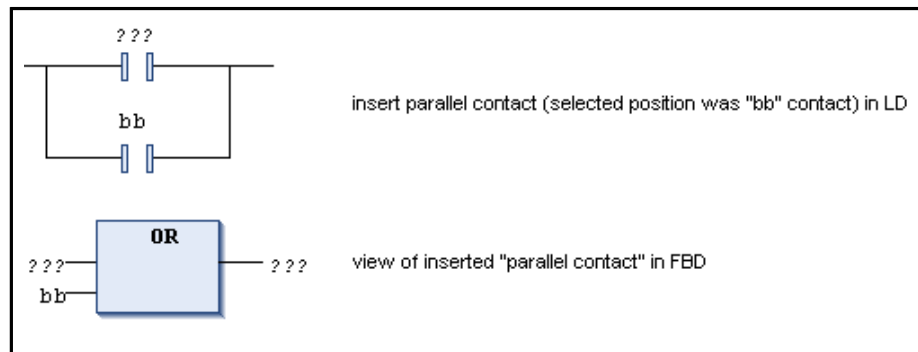
Menu Items



The "Add parallel contact (above)" command is only available in the LD editor. However, when switching to the IL or FBD editor, the contact is converted accordingly.

By default, a new contact element contains the text "???". To modify this text, highlight the three question marks and enter the desired variable name or an address (depending on the current settings in the "FBD, LD and IL options")

or a constant. In addition, the input assistance can be accessed with the  button.



- ① Inserting a parallel contact in the LD view (the highlighted position was contact "bb")
- ② Display of the inserted parallel contact in the FBD view

Fig.3-234: Inserting a parallel contact

3.11.22 Add Contacts: Add Below

Default shortcut: <Ctrl>+<Shift>+<D>

Menu: **FBD/LD/IL** ► **Add contacts** ► **Add below**

Use this command to add contacts or a section of the network, which was previously placed on the clipboard using the "Copy" or "Cut" command.

The contacts are inserted below the area currently highlighted. This corresponds with the general "Insert" command.



The "Insert below" command is only available in the LD editor. However, when switching to the IL or FBD editor, the contact is converted accordingly.

To insert contacts below the selected area, highlight one or more contacts and execute the "Copy" or "Cut" command. Then activate the command.

Information about highlighting in networks can be found in the editors under [Working in the FBD and LD editors, page 339](#).

3.11.23 Add Contacts: Add Right (After)

Default shortcut: <Ctrl>+<Shift>+<R>

Menu: **FBD/LD/IL** ► **Add contacts** ► **Add right (after)**

Use this command to add contacts or a section of the network, which was previously placed on the clipboard using the "Copy" or "Cut" command.

The contacts are inserted to the right of the area currently highlighted.

This corresponds with the "Insert" command.



The "Add right (after)" command is only available in the LD editor. However, when switching to the IL or FBD editor, the contact is converted accordingly.

To insert contacts below the selected area, highlight one or more contacts and execute the "Copy" or "Cut" command. Then activate the command.

Information about highlighting in networks can be found in the editors under [Working in the FBD and LD editors, page 339](#).

3.11.24 Add Contacts: Add Above

Default shortcut: <Ctrl>+<Shift>+<E>

Menu: **FBD/LD/IL** ► **Add contacts** ► **Insert above**

Use this command to add contacts or a section of the network, which was previously placed on the clipboard using the "Copy" or "Cut" command.

The contacts are inserted above the area currently highlighted.

This corresponds with the "Insert" command.



The "Insert above" command is only available in the LD editor. However, when switching to the IL or FBD editor, the contact is converted accordingly.

To insert contacts above the selected area, highlight one or more contacts and execute the "Copy" or "Cut" command. Then activate the command.

Information about highlighting in networks can be found in the editors under [Working in the FBD and LD editors, page 339](#).

3.11.25 Add IL Line After

Icon: 

Menu: **FBD/LD/IL** ► **Add IL line after**

Use this command to insert another instruction line below the line in which the cursor is currently located.



The "Add IL line after" command is only available in the IL editor. When switching to the LD or FBD editor, the function of the line is converted accordingly.

3.11.26 Delete IL Line

Icon: 

Default shortcut: <Ctrl>+



Menu: **FBD/LD/IL** ► **Delete IL line**

Use this command to delete the instruction line in which the cursor is located.



The "Delete IL line" command is only available in the IL editor. When switching to the LD or FBD editor, the function of the line is converted accordingly, and in this case, deleted.

3.11.27 Negation

Icon:  (FBD),  (LD)

Default shortcut: <Ctrl>+<T>

Menu Items

Menu: **FBD/LD/IL ▶ Negation**

Use this command to toggle the currently selected input, output, jump or RETURN instruction between "negated" and "not negated".



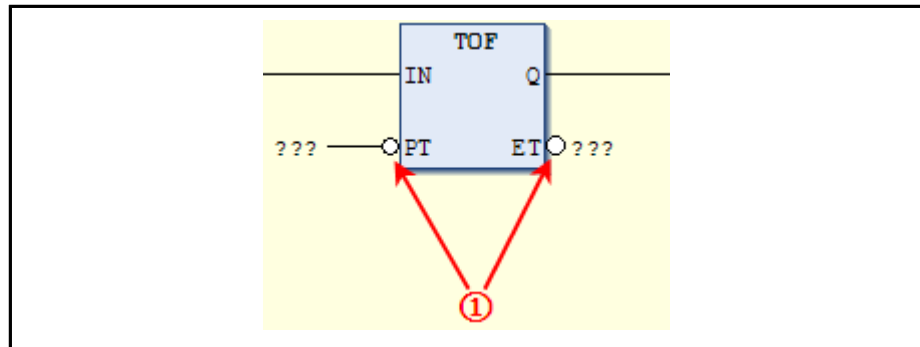
The "Negation" command is not available in the IL editor. In the IL editor, the corresponding modifiers have to be set. For information about the modifiers, see [Modifiers and operators in IL, page 339](#).



When switching back and forth among the FBD, LD and IL views, the negations of some constructs might be reset, since a unique conversion is not possible.

Types of display for negation

- For the function block, jump and RETURN instruction elements, a negation is shown with a small circle icon at the respective input or output connection; see the following figure of a "negated block".



① Negation

Fig.3-235: Negated block

To negate a jump or a RETURN instruction, highlight the most recent, previous output ([cursor position 4, page 351](#)).

- In the LD editor, a negated contact is indicated by a slash in the contact icon; see the following figure of a "negated contact".

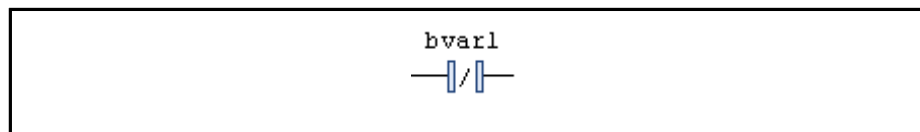


Fig.3-236: Negated contact



The "Tools", [page 355](#), tab also provides negated contact elements in the "ladder diagram elements" category. Add elements from the "ToolBox" tab by using the mouse to drag them to the corresponding position.

- In LD, a negated coil is indicated with a slash in the coil icon; see the following figure of a "negated coil".



Fig.3-237: Negated coil

To negate an input or output, highlight it at [cursor position 2 or 4, page 351](#).

3.11.28 Edge Detection

Icon: (FBD), (LD)

Menu Items

Default shortcut: <Ctrl>+<Shift>+<G>

Menu: **FBD/LD/IL ▶ Edge Detection**

Use this command to insert a detector for rising or falling edge at a Boolean input in the FBD or LD editor. This is comparable to using the R_TRIG function block (rising edge: FALSE → TRUE) and the F_TRIG function block (falling edge TRUE → FALSE).



The "Edge detection" command is not available in the IL editor. FBD or LD networks that contain edge detection are accepted without modification in the IL view.

Execute the "Edge detection" command several times until the desired icon appears. The following are in the sequence:

- for rising edges
- for falling edges
- no icon

Example:

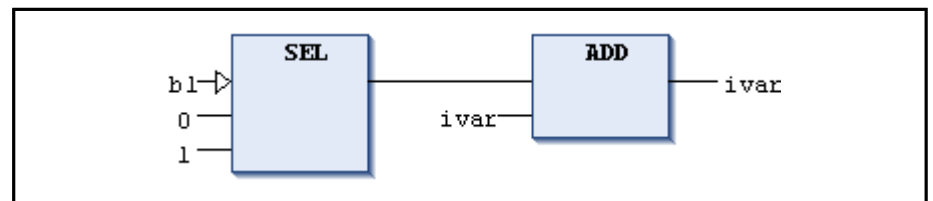


Fig.3-238: Detector for rising edges at the SEL operator

In this example, the input at "b1" was highlighted and the "Edge detection" command executed. The detector for rising edges was inserted, i.e. the output of "SEL" delivers "1" if "b1" changes from FALSE to TRUE.

The command is not available in the IL editor. FBD or LD networks that contain edge detection are accepted without modification in the IL view.

3.11.29 Set/Reset

Icon:

Default shortcut: <Ctrl>+<Shift>+<S>

Menu: **FBD/LD/IL ▶ Set/Reset**

Use this command to define Boolean outputs as "set" or "reset" outputs and coils to "set" or "reset" coils.



The "Set/Reset" command is not available in the IL editor. In the IL editor, the corresponding modifiers have to be set. For information about the modifiers, see ["Modifiers and operators in IL," page 341.](#)

If the "Set/Reset" command should be executed at the same output several times in a row, "Set", "Reset" and normal state is set in sequence.



Further information about this command is located in [Set/Reset, page 355.](#)

Menu Items

3.11.30 Determine Further Connection

Icon:

Default shortcut: <Ctrl>+<Shift>+<W>

Menu: **FBD/LD/IL ▶ Determine further connection**

Use this command for function block elements with several outputs to determine which of these outputs are to be connected with the continuing processing line in the network.



The "Determine further connection" command is not available in the IL editor.

Highlight the output that is to be connected further and activate the command in order to specify the further connection.

Note that the output assignments are displaced when the further connection is redefined; see the following figure.

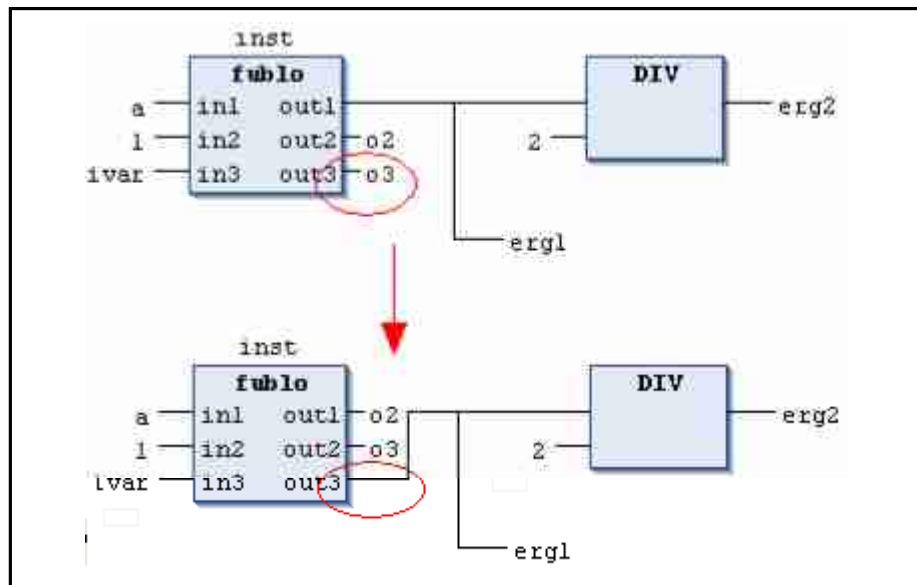


Fig.3-239: Further connection is positioned at out3.

3.11.31 Insert Line Branching

Icon:

Menu: **FBD/LD/IL ▶ Insert line branching**

This command branches the current processing line in a [network, page 343](#), in FBD.

The processing line is divided into two [subnetworks, page 264](#): Another processing line is inserted below the existing one. The two subnetworks are processed in online mode according to their position from top to bottom.

If a "line branching" is dragged from the toolbox or from any other position in the editor, the possible [insertion positions, page 343](#), are displayed by gray position labels.

A branch can be inserted at the input connectors of function blocks that are not located within a subnetwork, at output connectors (cursor position 4) of a block if it is not (not indirectly either) connected to the input of another function block in the subnetwork, at the connection between contacts and coils (cursor position 10) or at a contact (cursor position 8). A branch cannot be

Menu Items

inserted within an "ORed" contact group and within multiple assignment groups [cursor positions](#), [page 351](#)).

Every subnetwork gets an own "label", an upright rectangular symbol serving as possible cursor position (11) for selection of the subnetwork.

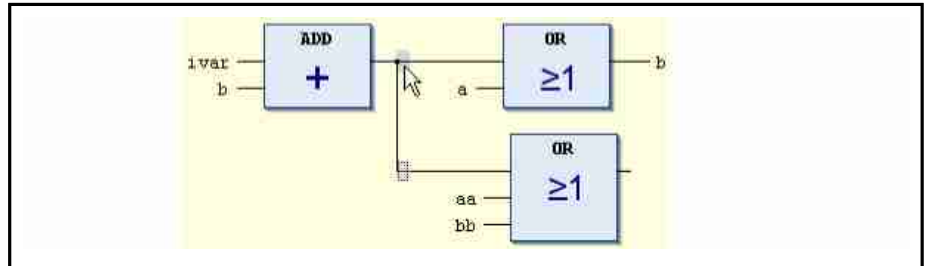


Fig.3-240: Subnetwork label in an FBD network

Every subnetwork can again be branched which allows for the generation of a widely ramified network and "subnetworks" within the main network.

For adding subnetworks to subnetworks see the commands:

[Insert line branching above](#), [page 265](#), and

[Insert line branching below](#), [page 265](#).

See [branching in FBD/LD/IL](#), [page 358](#), for more information on branching and subnetworks.

Connections between different branching arms or subnetworks can be removed or established by means of the Separate connections and Connect connections commands.

The command is not available in the IL editor. Networks with branching cannot be converted into IL.

3.11.32 Insert Line Branching Above

Icon:

Menu: **FBD/LD/IL** ► **Insert line branching above**

This command is used to insert a [subnetwork](#), [page 264](#), within the existing one, namely above the current cursor position.

3.11.33 Insert Line Branching Below

Icon:

Menu: **FBD/LD/IL** ► **Insert line branching below**

This command is used to insert a [subnetwork](#), [page 264](#), within the existing one, namely below the current cursor position.

3.11.34 Update Parameters

Default shortcut: <Ctrl>+<Shift>+<P>

Menu: **FBD/LD/IL** ► **Updating parameters**

Use this command to update the input and output parameters for previously inserted blocks if their interface has changed (if, for example, an output was inserted).

Menu Items

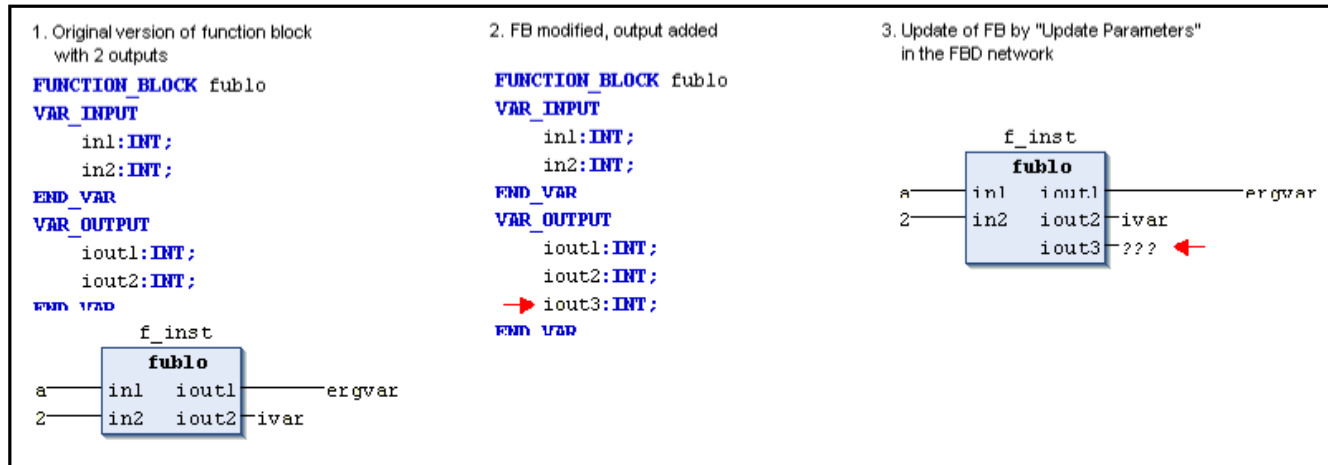


Fig.3-241: Updating parameters

The existing connections to the remaining inputs or outputs remain intact. If an input or output is added, it appears with the text "???" and can then be assigned.

3.11.35 Remove Unused FB Call Parameters

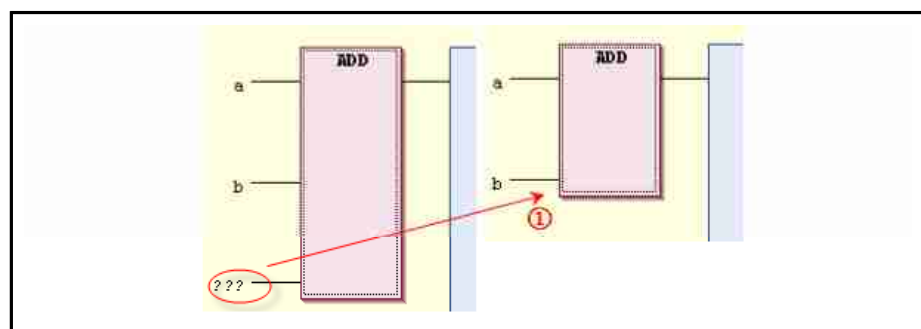
Icon:

Menu: **FBD/LD/IL** ▶ **Remove unused FB call parameters**

Use this command to remove all unused inputs or outputs for a highlighted function block (all inputs and outputs that contain the text "???"). The lowest amount of necessary inputs or outputs for a function block are retained; see the figure below.



The "Remove unused FB call parameters" command is not available in IL.



① Unused input removed

Fig.3-242: Command - Remove unused FB call parameters

3.11.36 View: Show as Function Block Diagram

Default shortcut: <Ctrl>+<Shift>+<F>

Menu: **FBD/LD/IL** ▶ **View** ▶ **Show as function block diagram**

Use this command to switch among the [FBD/LD/IL editors](#), page 339, in off-line and online mode in the FBD view.

The "Show as function block diagram" command allows an LD or IL network to be converted into an FBD network.



However, note that some special elements cannot be converted and remain provided only in the original editor view. In addition, not all constructs can be converted uniquely; see also "[Show as instruction list](#)", page 267.

To switch back to the LD view, use the command "[Show as ladder diagram](#)", page 267.

To switch back to the IL view, use the command "[Show as instruction list](#)", page 267.

⚠ CAUTION

Error-free conversion requires code with correct syntax. Otherwise, parts of the implementation might be lost.

3.11.37 View: Show as Ladder Diagram

Default shortcut: <Ctrl>+<Shift>+<L>

Menu: **FBD/LD/IL** ► **View** ► **Show as ladder diagram**

Use this command to switch among the [FBD/LD/IL editors](#), page 339 in offline and online mode in the LD view.

The "Show as ladder diagram" command allows an IL or FBD network to be converted into an LD network. However, note that some special elements cannot be converted and remain provided only in the original editor view. In addition, not all constructs can be converted uniquely; see also "[Show as instruction list](#)", page 267.

FBD elements that cannot be displayed as LD element (e.g. XOR) are transferred into the LD network as FBD blocks.

To switch back to the FBD view, use the command "[Show as function block diagram](#)", page 266.

To switch back to the IL view, use the command "[Show as instruction list](#)", page 267.

⚠ CAUTION

Error-free conversion requires code with correct syntax. Otherwise, parts of the implementation might be lost.

3.11.38 View: Show as Instruction List

Default shortcut: <Ctrl>+<Shift>+<I>

Menu: **FBD/LD/IL** ► **View** ► **Show as instruction list**

Use this command to switch among the [FBD/LD/IL editors](#), page 339 in the IL view.

The "Show as instruction list" command allows an LD or FBD network to be converted into an IL network.

Menu Items



Since there are elements that cannot be converted, the network affected continues to be displayed in the original editor view.

Example: Operators CALC, CALCN

If there are syntax errors, conversion is not possible either, and a corresponding error message is output. A few constructs cannot be converted uniquely. Thus, they are "normalized" when switching back and forth among IL and FBD/LD. This applies to negation and explicit/implicit assignments for function block inputs and outputs.

To switch back to the FBD view, use the command "[Show as function block diagram](#)", page 266.

To switch back to the LD view, use the command "[Show as ladder diagram](#)", page 267.

⚠ CAUTION

Error-free conversion requires code with correct syntax. Otherwise, parts of the implementation might be lost.

3.12 CFC Commands

3.12.1 CFC Commands, Overview

The commands of the [CFC Editor](#), page 309, (Continuous Function Chart) are described in this section.

By default, they are included in the **CFC menu** which is available when the CFC editor is active. Alternatively, they are provided in the context menu for the CFC elements in the working sheet or toolbox.

If required, the menu structure can be reconfigured via the **IndraWorks ▶ Tools ▶ Customize ▶ Commands** dialog.

The commands:

- [Edit Working Sheet...](#), page 269
- [Negation](#), page 270
- [EN/ENO](#), page 270
- **Set/Reset:**
 - [None](#), page 271
 - [R - Reset](#), page 272
 - [S - Set](#), page 272
- **Execution order:**
 - [At the Beginning](#), page 272
 - [At the End](#), page 273
 - [One Ahead](#), page 273
 - [One Back](#), page 273
- [Setting Execution Order](#), page 276
- [Arranging by Dataflow](#), page 273
- [Arranging Topologically](#), page 275
- [Connecting Selected Connections](#), page 277

- [Resetting Connections, page 277](#)
- [Removing Unconnected Connections, page 278](#)
- [Editing Parameters..., page 276](#)

3.12.2 Edit Working Sheet...

This command opens the **Edit Working Sheet** dialog, in which the size of the current CFC working sheet can be modified.

The size of the rectangular working area that includes all existing CFC elements is defined by entering height and width, where the origin (X: 0, Y: 0) is located at the upper left corner of the editor window. Height and width are given in integer **grid units**. The size of a grid unit cannot be modified by the user. The height value (Y) increases from top to bottom and the width value (X) from left to right.

The maximum size is 2048 grid units for both the height and the width.

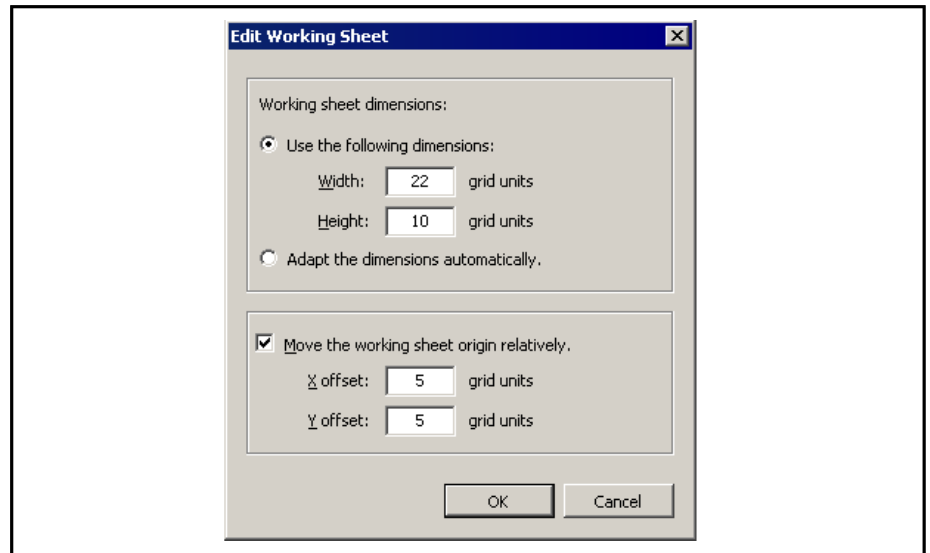


Fig.3-243: Dialog - Edit Working Sheet...

Working sheet dimensions:

Use the following dimensions:

If this option is selected, the working sheet is sized according to the dimensions given for height and width.

- **Width:** Contains the current width in grid units. The width can be changed; however, it is not possible to enter a value that is less than required for the currently existing elements. The width value (X) increases from left to right.
- **Height:** Contains the current height in grid units. The height can be changed; however, it is not possible to enter a value that is less than required for the currently existing elements. The height value (Y) increases from top to bottom.

Adapt the dimensions automatically.:

This option is selected by default. The height of the working sheet is defined by the element at the bottom; the width is defined by the element positioned furthest to the right. The origin (X=0, Y=0) is identical to the upper left corner.

Menu Items

Move the working sheet origin relatively:

After this option is selected, the working area can be moved vertically or horizontally relative to the origin based on the offset values entered (in grid units).

Restrictions:

The offsets have to be integers and may not move the upper left corner of the working area out of the window.

When the 'Use the following dimensions' option in the upper dialog section is used, the movement must not exceed the values defined there for width and height.

If the 'Adapt dimensions automatically' option is selected, the movement may exceed the current values for height and width; these are automatically adjusted afterward.

- **X-offset:** By default, 0. Entering a positive number moves the working area to the right and might thus become wider. Entering a negative number moves the canvas to the left. Thus, it is only possible if there is enough space between the element positioned furthest to the left and the left window margin.
- **Y-offset:** By default, 0. Entering a positive number moves the working area down and might it thus enlarge. Entering a negative number moves the working area up. Thus, it is only possible if there is enough space between the element positioned furthest towards the top and the upper window margin.

If entering invalid dimension values, an error message is output listing the restriction explained above.

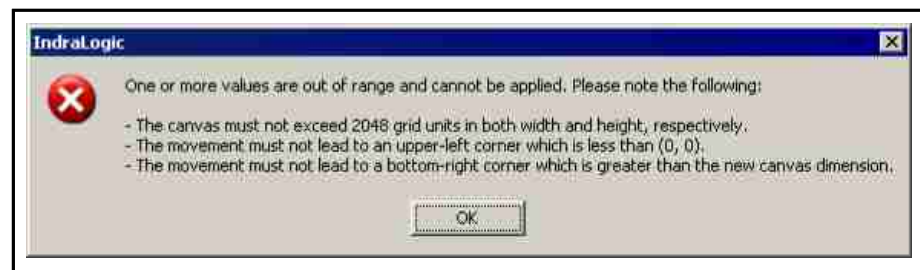


Fig.3-244: Error message in case of dimension values that are not permissible

3.12.3 Negation

Icon:

Menu: **CFC ▶ Negation**

This command is used to negate inputs, outputs, jumps or RETURN instructions.

The icon for negation is a small circle on the connection line.

To insert the negation, select the corresponding input or output pin for the respective element and execute the command.

A detailed description of the possible cursor positions is located on the help page [cursor positions in CFC. page 310](#).

A negation can be canceled by executing the command again.

3.12.4 EN/ENO

Icon:

Menu: **CFC ▶ EN/ENO**

Menu Items

This command is used to give the selected function block (cursor position 3, page 310) an additional Boolean "Enable" input EN and an additional Boolean output ENO (enable out).

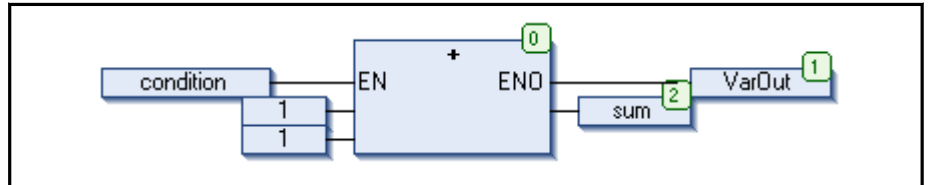


Fig.3-245: ADD function block with EN/ENO

In this example, ADD is only executed if the Boolean variable **condition** is TRUE.

VarOut is set to TRUE after ADD is executed.

If **condition** is set to FALSE again later, ADD is not executed again and **VarOut** becomes FALSE as well!

The following example shows how the ENO value can be used for other function blocks:

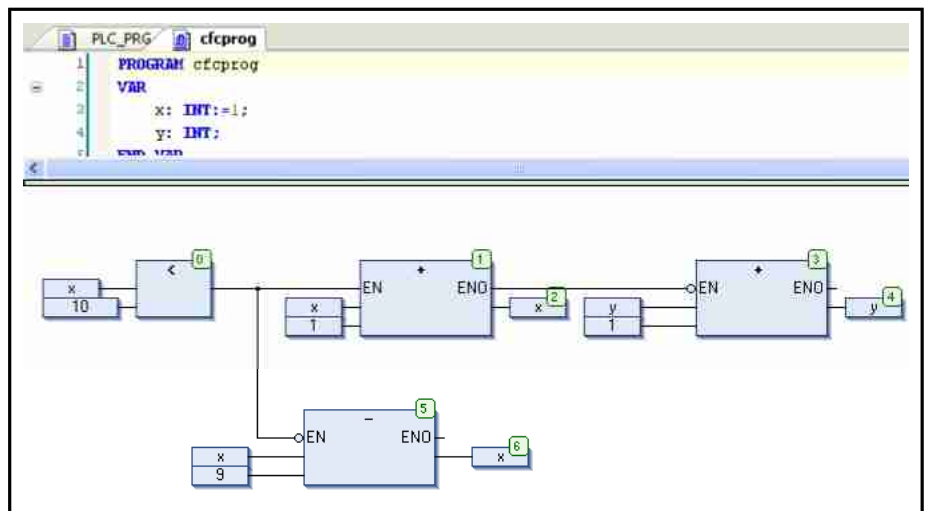


Fig.3-246: Example use of EN/ENO

For this example, initialized **x** with "1".

The numbers in the upper right corner of every function block indicate the execution order in the network.

As long as **x** is less than 10 (0), it increases by 1 (1).

As soon as **x** = 10, the output from LT(0) delivers the value FALSE and SUB (6) and ADD (4) are executed.

x is reset to "1" and **y** increases to 1.

LT (0) continues to be executed as long as **x** is less than 10.

In this way **y** counts the number of times **x** passes through the value range 1 through 10.

3.12.5 Set/Reset

None (Set/Reset)

Icon:

Menu: CFC ► Set/Reset ► None

Menu Items

This command is used to remove a [Set](#) or [Reset](#) from an output element. To do this, select the input pin for the output element and execute the command. The "Set" or "Reset" icon at the input will then disappear.

A detailed description of the possible cursor positions is located on the help page [cursor positions in CFC. page 310](#).

R - Reset

Icon:

Menu: **CFC ▶ Set/Reset ▶ R (Reset)**

This command is used to add a "Reset" to a Boolean output element. That means that when the input provides the value TRUE, the output is set to FALSE and retains this value.

To insert a reset, select the input pin for the output and execute the command.

A detailed description of the possible cursor positions for a selection is located in [cursor positions in CFC. page 310](#).

The reset output is designated by a small "R" icon on its input.

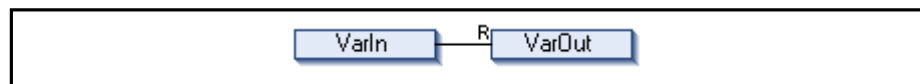


Fig.3-247: Example

In this example, **VarOut** is set to FALSE as soon as **VarIn** delivers the value TRUE. **VarOut** retains this value, even if **VarIn** becomes FALSE again.

Alternatively, either a [S \(Set\)](#) or a [None](#) command can be used to define the Set/Reset properties of an output.

S - Set

Icon:

Menu: **CFC ▶ Set/Reset ▶ S (Set)**

This command is used to add a "Set" to a Boolean output element. That means that when the input becomes TRUE, the output is also set to TRUE and retains this value.

To insert a reset, select the input pin for the output and execute the command.

A detailed description of the possible cursor positions for a selection is located in [cursor positions in CFC. page 310](#).

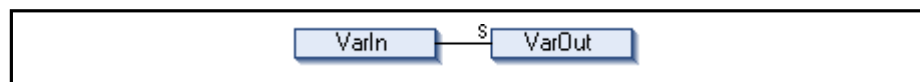


Fig.3-248: Example

In this example, **VarOut** is set to TRUE as soon as **VarIn** delivers the value TRUE. **VarOut** retains this value, even if **VarIn** becomes FALSE again.

Alternatively, either a [R - Reset](#) or a [None](#) command can be used to define the Set/Reset properties of an output.

3.12.6 Execution Order

At the Beginning

Icon:

Menu: **CFC ▶ Execution Order ▶ At the beginning**



Menu Items

This command is use to move all of the selected elements in the CFC editor to the beginning of the [execution order, page 315](#); the sequence among the selected elements is retained. Likewise, the order within the group of elements that were not selected is retained.

At the End

Icon:

Menu: **CFC ▶ Execution Order ▶ At the end**

This command is use to move all of the selected elements in the CFC editor to the end of the [execution order, page 315](#); the sequence among the selected elements is retained. Likewise, the order within the group of elements that were not selected is retained.

One Ahead

Icon:

Menu: **CFC ▶ Execution Order ▶ One ahead**

This command is used to move all of the selected elements in the CFC editor one place ahead in the execution order, except for the elements that are at the beginning of the [execution order, page 315](#).

One Back

Icon:

Menu: **CFC ▶ Execution Order ▶ One back**

This command is used to move all of the selected elements in the CFC editor one place back in the execution order, except for the elements that are at the beginning of the [execution order, page 315](#).

Arrange by Dataflow

Menu: **CFC ▶ Arrange by dataflow**

This command is used to arrange all of the currently selected elements in the CFC editor in the [execution order, page 315](#), (indicated by the element numbers in the upper right corner of each element).

This command is applied to all elements in the CFC editor.

The execution order is determined by the dataflow of the elements and not by their positions.

The following figure shows elements arranged topologically:

Menu Items

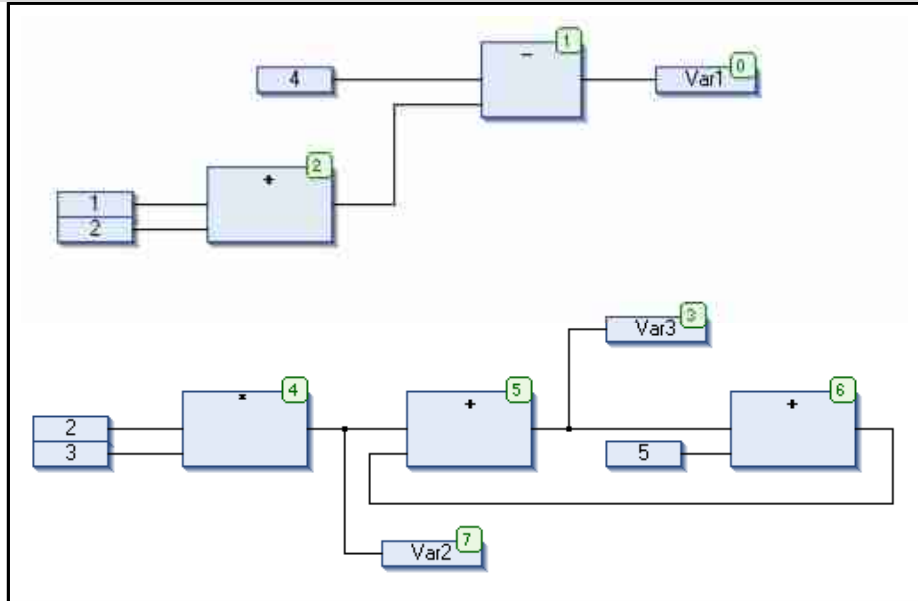


Fig.3-249: Example of topological execution order

After the command is executed, it determines the following order based on dataflow:

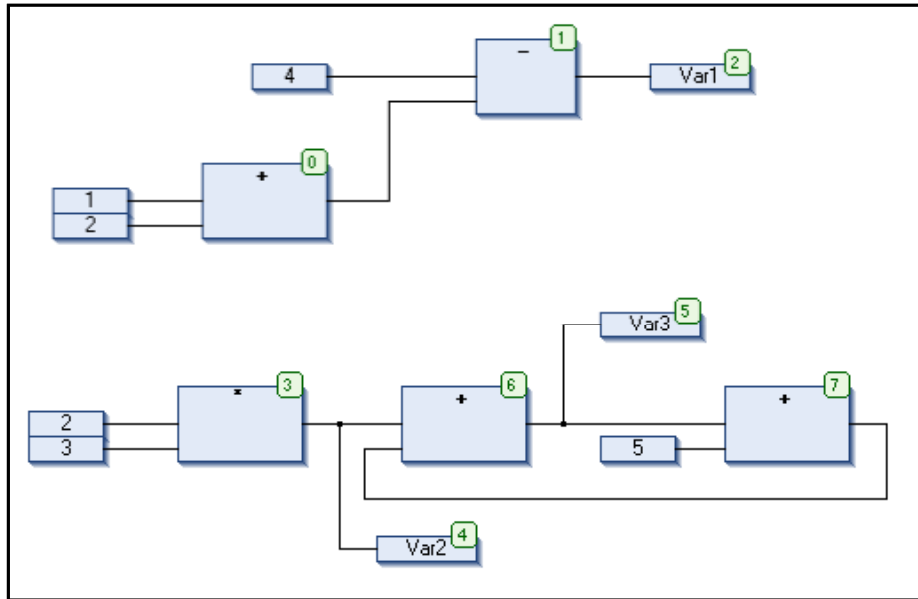


Fig.3-250: Example of execution order by dataflow

When the command is selected, the following happens internally first:

All elements are sorted topologically. Then a new processing list is put together:

based on the known values of the inputs it is determined which of the elements that are not yet numbered can be processed next.

In the upper "network", for example, the function block ADD (2) can be processed immediately, since the values at its inputs ("1" and "2") are known. Only then can the function block SUB (1) be processed, since the result of ADD has to be known, etc.

Feedback, however, is added last. In this way, the new execution order is formed according to dataflow.

Menu Items

The advantage of a sequence based on dataflow is that the output box, which is connected with the output of a function block, follows that output immediately during execution, which is not always the case with topological arrangement.



The topological sequence might also provide a different result than the sequence according to dataflow, as shown from the examples above.

Arrange Topologically

Menu: **CFC ▶ Arrange topologically**

This command is used to arrange all of the currently selected elements in the CFC editor in the topologic [execution order](#), [page 315](#), (indicated by the element numbers in the upper right corner of each element).

Topological arrangement means that processing occurs from left to right and from top to bottom, i.e. with topologically ordered elements, the element numbers that indicate the position of an element in the processing list increase from left to right and from top to bottom. The position of the connections are not significant. Only the position of the elements is important.

When executing the order, first all of the selected elements are removed from the processing list in an internal procedure; then they are added back into the remaining list one by one from bottom right to top left. Each highlighted element is inserted in the processing list before the topological successor. This is shown in an example.

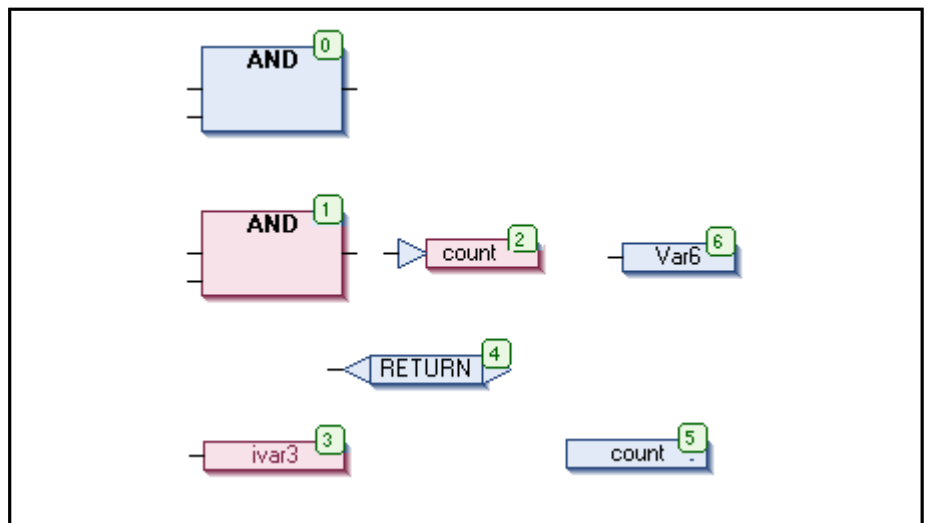


Fig.3-251: Example of topological arrangement, before

The elements with numbers 1, 2 and 3 are selected.

If the 'Arrange topologically' command is selected now, the three selected elements are first removed from the processing list. The following re-insertion is completed in reverse order as compared to the removal:

First of all, ivar3 is arranged in front of the count label, i.e. it becomes 4 and consequently RETURN falls back to 3.

Then, the count jump is arranged in front of Var6 and therefore receives 5.

Consequently, the count label (which just was on 5), ivar3 output and RETURN are in each case moved one downward.

Finally, the AND block is re-inserted in front of the count jump and therefore receives 4.

Menu Items

Therefore, the count label (which just was on 4), ivar3 output and RETURN fall by 1 again. This results in the following new processing order: sequence after

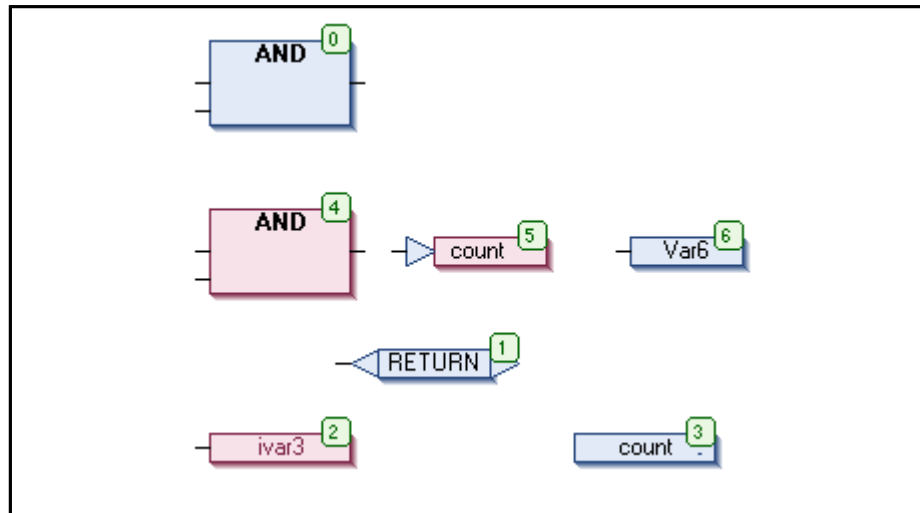


Fig.3-252: Example of topological arrangement, after

When adding a new element, by default it is inserted before its topological successor in the processing list.



The topological sequence might also provide a different result than the sequence according to dataflow, as shown from the examples above.

Set Execution Order

Menu: **CFC ▶ Set execution order...**

With this command, the element number of the currently selected element can be redefined in order to change the position of the element within the [execution order, page 315](#).

The command opens the 'Set execution order' dialog.

The current element number is displayed in the **Current Execution Order** field and the new number can be entered in **New Execution Order**. The possible values are given in parentheses.

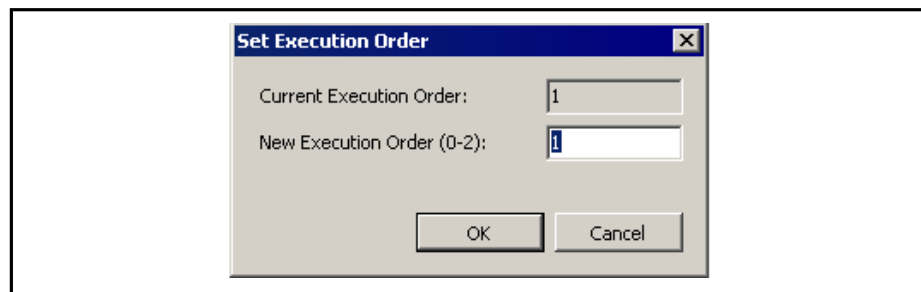


Fig.3-253: 'Execution order' dialog

3.12.7 Edit Parameters

Menu: **CFC ▶ Edit Parameters...**

Constant input parameters (VAR_INPUT CONSTANT) of functions and function blocks are not directly displayed in the CFC editor (see the following example for a declaration).

Example for declaration

Declaration of VAR_INPUT CONSTANT

Menu Items

```
FUNCTION_BLOCK fublo2
VAR_INPUT CONSTANT
  fbin1:INT;
  fbin2:DWORD:=24354333;
  fbin3:STRING:='hallo';
END_VAR
...
```

In order to access these parameters, use the 'Edit parameters' command which opens the dialog with the same name. Here the name (parameter), data type (type) and initialization value are displayed for each parameter.

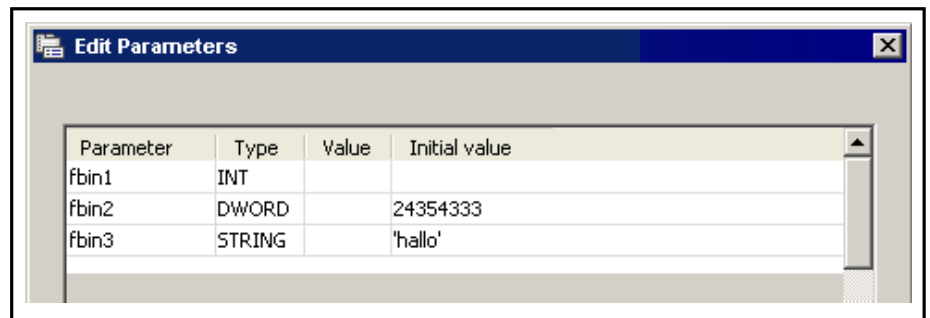


Fig.3-254: 'Edit parameters' dialog

The values of the constant input parameters (VAR_INPUT CONSTANT) can be changed here.


To do this, click on the respective field in the value column to select the value and click again (or press <Enter>) to open the input field. Enter the desired value and confirm it with the <Enter key>. <Cancel> discards the entry. <OK> saves all of the changes made.



This functionality and the declaration of variables with the keyword "VAR_INPUT CONSTANT" applies only for the CFC editor.

In the FBD editor all of the input parameters are always displayed at the function block, no matter if they are declared as VAR_INPUT or VAR_INPUT CONSTANT. Even for text editors this does not make any difference.

3.12.8 Connect Selected Connections

Icon: 

Menu: **CFC ▶ Connect Selected Connections**

This command can only be executed if exactly one output and several inputs are selected.

Press and hold the <Ctrl> key and click on the desired inputs and outputs to select them.

When the command is called, a connection is established between the connection of the output and the connections of the inputs.

3.12.9 Reset Connections

Icon: 

Menu: **CFC ▶ Reset connections**

If an input pin or an output pin was removed from a function block, e.g. since it was not used, or if the interface of a function block has changed, this command can be used to display all of the inputs and outputs of the function block defined in its implementation again.

Menu Items

The command can also be used to display the type VAR_IN_OUT parameters for a function block, which are hidden by default.

In the following example, the input pin fbin2 of the function block instance was deleted in the CFC editor, since it was not used. When the instance element is selected and the "Reset connections" command is executed, fbin2 is displayed again.

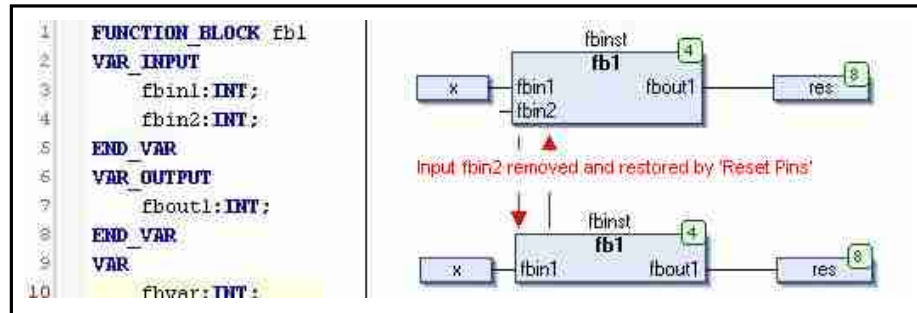


Fig.3-255: Example of resetting connections

3.12.10 Remove Unconnected Connections

Icon:

Menu: **CFC ▶ Remove Unconnected Connections**

In the currently selected editor, this command removes the unconnected connections from the program call, function block or for inst actions that are not local.

However, the disconnected connections are not removed from calls of functions, methods or operators, since it would result in invalid syntax.

3.13 Sequential Function Chart - Commands

3.13.1 Sequential Function Chart - Commands, Overview

The commands for Sequential Function Chart (SFC) are available for programming in the [SFC editor, page 399](#).

By default, they are included in the SFC menu which appears in the menu bar if the SFC editor is active.

Alternatively, they are provided in the context menu for the SFC elements in the working sheet.

An [SFC elements Toolbox, page 405](#) is ### in preparation ###.

If necessary, use the dialog under **IndraWorks ▶ Tools ▶ Customize ▶ Commands ▶ SFC** to change the menu configuration.

The commands:

- [Initial step, page 279](#)
- [Add input action, page 279](#)
- [Add output action, page 280](#)
- [Add step transition, page 280](#)
- [Add step transition after, page 281](#)
- [Parallel, page 281,](#)
- [Alternate, page 282](#)
- [Add branching, page 282](#)
- [Add branch right, page 282,](#)
- [Add action association, page 283,](#)

- [Add action association after, page 284](#)
- [Add jump, page 284](#)
- [Add jump after, page 284](#)
- [Add macro, page 285,](#)
- [Add macro after, page 285](#)
- [Display macro, page 285,](#)
- [Exit macro, page 286,](#)

3.13.2 Initial Step

Icon:

Menu: **SFC ► Initial step**

This command is used to perform the currently selected step into the [initial step, page 405](#).

After the command is executed, the frame of the step element becomes a double line.

The step that was previously the initial step automatically becomes a normal step and is displayed with a simple frame.

The command can be useful if converting the chart.

When a new SFC object is created, it automatically contains an initial step, followed by a transition (TRUE) and a jump back to the initial step.

The ["Initial step" property404](#), can also be enabled or disabled in the element properties of a step element; however, there is no automatic adjustment of this setting for other steps.

Note that the diagram to the initial step can be reset with the following [variables, page 414](#), "SFCInit" or "SFCReset".

3.13.3 Add Input Action

Icon:

Menu: **SFC ► Add input action**

This command is used in the SFC editor to add an input action to the currently selected position.

"<stepname>_entry" is the suggested name. It can be changed. The language for implementing the action can also be selected.

An input action is only executed once, right after the step has become active. A step with an input action is designated by an "E" in the upper left corner.

In the Project Explorer, the input action appears under the POU that contains the SFC.

To edit an input action in the sequence, double-click the corner where the "E" is located in the step.

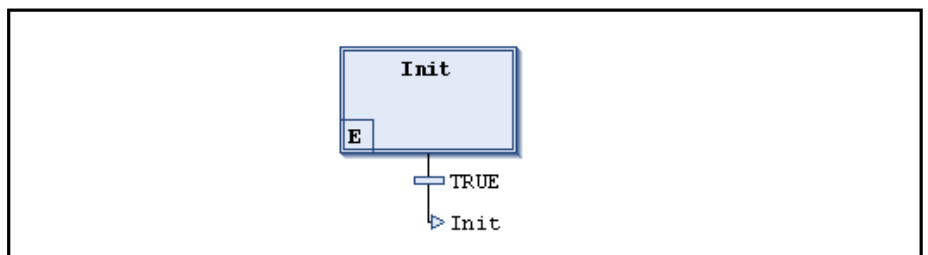


Fig.3-256: Input action added

Menu Items

3.13.4 Add Output Action

Icon:

Menu: **SFC ▶ Add output action**

This command is used in the SFC editor to add an output action to the currently selected position.

"<stepname>_exit" is the suggested name. It can be changed. The language for implementing the action can also be selected.

An output action is only executed once, before the step is disabled. A step with an output action is designated by an "X" in the lower right corner.

In the Project Explorer, the output action appears under the POU that contains the SFC.

To edit an output action in the sequence, double-click the corner where the "X" is located in the step.

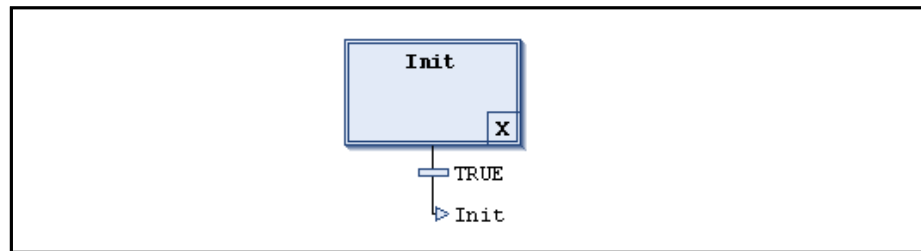


Fig.3-257: Output action added

3.13.5 Add Step Transition

Icon:

Menu: **SFC ▶ Add step transition**

This command is used in the SFC editor to add a step and a transition **before** the currently selected position.

The arrangement (order) of the new step-transition combination depends on which step or which transition is currently selected when adding them. The basic order step-transition-step-transition-... is automatically used.

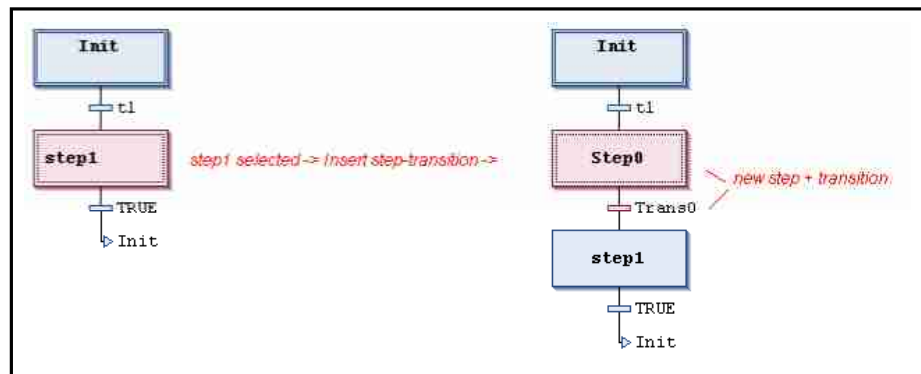


Fig.3-258: Step-transition added after step

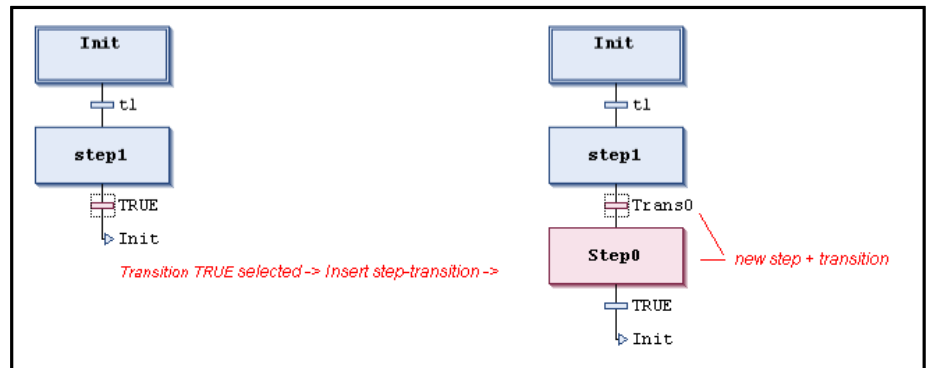


Fig.3-259: Step-transition added after transition

By default, the new step is named "Step<n>".

n is a consecutive number, starting with "0" for the first step, that is added to the initial step.

By default, the new transition is accordingly named "Trans<n>".

The default names can be edited directly (click on the text to open an input field).

3.13.6 Add Step Transition After

Icon:

Menu: **SFC ▶ Insert Step Transition Later**

This command is used in the SFC editor to add a step and a transition **after** the currently selected position.

The arrangement (order) of the new step-transition combination depends on which step or which transition is currently selected when adding them. The basic order step-transition-step-transition-... is automatically used.

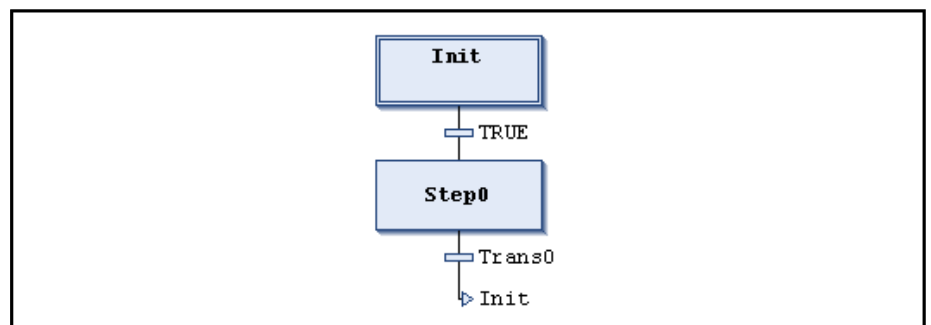


Fig.3-260: Example of a step-transition added "after"

By default, the new step is named "Step<n>". n. "n" is a consecutive number, starting with "0" for the first step, that is added to the initial step.

By default, the new transition is accordingly named "Trans<n>".

The default names can be edited directly (click on the text to open an input field).

3.13.7 Parallel

Icon:

Menu: **SFC ▶ Parallel**

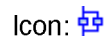
This command is used in the SFC editor to convert the currently selected "alternate branch" into a "parallel branch".

Menu Items



After a branch is converted, the arrangement of the steps and transitions before and after the branch has to be checked and adapted.

3.13.8 Alternate



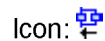
Menu: **SFC ▶ Alternative**

This command is used in the SFC editor to convert the currently selected "parallel branch" into an "alternate branch".



After a branch is converted, the arrangement of the steps and transitions before and after the branch has to be checked and adapted.

3.13.9 Add Branch

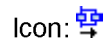


Menu: **SFC ▶ Add branch**

This command is used in the SFC editor to add a "branch" **to the left** of the current position.

See also the description for [Add branch right](#).

3.13.10 Add Branch Right



Menu: **SFC ▶ Add branch right**

This command is used in the SFC editor to add a "branch" **to the right** of the current position. (To add the branch to the left, use the [Add branch](#) command).

- If the topmost element among the currently selected elements is a transition or an alternate branch, an "alternate branch" is added.
- If the topmost element among the currently selected elements is a step, a macro, a jump or a parallel branch, a "parallel branch" with the **jump label** "Branch<x>" is added, where x is a consecutive number, starting with "0" for the first label in the diagram. This default name for the jump label can be edited. The jump label can be indicated as the target for a [jump](#).
- If a common element is currently selected in an existing branch (horizontal line), the new branch is added as new arm at the complete right. If an entire arm of an existing branch is currently selected, the new branch is added directly to the right as new arm.



Note that a branch added with the command "[Alternate](#)" or "[Parallel](#)" can be transferred into the other type.

Example:

Parallel branch:

In the following figure is a parallel branch that was just added, created with the "Add branch right" command while step11 was selected. A step ("Step2" in the example) is automatically added.

Menu Items

Processing in online mode: If t2 delivers the value TRUE, Step2 is immediately executed after step11 and before t3 is evaluated.

In this case, in contrast to alternate branches, both branches are executed.

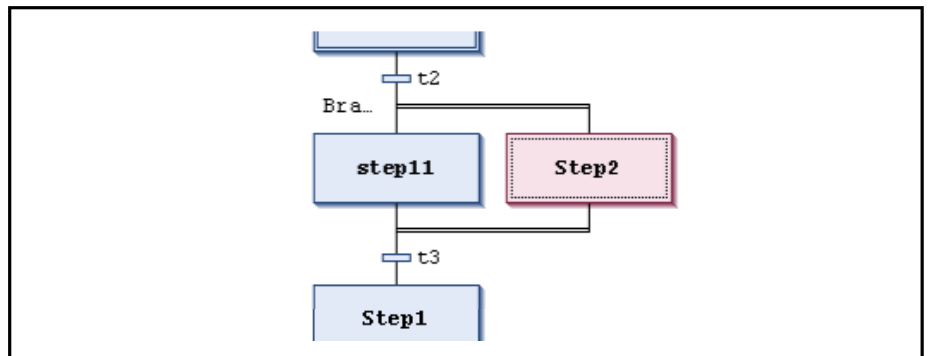


Fig.3-261: Parallel branch

Example:

Alternate branch:

In the follow figure is an alternate branch that was just added, created with the "Add branch right" command while Transition t4 was selected. A step ("Step32" in the example), and two transitions (t41, t42) are added, one before and one after.

Processing in online mode: When Step3 is active, the subsequent transitions (t4, t41) are evaluated from left to right. The first arm of the branch in which the first transition delivers the value TRUE is executed. In this way, in contrast to a parallel branch, only one arm is executed.

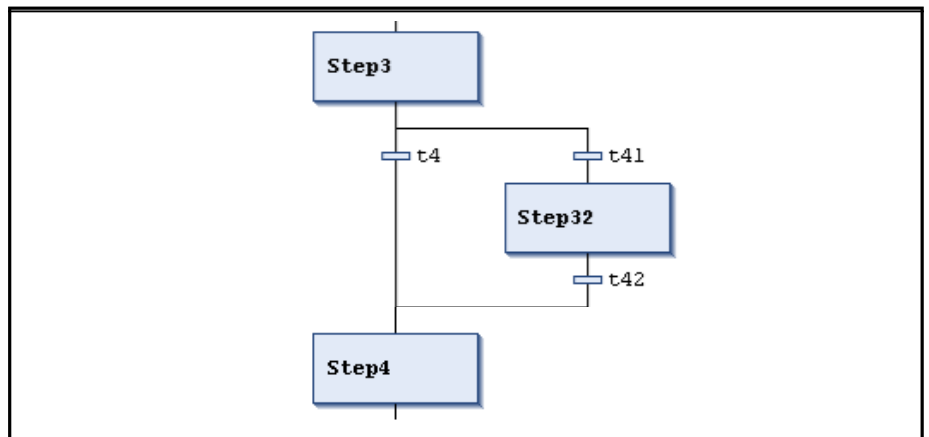


Fig.3-262: Alternative branch

3.13.11 Add Action Association

Icon: 

Menu: **SFC ► Add action association**

This command is used in the SFC editor to assign an "IEC action" to a step.

The action element is added to the right, next to the currently selected step element.

If the step has been assigned one or more actions, they are displayed in an "actions list". The new action is positioned as follows:

Menu Items

- If the step element is selected, it is positioned as the first action of the step, i.e. in the top position in the action list.
- If one of the existing actions in the action list for the step is selected, it is positioned directly before this action, i.e. directly above.

See also [Add action association after](#).

The left section of the action element contains the **qualifier**, "N" by default. Enter the **action name** in the right section. To do this, click in the box to obtain an editing frame.

An action with the name entered has to be created in the project as a POU. See the [Add object234](#) command.

The qualifier can also be edited.

See the [List of valid qualifiers, page 414](#).

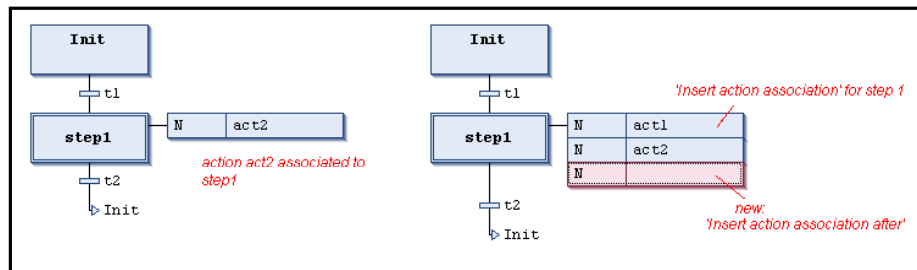


Fig.3-263: Example of actions assigned to a step

3.13.12 Add Action Association After

Icon:

Menu: **SFC ▶ Add action association after**

This command is used in the SFC editor to assign another "IEC action" to a step.

See also in this regard the description for the [Add action association](#) command. The difference is that with "add after", the new action is not placed in the first position, but instead in the last position in the actions list, i.e. not above, but instead, below the action currently selected in this list.

[SFC elements, ToolBox, page 405](#)

3.13.13 Add Jump

Icon:

Menu: **SFC ▶ Add jump**

This command is used in the SFC editor to add a "jump element" **before** the currently selected element.

The jump is automatically added with the jump target "Step". Replace "Step" with the name of a step or the jump label of a "parallel branch" to which the jump is to be made.

[SFC elements, ToolBox, page 405](#)

3.13.14 Add Jump After

Icon:

Menu: **SFC ▶ Add jump after**

This command is used in the SFC editor to add a "jump element" **after** the currently selected element.

Menu Items

The jump is automatically added with the jump target "Step". Replace "Step" with the name of a step or the jump label of a "parallel branch" to which the jump is to be made.

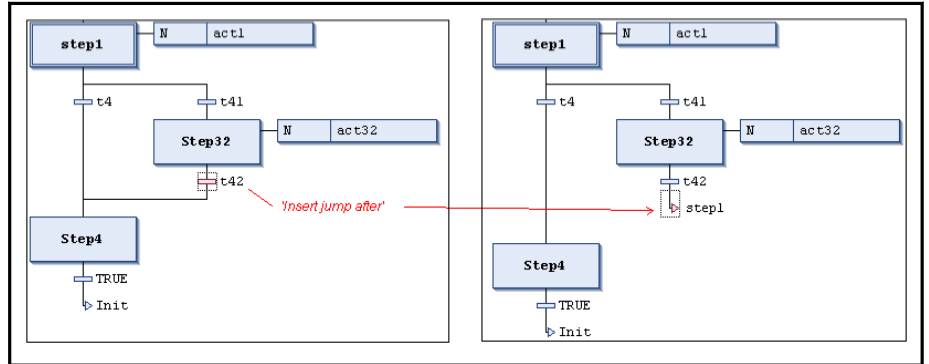


Fig.3-264: Example of a jump added after SFC elements, ToolBox, page 405

3.13.15 Insert Macro

Icon:

Menu: **SFC ▶ Insert macro**

This command is used in the SFC editor to add a macro element **before** the currently selected element.

By default, the macro is named "Macro<x>", where x is a consecutive number, beginning with "0" for the first macro inserted.

This name can be edited.

To edit or display a macro, the [Display macro](#) command can be used to open the macro editor.

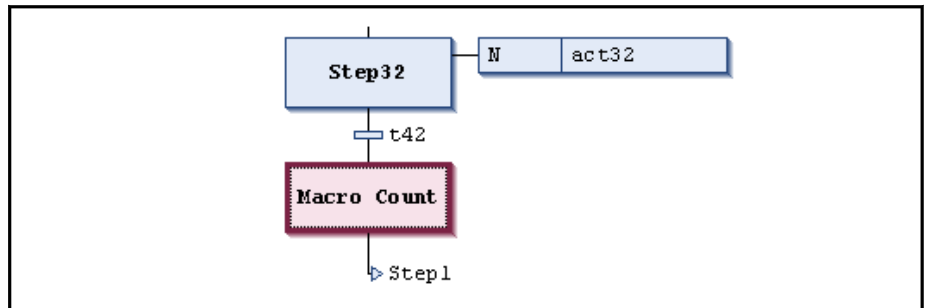


Fig.3-265: Example of macro in an SFC chart (currently selected)

3.13.16 Add Macro After

Icon:

Menu: **SFC ▶ Add macro after**

This command is used in the SFC editor to add a macro after the currently selected element. See the related information at [Add macro](#).

3.13.17 Display Macro

Icon:

Menu: **SFC ▶ Display macro**

This command is available in the SFC editor if a macro element is selected in the SFC chart.

Menu Items

It is used to open the macro in the macro editor to view or edit it.

The main view of the SFC editor is closed and the macro editor is opened. It is also an SFC editor in that you can now view and edit a section of the SFC diagram that is only displayed in the main view as a macro box. A zoom menu is also available here in the lower right corner.

To return to the main view of the SFC editor, use the [Exit macro](#) command.

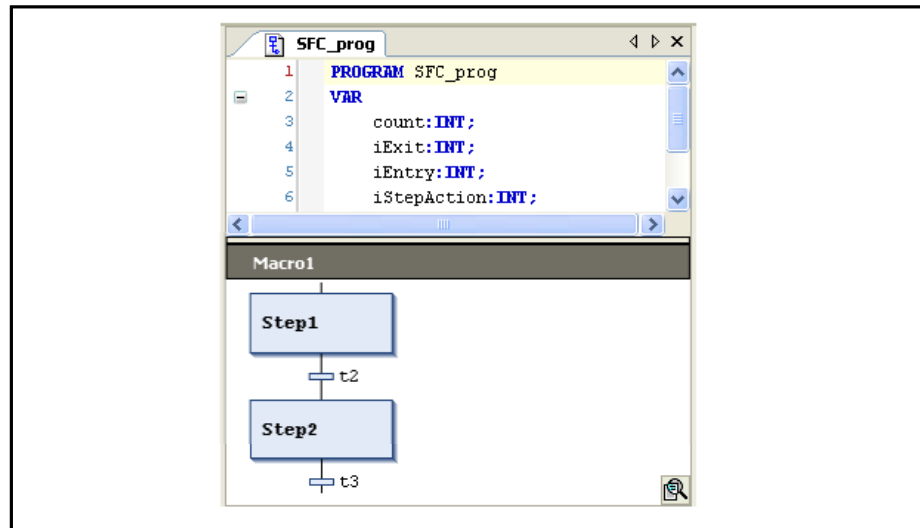


Fig.3-266: Example of the macro editor

The command can be used in offline and online mode.

3.13.18 Exit Macro

Icon:

Menu: **SFC ▶ Exit macro**

This command is available in the SFC editor when the macro editor is currently open (using the [Display macro](#) command).

The macro editor is closed again and you return to the main view of the SFC editor.

The command can be used in offline and online mode.

3.14 Text Lists - Commands

3.14.1 Text Lists - Overview of Commands

The commands for working with text lists are described in this section.

By default, these commands are available in a 'Text list' menu as soon as a text list is being edited or partially in the 'VI-Logic visualization' menu.

If required, the menu structure can be reconfigured via the **IndraWorks ▶ Tools ▶ Customize**.

Commands:

- [Create global text list, page 287](#)
- [Add language, page 287](#)
- [Remove language, page 288](#)
- [Import/Export text lists, page 288](#)
- [Update visualization text IDs, page 293](#)
- [Check visualization text IDs, page 293](#)

- [Remove unused text list entries, page 294](#)

3.14.2 Create Global Text List

Icon: 

Use "Create global text list" to create a global text list explicitly.


The object is named "GlobalTextList" and is added to the "General module" folder.

To create a global text list, in the main menu click on **VI Logic Visualization ► Create global text list**. Alternatively, "Create global text list" is available in the context menu if the cursor is in the working area of the visualization editor.

The "GlobalTextList" object is **automatically** created as soon as the first text is defined when configuring a visualization.

Refer to [text list, page 55](#) for detailed information.

3.14.3 Add Language

Icon: 

Use "Add language" to add a column in a text list for an additional local country language.

To add a new language in a text list, open the editor for the "text list" or "GlobalTextList" object in the working area first.

Then, in the main menu, click on **VI Logic text list ► Add language**.

This opens the "Choose language" dialog.

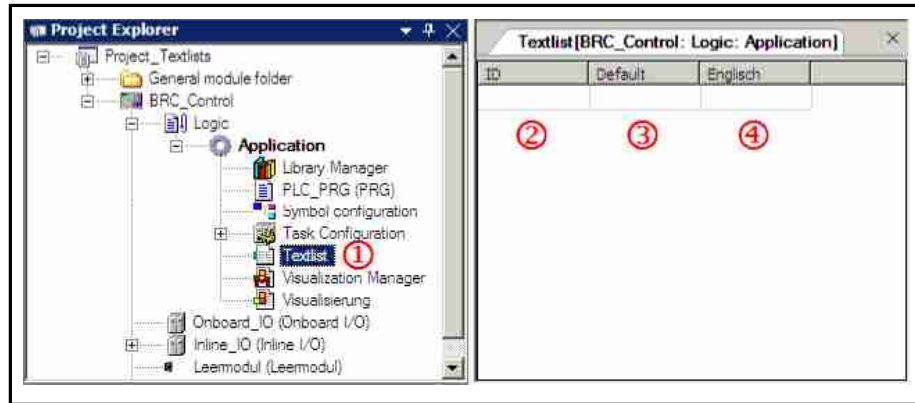
In the "Choose language" dialog, enter a name (column heading in the text list) for the new language and confirm your entry with the "OK" button.



Fig.3-267: 'Choose language' dialog

Then the corresponding column is created in the text list.

Menu Items



- ① Object text list
- ② Unique ID (qualifier, index) specified in the configuration of a visualization element.
- ③ Default language, always available. If there is no entry in the text list that matches the language currently set in IndraLogic, the entry defined as default is used.
- ④ Newly defined language, English in this case

Fig.3-268: Object text list

Refer to [text list](#), page 55 for detailed information.

3.14.4 Remove Language

Icon:

Use "Remove language" to remove the column for a certain language from the text list.

To remove a language from a text list, place the cursor in one of the column cells and click in the main menu on **VI Logic text list ► Remove language**.

Alternatively, "Remove language" is also available in the context menu.

3.14.5 Import/Export Text Lists

Icon:

Use "Import/Export text lists" to exchange data with other programs, e.g. Excel.

The data format used is .csv ("Comma Separated Values").

ID	Standard	DE	EN	RU	ZH
u0001	Device parameter	Geräte-Parameter	Device parameter	Параметры устройств	设备参数

Fig.3-269: Unicode text list object in IndraWorks

	A	B	C	D	E	F	G
1	TextList	Id	Default	DE	EN	RU	ZH
2	TextList_uni	u0001	Device parameter	Geräte-Parameter	Device parameter	Параметры устройств	设备参数

Fig.3-270: Unicode text list object opened in MS Excel



Both the text list object and the csv import/export format can be used for the characters.

The following dialog appears when calling the command

Menu Items

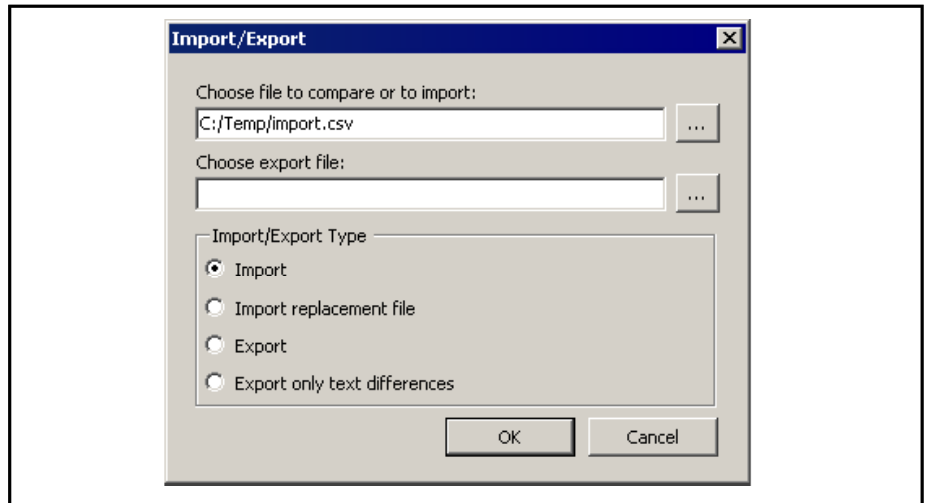



Fig.3-271: Import/Export dialog

The files for an import, export or for comparison can be determined by specifying the corresponding path or by using the input assistance (). Specify the actions to be carried out by a corresponding selection in the lower section of the dialog:

Import:

When an external file is imported, the data sets of the external file are compared with the data sets of the project. The data sets in the project are adapted according to the following rules:

- If the text content of the files is the same, the data set is not changed.
- If a translation has been added to the external file, it is accepted into the data set in the project.
- If the text has been modified within a translation, the change is accepted into the data set in the project.
- If translated texts are missing in the external file, the data set in the project is not changed.
- If a new line has been added to the external file, the new data set is accepted into the project.
- If the project contains an additional data set, it is retained.
- A change within the Default column can be considered the same as adding a new text. If text passages have, however, several spaces instead of one, this is not regarded as change!

See [example - importing a .csv file, page 289](#).

Example - Importing a .csv file

Example:

Data content in the external file:

Text list	ID	Standard	German	English
GlobalTextList		Automobile	Automobil	Automobile
GlobalTextList		Steering wheel	Lenkrad	Steering wheel
TextList1	0	Cancel	Abbrechen	Cancel



Menu Items

Text list	ID	Standard	German	English
TextList1	1	Door		
TextList2	2	Light		

Data content of the text lists for the current project before import:

Text list	ID	Standard	German	English
GlobalTextList		Automobile	Automobil	Automobile
GlobalTextList		Steering wheel	Lenkrad	Steering wheel
TextList1	0	Cancel	Abbrechen	Cancel
TextList1	1	Door	Tür	Door
TextList2	2	Seat	Sitz	Seat

During import, all differences are accepted into the project. At this time, both lists are compared and adapted so that the content of the following text lists in the project is created:

Result:

Text list	ID	Standard	German	English
GlobalTextList		Automobile	Automobil	Automobile
GlobalTextList		Steering wheel	Lenkrad	Steering wheel
TextList1	0	Cancel	Abbrechen	Cancel
TextList1	1	Door	Tür	Door
TextList1	2	Light		
TextList2	3	Seat	Sitz	Seat

Import list of changes (replacement file)

When importing text lists, a change within the Default column is considered the same as adding a new line. The reason for this is that the Default column serves as the key for comparing lines during export/import.

Note: If text passages in the "Default" column have, however, several spaces instead of one, this is not regarded as change!

If the text within the Default column is to be changed (correcting a typing error or supplementing existing text), a list of changes (replacement file) is required.

Example - Importing a list of changes

Default, old	Default, new	Command
Cancel ?	Cancel	REPLACE
Do you want to save ?	Do you really want to save ?	REPLACE_AND_REMOVE
Do you really want to save ?!	Do you really want to save ?	REPLACE_AND_REMOVE

The list of changes is processed from top to bottom. This way, the change history can also be considered.

The command describes what is to be done with the text line.



Menu Items

Currently, only the **REPLACE** command is available. It has the following effect:

Normally, the text in the "Default" column is replaced by the new text. In the example, "Cancel ?" is replaced by "Cancel" and "Do you want to save ?" by "Do you really want to save ?".

At the same time, the texts for all of the visualization elements are adapted, i.e. the old entries are replaced by the new texts here as well.

If the new default text is already contained in a field of the "Default" column in another line in the text list, the line that contains the entry to be replaced is deleted entirely. The respective visualization elements receive entries from the remaining column with the same default entry.

In the example, this occurs for the entry "Do you really want to save !", which is to be replaced by "Do you really want to save ?". Since the REPLACE command is called for the entry "Do you really want to save !", based on the history of the changes, at this point in time there is already a default entry "Do you really want to save ?". To prevent this key from **appearing multiple times**, the line with the old default entry "Do you really want to save !" is **deleted** entirely from the text list.

Export When exporting text lists, all of the changes within the project are compared with an external comparison file. A new export file is created, which follows these rules:

1. If the text contents are identical, the data set is exported unchanged.
2. If a translation has been added to the project, it is accepted as new.
3. If the text has been modified within a translation, the changed text from the project is used for the new data set.
4. If translated texts are missing in the project, the translations of the template are used for the new data set.
5. If a line has been added to the project, it is accepted as a new data set.
6. If an additional line is contained in the external file, it is exported again.
7. A change within the "Default" column can be considered the same as adding a new text.

See [example - exporting a .csv file, page 291](#).

Example - Exporting a .csv file

Example:

Data content in the external file:

Text list	ID	Standard	German	English
GlobalTextList		Automobile	Automobil	Automobile
GlobalTextList		Steering wheel		
TextList1	0	Cancel	Abbrechen	Abort
TextList1	1	Door	Tür	Door
TextList2	2	Seat	Sitz	Seat

Data content of the text lists for the current project before export:

Text list	ID	Standard	German	English
GlobalTextList		Automobile	Automobil	Automobile
GlobalTextList		Steering wheel	Lenkrad	Steering wheel



Menu Items

Text list	ID	Standard	German	English
TextList1	0	Cancel	Abbrechen	Cancel
TextList1	1	Door		
TextList1	3	Light		
TextList2				

When exporting, all differences are accepted into the external file. At this time, both lists are compared and adapted such that the following external file is created.

Data content of the external file after the export:

Text list	ID	Standard	German	English
GlobalTextList		Automobile	Automobil	Automobile
GlobalTextList		Steering wheel	Lenkrad	Steering wheel
TextList1	0	Cancel	Abbrechen	Abort
TextList1	1	Door	Tür	Door
TextList1	3	Light		
TextList2	2	Seat	Sitz	Seat

Export only text differences:

If this option is selected, only the lines that differ from each other in different versions are accepted into the export file. Such difference files are e.g. suitable as template for new compilations to be prepared. As it is to be kept as small as possible, missing entries in the current lists are not considered as deviation.

Also see [example - exporting the text differences, page 292](#).



To find the corresponding data set, the "Default" column is used for the **GlobalTextList** and the ID column is used for all other text lists. For this reason, the ID column is empty for all data sets of the GlobalTextList.

Example - Exporting the text differences

Example:

Data content in the external file:

Text list	ID	Standard	German	English
GlobalTextList		Automobile	Automobil	Automobile
GlobalTextList		Steering wheel		
TextList1	0	Cancel	Abbrechen	Abort
TextList1	1	Door	Tür	Door
TextList2	2	Seat	Sitz	Seat



Menu Items

Data content of the text lists for the current project before export:

Text list	ID	Standard	German	English
GlobalTextList		Automobile	Automobil	Automobile
GlobalTextList		Steering wheel	Lenkrad	Steering wheel
TextList1	0	Cancel	Abbrechen	Cancel
TextList1	1	Door		
TextList1	3	Light		
TextList2				

During export, all of the lines that differ from each other in different versions (lines 2, 3 and 5 of the current list) are accepted into the export file.

Data content of the external file after the export:

Text list	ID	Standard	German	English
GlobalTextList		Steering wheel	Lenkrad	Steering wheel
TextList1	0	Cancel	Abbrechen	Cancel
TextList1	3	Light		

3.14.6 Update Visualization Text IDs

Icon:

If a static text is modified within a visualization element, the visualization and perhaps the GlobalTextList have to have write access (see [TextList/Global-TextList, page 55](#)).



If they are modified even though no changes were made to the write permission, it can happen that the text IDs no longer match the texts within a visualization element.

The [Check visualization text IDs, page 293](#), command can determine these kinds of errors within all visualizations.

The "Update visualization text IDs" command can automatically correct these cases of error. To do this, all visualizations with error cases and the Global-TextList have to have write access.

3.14.7 Check Visualization Text IDs

Icon:

If a static text is modified within a visualization element, the visualization and perhaps the GlobalTextList have to have write access (see [TextList/Global-TextList, page 55](#)).

If they are modified even though no changes were made to the write permission, it can happen that the text IDs no longer match the texts within a visualization element.

The "Check visualization text IDs" command can determine these kinds of errors within all visualizations.

The [Update visualization text IDs, page 293](#), command can automatically remedy these kinds of errors within all visualizations.

Menu Items

3.14.8 Remove Unused Text List Entries

Icon:

This command is used to remove texts within the GlobalTextList that are not used in any visualization element.

3.15 Trace - Commands

3.15.1 Trace - Commands, Overview

The commands in the menu item "Trace" are used to work with the [trace editor](#), [page 431](#).

The menu item appears in the main menu or as a context menu (right click in the left section of the window) when the trace editor is active.

If required, the menu structure can be reconfigured via the **IndraWorks ▶ Tools ▶ Customize ▶ Commands** dialog.

Commands:



- [Login trace \(download\)](#), [page 294](#),
- [Start/stop trace](#), [page 294](#),
- [Reset trigger](#), [page 295](#)
- [Cursor](#), [page 295](#)
- [Scrolling with mouse](#), [page 296](#),
- [Zooming with mouse](#), [page 297](#),
- [Reset display](#), [page 297](#)
- [Compress](#) [297](#),
- [Expanding](#), [page 298](#),
- [Multi-channel](#), [page 298](#),
- [Upload trace...](#), [page 299](#)
- [Load trace...](#), [page 299](#)
- [Save trace...](#), [page 299](#)
- [Shortcuts in the trace diagram](#), [page 444](#).

3.15.2 Login Trace (Download)


This command is used to load the code generated for the trace explicitly in the control in order to enable tracing there. This is required for the first use of the trace functionality on an application and later as well if the trace configuration or the application program have been modified.


See also [trace editor in online mode](#), [page 442](#).

3.15.3 Start/Stop Trace

Icons: , 

This command toggles between the start and stop tracing commands.

If tracing is stopped, the  symbol is displayed. Usage of the command then starts tracing. This restarts the data acquisition on the runtime system and the current values are continuously displayed in the trace editor.

If tracing is just running, the  symbol is displayed. Use of the command stops tracing on the runtime system and the latest data acquired is then displayed in the trace editor.

3.15.4 Reset Trigger

Icon:

This command resets the trigger so that the trace is restarted after the trigger has been enabled.

3.15.5 Cursor

Icon:

If the "Cursor" command is activated, the cursors can be managed in the trace diagram of the [trace editor, page 431](#).

The command is disabled when one of the commands [Scrolling with mouse, page 296](#), or [Zooming with mouse, page 297](#), is enabled.

All trace shortcuts are listed in the [Shortcuts in the trace diagram, page 444](#).

One cursor

If there is no cursor, one is created by clicking into the trace diagram.

One cursor is displayed as black triangle located at the upper edge of the diagram and connected with a fine black line ending at the lower edge of the diagram. The related time value is displayed in the status bar. To move the cursor, click and drag it to the desired position while keeping the left mouse button pressed.

It can also be approached using the <Left> or <Right> arrow keys. The selected cursor is moved step by step as long as the key is held down.

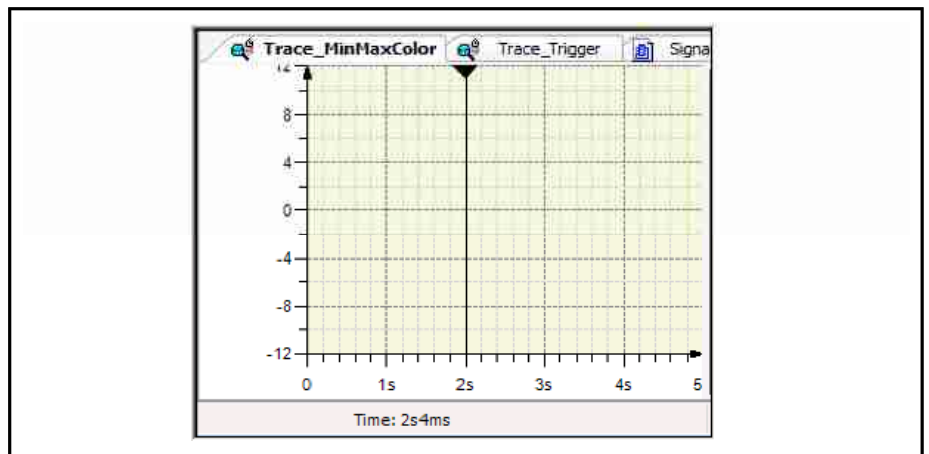


Fig.3-272: Trace diagram with **one** cursor

The commands are also available if the "Zooming with mouse" or "Scrolling with mouse" mode is enabled.

If you are in the cursor mode, you can also click at the desired position and the cursor exactly positions at this place.

Two cursors

If there is one cursor, can generate a second one by clicking on the command. The related time of both cursors as well as their time difference in brackets is now displayed in the status bar as long as the cursor mode is enabled.

Menu Items

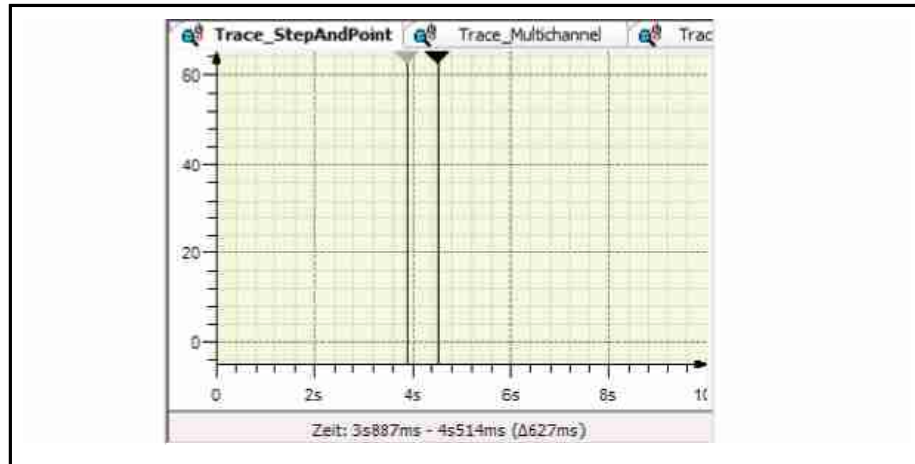


Fig.3-273: Trace diagram with **two** cursors

Move one cursor by clicking and moving it along the time axis while keeping the left mouse button pressed.

This is also possible using the <Left> or <Right> arrow keys to move the **selected** cursor.

In order to move the **unselected** cursor, use the <Left>/<Right> arrow keys while keeping the <Shift> key pressed.

Deleting the cursor

If there are two cursors, they can both be deleted by clicking on .

If or is used, only the **selected** cursor is removed. These commands can also be used if you are in the "Zooming with mouse" or "Scrolling with mouse" mode.

If there is no cursor in the trace diagram, the time value and the y-value of the mouse pointer position are displayed in the status bar.

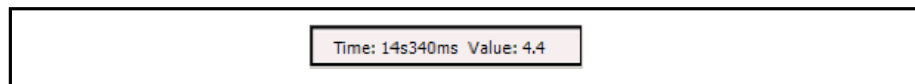


Fig.3-274: Status bar: Time and y-value of the mouse pointer position

3.15.6 Scrolling with Mouse

Icon:

This command activates the "Scrolling with mouse" mode

If it is activated and you click into the trace diagram while keeping the left mouse button pressed simultaneously, the mouse cursor is displayed as double arrow and you can scroll the time axis of the currently displayed trace by means of mouse movements. Using the <Left>/<Right> arrow keys while keeping <Alt> pressed simultaneously has the same effect.

If you click into the diagram and hold the left mouse button and the <Ctrl> key pressed at the same time, the mouse cursor is displayed as double up/down arrow and you can scroll the y-axis of the currently displayed trace by means of mouse movements. The same is achieved by using the <Up> or <Down> arrow keys while keeping <Ctrl> key pressed.

If the "Multi-channel" option is enabled, you can scroll the time and y-axis of all diagrams in the same way with mouse or keyboard. Horizontal scrolling acts on the time axis, vertical scrolling acts on the y-axis of the selected diagram.

To disable the "Scrolling with mouse" mode, activate the "Cursor" mode, page 295 or the "Zooming with mouse" mode, page 297,.


To reset the display, use the [Reset display, page 297](#) command.

All keyboard commands are listed in [Shortcuts in the trace diagram, page 444](#).

3.15.7 Zooming with Mouse

Icon: 

This command activates the "Zooming with mouse" mode.

If it is enabled, the mouse cursor is displayed as  and by keeping the left mouse button pressed, draw a rectangle in the trace window to reset the area of the displayed trace graph. If the mouse button is released, the diagram is zoomed in time and y-direction so that the content of the rectangular now fills the entire diagram.

To disable the zoom mode, enable the [cursor mode, page 295](#) or the "[Scrolling with mouse" mode, page 296](#).,.

To reset the display, use the [Reset display, page 297](#) command.


All keyboard commands are listed in [Shortcuts in the trace diagram, page 444](#).



Note that the **scroll mouse** or the keyboard can also be used to zoom in the trace window:

- Spinning the mouse wheel enlarges/reduces the size of the coordinate system along the x- **and** y-axis. The same action can be performed in the number pad using the <+> and <-> keys.
- It is only zoomed along the x-axis by spinning the mouse wheel while pressing and holding the <Shift> key. The same action can be performed in the number pad by using the <+> and <-> keys while holding down the <Shift> key.
- It is only zoomed along the y-axis by spinning the mouse wheel while pressing and holding the <Ctrl> key. The same action can be performed in the number pad by using the <+> and <-> keys while holding down the <Ctrl> key.


3.15.8 Reset Display

Icon: 

This command can be used to reset the display of the recording(s) to the default values after they have been changed by zooming, for example.

The default settings are defined in the [configuration dialog, page 434](#).,.

3.15.9 Compress

Icon: 

Using this command, the time area that is displayed in the trace editor can be enlarged by a fixed percentage. The graph is compressed. The command can be executed several times.

Use this command to display the values in the sampled graphs in an compressed fashion. With further compressions performed one after another the size of the section of the trace displayed in the window can be enlarged even more.

Menu Items

This command is the counterpart to [expand](#).

3.15.10 Expand

Icon:

Using this command, the size of the time area that is displayed in the trace editor can be reduced by a fixed percentage. The graph is expanded.

This command is the counterpart to [Compress](#).

3.15.11 Multi-Channel

This command enables the view of the Trace graphs to be changed. By default, the graphs of all variables are shown in the same diagram. In Multi-channel view, the graph of each variable is visualized in its own diagram with with an identical X-axis. Zooming and scrolling simultaneously affect the X-axis of all diagrams.

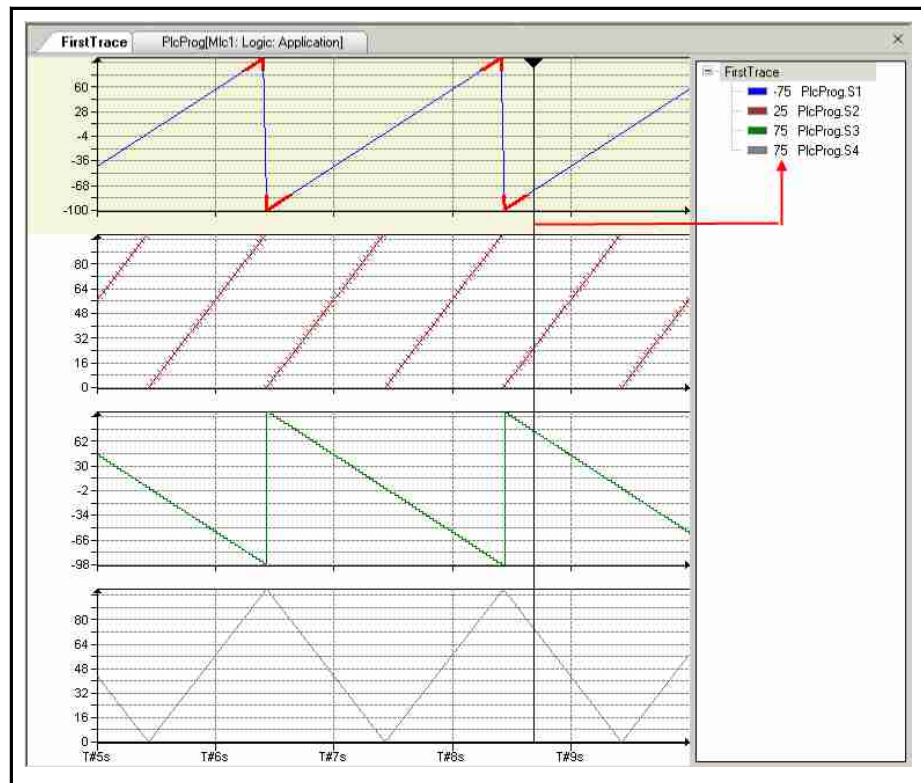


Fig.3-275: Trace in the multi-channel mode

All keyboard commands are listed in [Shortcuts in the trace diagram, page 444](#).

3.15.12 Online List...

This command opens a dialog listing all currently existing control traces. If the trace object is located under an application, all traces from there are listed. If the trace object is placed below a device, the traces of all applications are displayed and moreover the device-specific traces that are implemented in the device.

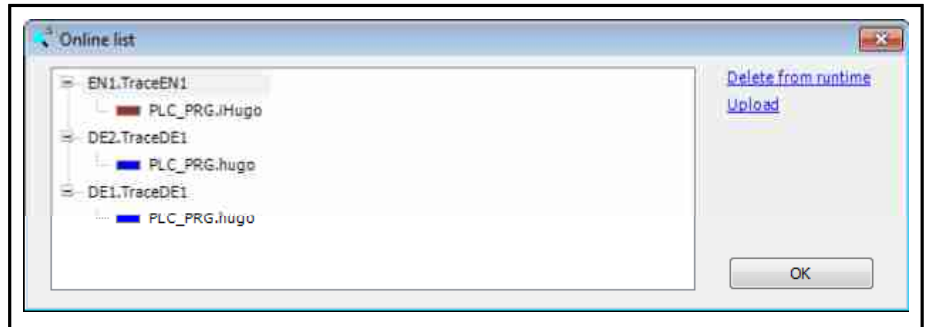


Fig.3-276: "Online list" dialog

Remove from runtime system: By means of this command, the selected trace on the runtime system is removed.

Upload: By means of this command, the selected trace is uploaded from the runtime system into the trace editor.

This command is only available if tracing in the runtime system is carried out by the 'CmpTraceMgr' component.

3.15.13 Upload Trace

This command loads all existing traces. If the trace object is positioned below an application, a trace of the runtime system with the same name as the trace object is uploaded into the trace editor and the previous trace configuration and record is overwritten. If the trace object is positioned below a device, every trace that ran on the runtime system can be loaded into the trace editor.

The extended name with instance path provides for unique names (e.g. "Application.Trace.MyRecord").

3.15.14 Load Trace...

This command is only available if tracing is carried out in the 'CmpTraceMgr' runtime system component.

Use this command to load the trace previously saved in a file into the editor again, completely with configuration, data and time stamp.

The file can be generated using the [Save trace...](#) page command.

Calling the command opens the "Load Trace" dialog (default dialog for browsing the file system) and a trace file can be searched.

Choose between the trace dump (*.trace) and trace file (*.trace) file formats from a dropdown list. Then, a suitable file can be selected from the file list and the dialog can be finished by means of <Open>.

The graph is then displayed according to the trace configuration in the trace editor as it is saved in the file.

3.15.15 Save Trace...

This command is used to save the current traces in a file.

Clicking it opens the Save trace dialog (default dialog for saving files). There, determine a file name and a storage location and choose between the trace file (*.trace) and text file (*.txt) file format in the dropdown list. Then, the dialog is to be finished by means of Save.



Menu Items

Traces that are saved as **trace file** (TRACE format: *.trace) contain the configuration settings with the recorded values which can then be loaded back into the trace editor. There, carry out an offline analysis of the trace values.

The configuration can be loaded back to the editor from a trace file with the command [Load trace...., page](#) .

Traces that are saved as **text file** (TXT format: *.txt) do not contain any configuration data; all recorded values are, however, listed. This file can be loaded and processed by tools supporting the CSV format. The trace editor itself cannot read this format. Loading back into the editor is not possible.

3.16 Visualization - Commands

3.16.1 Visualization - Commands, General Information

The commands for working in the [visualization editor, page 445](#), are described here.

They can be found in the main menu **VI Logic Visualization** and in the context menu.

If required, the menu structure can be reconfigured via the **IndraWorks ▶ Tools ▶ Customize ▶ Commands ▶ Visualization commands**.

Commands:

- [Add visualization element, page 301](#)
- [Frame selection, page 302](#)
- [Interface editor, page 303](#)
- [Keyboard operation, page 303,](#)
- [Element list, page 304,](#)
- [Background, page 304.](#)

Selection:

- [Select All, page 301](#)
- [Deselect All, page 301](#)
- [Group, page 301](#)
- [Ungroup, page 301](#)

Alignment:

- [Align left, page 305](#)
- [Align at top, page 305](#)
- [Align right, page 305](#)
- [Align at bottom, page 306](#)
- [Center horizontally, page 306](#)
- [Center vertically, page 306](#)

Order:

- [One level up, page 305](#)
- [Place on top, page 304](#)
- [Send to back, page 305](#)
- [One level down, page 305](#)
- [Enable keyboard operation, page 306-](#)

3.16.2 Add Visualization Element

This command is used to add a visualization element in the visualization editor.

This corresponds to add items from the [ToolBox window, page 446](#) via drag&drop.

The command opens a submenu in which the desired element can be selected from the currently available elements. The element is then inserted into the upper left corner of the editor window.

3.16.3 Select All

This command can be used to select all of the visualization elements in the visualization that is currently being edited in the editor.

Position the mouse in the visualization editor and in the context menu select **Select All**.

Also refer to

- [Deselect All, page 301](#).

3.16.4 Deselect All

Use this command to suspend the current selection of visualization elements.

Also refer to

- ["Select All, page 301"](#).

3.16.5 Group

Icon:



This command is used to group the currently selected visualization elements and to display the group as a single selected object.

To select multiple items, press and hold the <Shift> key while clicking on the desired elements. Alternatively, click in the editor window outside of an element and hold down the mouse key while drawing a rectangle around the desired elements.

To suspend the group, use the command [Ungroup, page 301](#).

The following figure shows grouping (from left to right) and ungrouping (from right to left) for two rectangle elements:

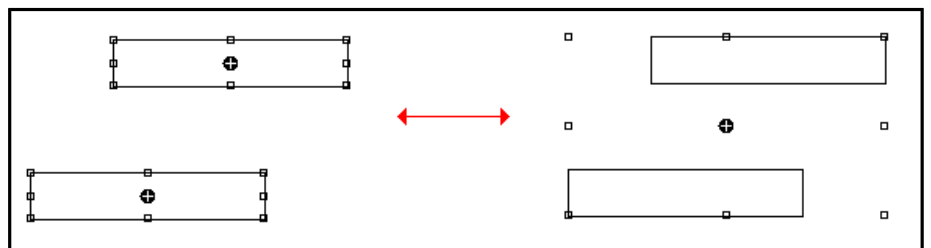


Fig.3-277: Group and Ungroup

3.16.6 Ungroup

Icon:



This command is used to suspend a group of visualization elements.

The individual elements are displayed as individually selected.

A group can be formed with the [Group, page 301](#), command.

Menu Items

3.16.7 Frame Selection

This command can be used to configure "frame" elements.

A "frame" element is used to define a sub-unit of a visualization. This sub-unit contains one or more visualizations, of which one is always displayed in online mode. Switch back and forth among the available visualizations.

The first visualization in the list of those assigned to the frame is displayed. One of the other visualizations in the frame can be displayed if the user makes an entry on another visualization element that is appropriately configured. This allows to switch back and forth among a variety of displays within one visualization, which contrasts with IndraLogic 1.x, where users had to make a complete jump from one visualization to another.



Further information about "Toggle frame visualizations" and "Input properties" can be found in the visualization editor; see [Visualization elements - Properties, page 451](#).

The visualizations in a frame are references, i. e. instances of the original visualizations, and placeholders defined in the original visualization can be replaced in the frame by local, suitable values.

See also [Frames, References, Interfaces, Placeholders, page 628](#).

The selection of the visualizations in a frame is made in the "Frame configuration" dialog.

To do this, highlight the frame element and select "Frame selection" in the context menu.

The "Configuring the frame visualization" opens; see [Dialog - Configuring the frame visualization, page 302](#).

Alternatively, open the dialog "Configuring the frame visualization" in the main menu using **VI Logic Visualization ► Frame Selection**.

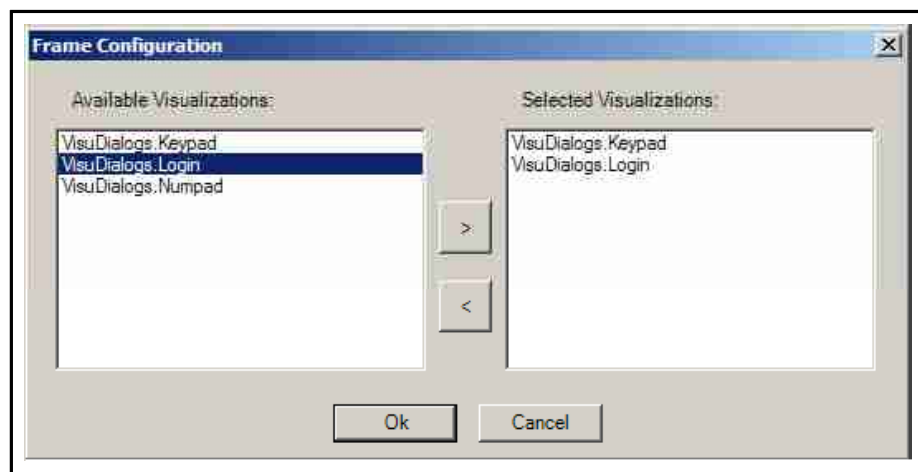


Fig.3-278: Dialog - Configuring the frame visualization

On the left side are the "Available Visualizations" in the project. Select those that are to be referenced in the frame. To do this, click on the arrow buttons to insert or remove visualizations into or from the "Selected visualizations" list.



Menu Items



It is recommended to assign only visualizations from the **global** pool to a frame.

Otherwise problems could occur if a device or application object is renamed later and the path of the assigned visualization is no longer valid.

The order of the selected visualizations from top to bottom determines the automatically generated index numbers for the visualizations. The topmost is assigned "0" and the following "1", "2", etc.

The index numbers are required for configuring the toggle function ([Toggle frame visualization, page 451](#)) for another element.

The visualization is initially displayed with index "0".

Example:

Visualization with button bar and display field:

Configure a visualization that contains a button bar and a display field, which displays another visualization in online mode, depending on which button is clicked. To do this, carry out the following steps:

1. Create a visualization "visu_xy" and three other visualizations "visu1", "visu2", "visu3", which are to be displayed later in "visu_xy".
2. Insert three rectangles (buttons) and a frame into "visu_xy". Configure the frame element with the "Frame selection" function and assign it visualizations "visu1", "visu2" and "visu3".
3. Configure the buttons such that when they are clicked, "visu1", "visu2" or "visu3" is called in the frame ([input configuration, page 451](#), OnMouseClicked property, toggle frame visualization).



Detailed information about configuring buttons can be found in the visualization editor, see [Visualization elements - Properties, page 451](#).

3.16.8 Interface Editor

Icon:

Default shortcut: <Alt>+<F6>

The interface editor is used to define "placeholder variables" in a visualization, which are then inserted into another visualization in the frame element, i. e. they are referenced.

In the main menu click on **VI Logic Visualization ► Interface editor** to open the interface editor.

A detailed description of the [interface editor, page 491](#) can be found in the editors.

3.16.9 Keyboard Operation

Icon:

This command opens the [configuration editor for the keyboard operation, page 494](#), for the current visualization. It is displayed in a tab view in the upper part of the visualization editor.

Menu Items

3.16.10 Element List

Icon:

This command opens the [element list editor, page 496](#), for the current visualization. It is displayed in a tab view in the upper part of the visualization editor.

3.16.11 Background

Icon:

A color and an image for the visualization background can be selected in the "Background Information" dialog.

In the main menu click on **VI Logic Visualization ► Background information** to open the "Background Information" dialog. Alternatively, open the "Background" dialog by positioning the mouse in the working area of the visualization editor and selecting **Background information** in the context menu.

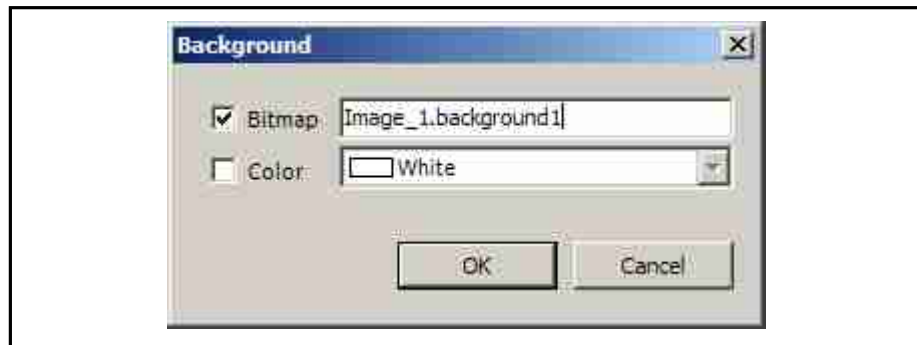


Fig.3-279: Background dialog

Select the desired options:

- **Bitmap:**
To define a background image, enter the path of an image file available in an image pool in the project here.
In addition, enter the name of the image pool and the ID of the image file - separated by a dot ".": - On:
<ImagePool> . <ID>
(e.g.. Image_1 . background1, Images_1 . 43).
- **Graph color:**
To define the background color of the visualization, select the desired color from the color selection list.

3.16.12 Order

Place on Top

Icon:

Use this command to position the [selected element, page 449](#), in the foreground of the visualization, i.e. on the "top" level. Elements on lower levels are covered by those on higher levels.

Highlight the visualization elements to be placed in the foreground and in the context menu click on **Order ► Place on top**. Alternatively, click in the main menu on **Order ► Place on top**.

One Level up

Menu Items

Icon:

Use this command to position the [selected element, page 449](#), one level higher, i.e. closer to the foreground in the visualization. Elements on lower levels are covered by those on higher levels.

Highlight the visualization elements to be placed one level further in the foreground and in the context menu click on **Order ▶ One level up**. Alternatively, click in the main menu on **Order ▶ One level up**.

Send to Back

Icon:

Use this command to position the [selected element, page 449](#), in the background of the visualization, i.e. on the "deepest" level. Elements on lower levels are covered by those on higher levels.

Highlight the visualization elements to be placed in the background and in the context menu click on **Order ▶ Send to Back**. Alternatively, click in the main menu on **Order ▶ Send to Back**.

One Level down

Icon:

Use this command to position the [selected element, page 449](#), one level lower in the visualization, i.e. closer to the background. Elements on lower levels are covered by those on higher levels.

Highlight the visualization elements to be placed one level further in the background and in the context menu click on **Order ▶ One level down**. Alternatively, click in the main menu on **Order ▶ One level down**.

3.16.13 Alignment

Align Left

Icon:

This command is used to align all of the [selected visualization elements, page 449](#), at the left edge of the element positioned farthest to the left.

Highlight the visualization elements to be aligned at the left and in the context menu click on **Direction ▶ Align left**. Alternatively, click in the main menu on **Direction ▶ Align left**.

Align at Top

Icon:

This command is used to align all of the [selected visualization elements, page 449](#), at the upper edge of the element positioned farthest toward the top.

Highlight the visualization elements to be aligned at the top and in the context menu click on **Direction ▶ Align top**. Alternatively, click in the main menu on **Direction ▶ Align top**.

Align Right

Icon:

This command is used to align all of the [selected visualization elements, page 449](#), at the right edge of the element positioned farthest to the right.



Menu Items

Highlight the visualization elements to be aligned at the right and in the context menu click on **Direction** ▶ **Align right**. Alternatively, click in the main menu on **Direction** ▶ **Align right**.

Align at Bottom

Icon:

This command is used to align all of the [selected visualization elements, page 449](#), at the lower edge of the element positioned farthest toward the bottom.

Highlight the visualization elements to be aligned at the bottom and in the context menu click on **Direction** ▶ **Align bottom**. Alternatively, click in the main menu on **Direction** ▶ **Align bottom**.

Center Horizontally

Icon:

This command is used to align all of the [selected visualization elements, page 449](#), based on their common middle point along the horizontal axis.

Highlight the visualization elements to be aligned at the horizontal center and in the context menu click on **Direction** ▶ **Center horizontally**. Alternatively, click in the main menu on **Direction** ▶ **Center horizontally**.

Center Vertically

Icon:

This command is used to align all of the [selected visualization elements, page 449](#), based on their common middle point along the vertical axis.

Highlight the visualization elements to be aligned at the vertical center and in the context menu click on **Direction** ▶ **Center vertically**. Alternatively, click in the main menu on **Direction** ▶ **Center vertically**.

3.16.14 Enable Keyboard Operation

Icon:

This command is in the provided in menu bar for an integrated visualization. It activates or deactivates [keyboard operation in online mode, page 657](#), of a visualization.

When keyboard operation is activated, input to elements and element selection can be performed using specific shortcuts. In this case, other commands given using the keyboard are not executed as long as the visualization editor is active and in online mode.

3.17 Symbol Configuration

The symbol configuration is used to create **symbols** with certain access rights that can be used from external locations to access project (application) variables, e. g. from an OPC server.

The description of the symbols (symbol information) is made available in an xml file (symbol file) of the project directory and simultaneously loaded to the control with the application.

To do this, a **symbol list** is generated that is exported into an XML file in the project directory and, during the application download, loaded into a file on the target system that is not visible to users.

The XML file is named according to the following syntax:

```
<ProjectName>.<DeviceName>.<ApplicationName>.xml
```

Menu Items

(in the following example: IndraLogic.IndraMotionMlp1.Application.xml).

Creating a symbol configuration

A "**Symbol configuration**" object is included below the respective **application** in the device window in the project.

The variables that are to be exported as symbols, can be defined either in the **symbol configuration editor**, page 398, or by using **Pragmas {attribute 'linkal-ways'}**, page 536, which are added to the variable declarations.

The SFC editor provides another possibility: Here, in the **element properties**, page 414, specify implicitly generated element flags that can be exported to the symbol configuration.

The **symbol name** is generated in the symbol configuration according to the following syntax:

```
<ApplicationName>.<POUName>.<VariableName>
```

in the following example: "Application.MOTIONPROG.en_PowerRA1" .



When accessing the variable, the complete symbol name always has to be entered using this spelling.

Symbol information

The symbols defined for an application are exported into an xml file in the project directory (**symbol file**).

This file is name according to the following syntax:

```
IndraLogic.<DeviceName>.<ApplicationName>.xml,
```

(in the following example: IndraLogic.DCC_Control.Application.xml).

The information is **loaded** together with the application onto the control.

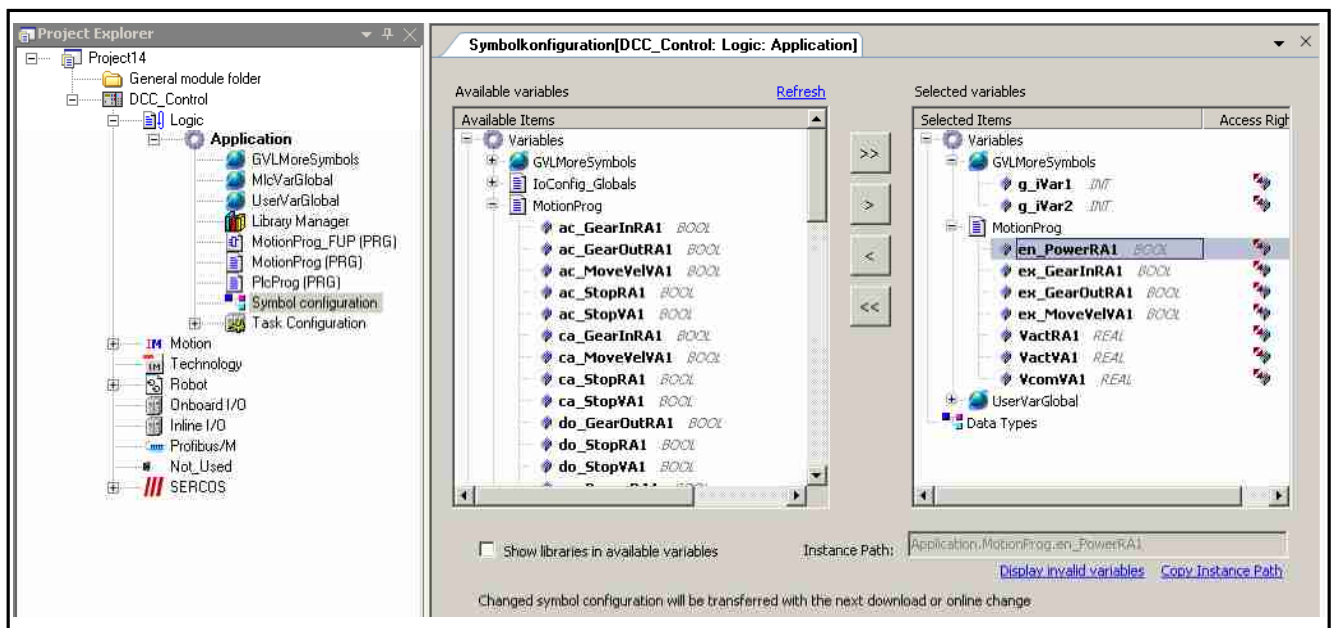


Fig.3-280: Example symbol configuration

If the symbol configuration is modified in online status, it can be explicitly re-loaded; see the **Download**, page 398, button in the editor window.



RODAVIGO, S.A.
RODAMIENTOS VIGO, S.A.

www.rodavigo.net

+34 986 288118
Servicio de Att. al Cliente

308/697

Bosch Rexroth AG

DOK-IWORKS-IL2GPRO*V12-AP01-EN-P
Rexroth IndraWorks 12VRS IndraLogic 2G PLC Programming System