Product Manual
Application Unit Logic

1. General

In modern buildings with KNX, logical and complex operations are becoming more important, e.g. logic gates, gates, timing modules and comparators which are required in the building system technology can be defined and logically linked to each other.

With these basic elements it is possible to implement logic control, logic operations, interlocks, fault signals, telegram multiplication and a range of further functions which are required in daily practice. Further applications, e.g. memory or sequence control can also be implemented using a combination of logical elements.

Working with logic functions differs in several points from working with a programmable logic controller (PLC) or a hard-wired programmed controller.

The applications described beforehand can be parameterised using the Application Unit Logic.

This manual provides you with detailed technical information relating to the Application Unit Logic, installation, programming and explains the Application Unit Logic using examples.

This manual is divided into the following sections:

•	Chapter 1	General
•	Chapter 2	Device technology
•	Chapter 3	Commissioning
•	Chapter 4	Planning and application
•		Appendix

1.1. Product and functional overview

The Application Unit Logic ABL/S 2.1 is a rail mounted device for insertion in the distribution board. The connection to the bus is implemented via a bus connection terminal on the front of the device. The assignment of the physical address as well as the parameter settings is carried out with ETS3 from version V1.0 onwards.

The device is powered via KNX and does not require an additional power supply.

The processing is implemented in the application program ***Logic Time 254 EA/2*.**

The device features comprehensive and clearly arranged functionality and permits usage in the most differing fields of application.

The following list provides and overview:

- 50 logical functions
  - AND
  - OR
  - One hot)
- 50 unidirectional and bidirectional gates
- 30 timers with
  - ON delay
  - OFF delay
  - Pulse duration
  - Staircase lighting function
- 10 comparators

2. Device technology

Fig. 1: ABL/S 2.1

The Application Unit Logic is a rail mounted device for insertion in the distribution board. The device contains logical function with the possibility of individual parameter parameterisation. Furthermore, timers with on and off delays and pulse duration, staircase lighting functions and comparators are available.

The device is ready for operation after connection to the bus voltage. The Application Logic Unit is parameterised via the ETS3. The connection to the bus is established using the front side bus connection terminal.

2.1. Technical data

Power supply	
- Bus voltage	21 … 32 V DC
- Current consumption, bus	< 12 mA
- Leakage loss, bus	Max. 250 mW
Connections	
- KNX	via bus connection terminals
Operating and display elements	
- Programming LED	for assignment of the physical address
- Programming button	for assignment of the physical address
Enclosure	
- IP 20	to DIN EN 60 529
Safety class	
- II	to DIN EN 61 140
Isolation category	
Overvoltage category	III to DIN EN 60 664-1
Pollution degree	2 to DIN EN 60 664-1
KNX safety extra low voltage	
SELV 24 V DC	
Temperature range	
- Operation	- 5°C...+ 45°C
- Storage	-25°C...+ 55°C
- Transport	-25°C...+ 70°C
Ambient conditions	
- Maximum air humidity	93 %, no condensation allowed
Design	
- Modular installation device (MDRC)	Modular installation device, Pro*M*
- Dimensions	90 x 36 x 64.5 mm (H x W x D)
- Mounting width in space units	2, 2 modules at 18 mm
- Mounting depth	64.5 mm
Installation	
- On 35 mm mounting rail	to DIN EN 60 715
Mounting position	
- as required	
Weight	
- 0.1 kg	
Housing/colour	
- Plastic housing, grey	
Approvals	
- KNX to EN 50 090-1, -2	Certification
CE mark	
- in accordance with the EMC guideline and low voltage guideline	

Application program	**Logic Time 254EA/2**
Max. number of communication objects	254
Max. number of group addresses	254
Max. number of associations	254

Note: ETS3 is required for programming. A ".VD3" type file must be imported. The application program is available in the ETS3 at ABB/controller/controller.

2.2. Circuit diagram

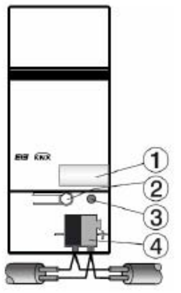


Fig. 2: Circuit diagram

1 Label carrier
2 Programming button
3 Programming LED
4 Bus connection terminal

2.3. Dimension drawing

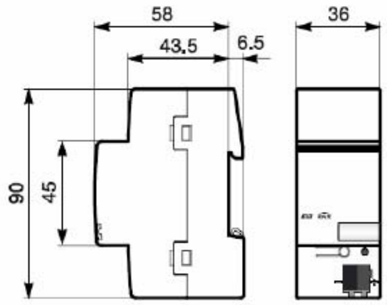


Fig. 3: Dimension drawing

2.4. Assembly and installation

The Application Unit Logic is a modular installation device for fast installation in the distribution board on 35 mm mounting rails to DIN EN 60 715.

The connection to the bus is implemented using the supplied bus connection terminal.

The device is ready for operation after connection to the bus voltage.

Accessibility of the devices for the purpose of operation, testing, visual inspection, maintenance and repair must be must be provided (conform to DIN VDE 0100-520).

Commissioning requirements

In order to commission the Application Unit Logic, a PC with ETS3 and a connection to KNX, e.g. via an RS232 interface or a USB interface is necessary.
The device is ready for operation after connection to the bus voltage.

The installation and commissioning may only be carried out by electrical specialists. The appropriate standards, guidelines, regulations and specifications should be observed when planning and setting up electrical installations.

- The device should be protected from damp, dirt and damage during transport, storage and operation.
- The device should not be operated outside the specified technical data!
- The device should only be operated in a closed housing (distribution board)!

Supplied state

The Application Unit Logic is supplied with the physical address 15.15.255. The application program is pre-installed. Hence, only group addresses and parameters must be loaded during commissioning. The entire application can be reloaded if required. A longer downtime may result if the application program is changed or after a discharge.

Download behaviour

Due to the complexity of the device the download bar can take up to 1.5 minutes to appear and depends on the PC which is used.

Assignment of the physical address

The assignment and programming of the physical address is carried out in the ETS.

Cleaning

If devices become dirty, they can be cleaned using a dry cloth. Should a dry cloth not remove the dirt, they can be cleaned using a slightly damp cloth and soap solution. Corrosive materials or solutions should never be used.

Maintenance

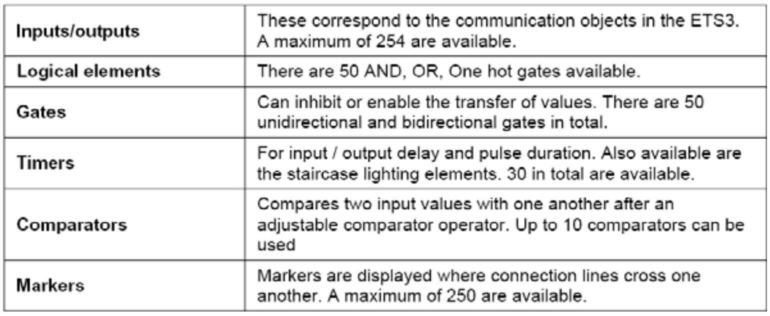
The device is maintenance-free. No repairs should be carried out by unauthorised personnel if damage occurs (e.g. during transport or storage). The warranty expires if the device is opened.

3. Commissioning

3.1. Overview

For the Application Unit Logic the application program ***"Logic Time 254 EA/2"*** is available. Programming requires the ETS3. A maximum of 254 communication objects, 254 group addresses and 254 associations can be linked.

The following functions are available:



Inputs/outputs	These correspond to the communication objects in the ETS3. A maximum of 254 are available.
Logical elements	There are 50 AND, OR, One hot gates available.
Gates	Can inhibit or enable the transfer of values. There are 50 unidirectional and bidirectional gates in total.
Timers	For input / output delay and pulse duration. Also available are the staircase lighting elements. 30 in total are available.
Comparators	Compares two input values with one another after an adjustable comparator operator. Up to 10 comparators can be used
Markers	Markers are displayed where connection lines cross one another. A maximum of 250 are available.

Table 1: Functions of the application program

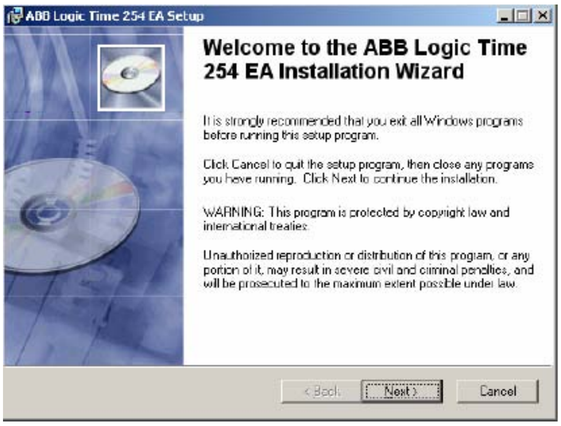
3.2. Import of the application program in the ETS3

The Application Unit Logic is imported via the *Import* function into the ETS3. Thereafter the product database is automatically imported and the project data is updated.

Note: The application program for the Application Unit Logic is installed on the computer.

Add the Application Unit Logic in the ETS3 via the function *Add device*.

Thereafter the dialog *Logic Time 254 EA Setup – Welcome to…* becomes visible.



Welcome to the ABB Logic Time 254 EA Installation Wizard

It is strongly recommended that you exit all Windows programs before running this setup program.

Click Cancel to quit the setup program, then close any programs you have running. Click Next to continue the installation.

WARNING: This program is protected by copyright law and international treaties.

Unauthorized reproduction or distribution of this program, or any portion of it, may result in severe civil and criminal penalties, and will be prosecuted to the maximum extent possible under law.

Fig. 4: Import of the application program Logic Time 254 EA Part 1

You access the following dialog via *Next* >. This accepts and acknowledges all instructions.

< Back

Is not active as there was not a previous installation page.

Next >

If you click on the *Next* > button, the next installation page is accessed. All of the information on the screen is acknowledged.

Cancel

If you click on the *Cancel* button the installation process is ended.

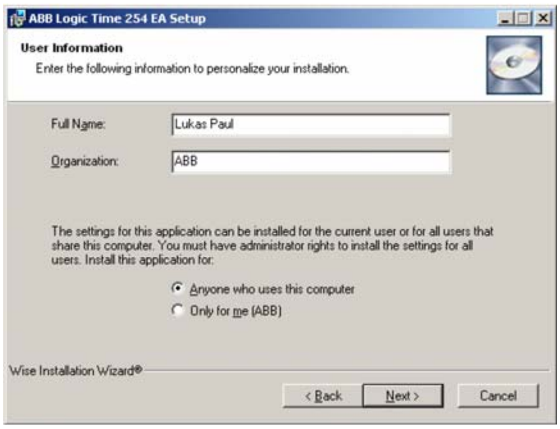


ABB Logic Time 254 EA Setup

User Information

Enter the following information to personalize your installation.

Full Name: Lukas Paul

Organization: ABB

The settings for this application can be installed for the current user or for all users that share this computer. You must have administrator rights to install the settings for all users. Install this application for:

● Anyone who uses this computer

○ Only for me (ABB)

Fig. 5: Import of the application program Logic Time 254 EA Part 2

In the *User Information* dialog the general user information is requested.

Full Name

Enter the user name here.

Organization

Enter the organization/company name here.

Anyone who uses this computer

By choosing *Anyone who uses this computer* all users of the computer are granted the right to use the application program.

Only for me (Full Name)

By choosing *Only for me (Full Name)* only the entered user on the PC is allowed to use the application program.

< Back

If you click on the <*Back* button the next installation page is accessed.

Next >

If you click on the *Next* > button the next installation page is accessed. All of the information on the screen is acknowledged.

Cancel

If you click on the *Cancel* button the installation process is ended.

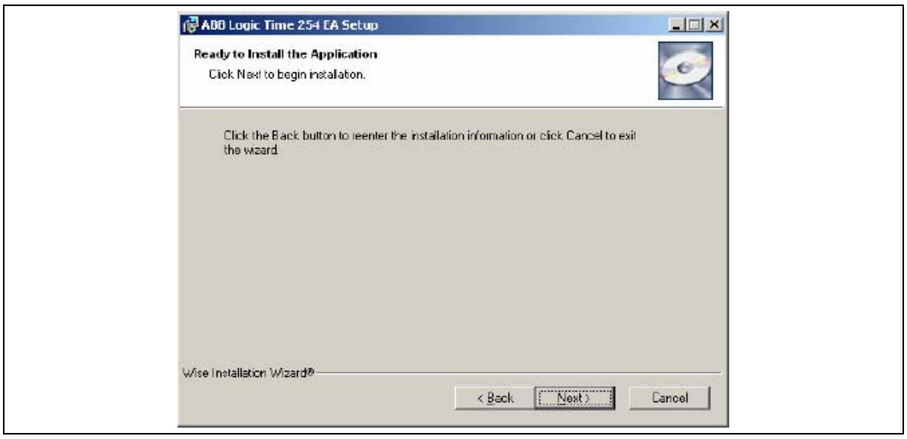


ABB Logic Time 254 EA Setup

Ready to Install the Application

Click Next to begin installation.

Click the Back button to reenter the installation information or click Cancel to exit the wizard.

Wise Installation Wizard®

Fig. 6: Import of the application program Logic Time 254 EA Part 3

In the dialog *Ready to Install the Application* the next dialog is started after the *Next* > button is pressed.

< Back

If you click on the <*Back* button the next installation page is accessed.

Next >

If you click on the *Next* > button the next installation page is accessed. All of the information on the screen is acknowledged.

Cancel

If you click on the *Cancel* button the installation process is ended.

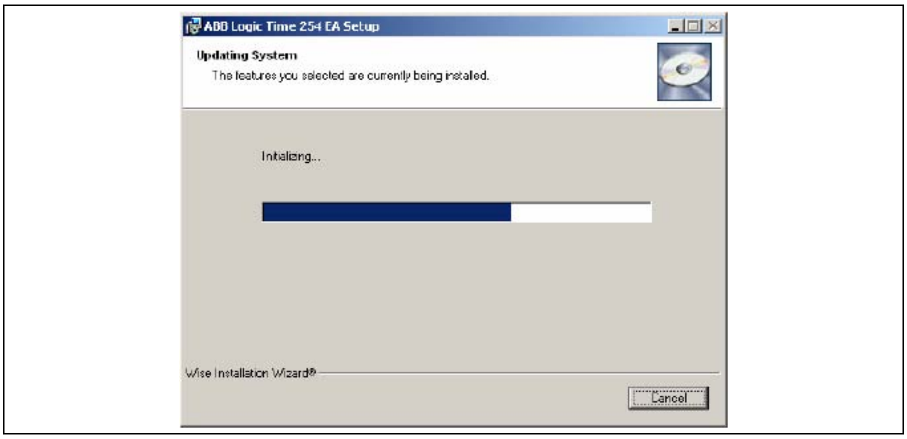


ABB Logic Time 254 EA Setup

Updating System

The features you selected are currently being installed.

Initializing...

Wise Installation Wizard®

Fig. 7: Import of the application program Logic Time 254 EA Part 4

The application is installed in the *Updating System* dialog.

Cancel

If you click on the *Cancel* button the installation process is ended.

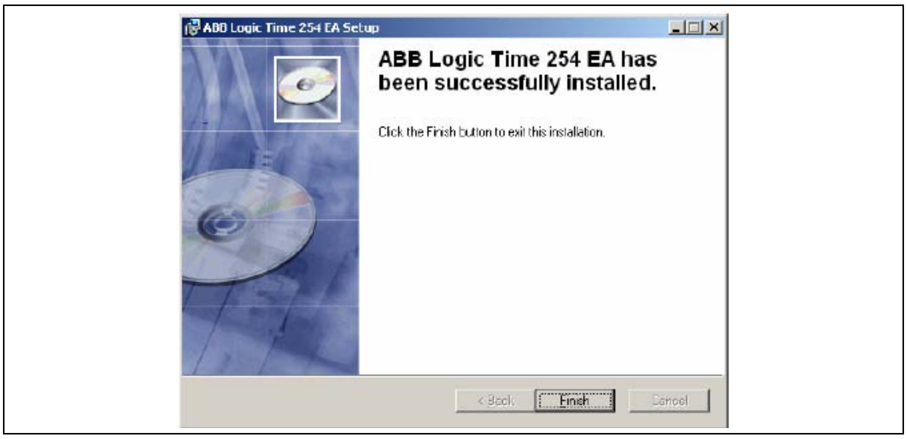


ABB Logic Time 254 EA has been successfully installed.

Click the Finish button to exit this installation.

Fig. 8: Import of the application program Logic Time 254 EA Part 5

The last dialog *Logic Time Plugin…* confirms successful installation.

Finish

By pressing on the *Finish* button the installation of the application program is ended and you return automatically to ETS3.

3.3. Verifying the installation outside of the ETS3

ABB_LT02

In the directory C:\Program Files\Common Files\EIBA sc\Baggage\2\ the *ABB_LT02* folder must exist.

Note: If the *ABB_LTO2* folder is deleted for some reason, it is automatically reinstalled the next time the product is accessed in the ETS3.

Logic Time Plugin

At *Start* > *Settings* > *Control Panel* > *Software* the *Logic Time 254 EA* must exist.

Note: If the *Logic Time Plugin* folder is deleted for some reason, it is automatically reinstalled the next time the product is accessed in the ETS3.

3.4. User interface

The user interface of the application program *Logic Time 254EA/2* is activated in the following manner:

- Select the device with the right mouse button of the *Edit parameter…* context menu.
- Double click on the device and then on the *Parameter parameter*.

3.4.1.Logic Time 254EA/2

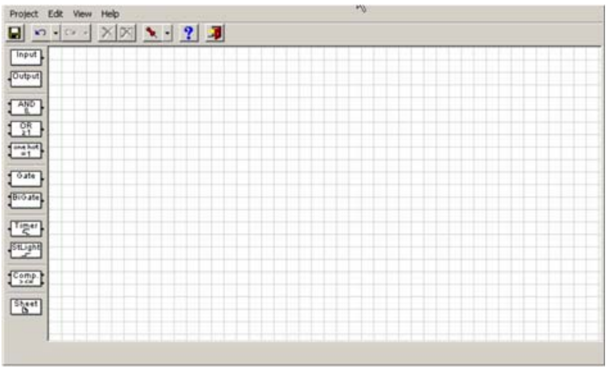


Fig. 9: “Logic Time 254 IO/2” user interface

3.4.2.Title bar

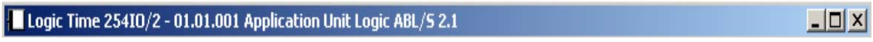

Logic Time 254IO/2 - 01.01.001 Application Unit Logic ABL/S 2.1

Fig. 10: “Title bar” user interface

On the title bar the name of the application program, the physical address, the type and the name or the device or the description of the user appear.

3.4.3.Menu bar

Fig. 11: “Menu bar” user interface

In the menu bar the selection menu Project, Edit, View and Help are contained.

3.4.4.Project selection menu

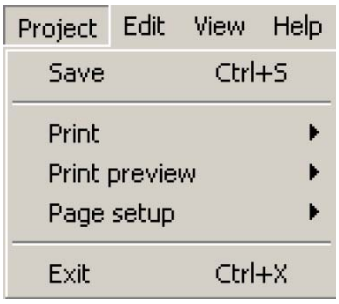


Fig. 12: “Project selection menu” user interface

Some menu points are context dependent and are only used for certain configurations. Only activated menu points are displayed with a grey background.

3.4.4.1. Save

By selection of the *Save* menu point the parameter data is saved in the ETS3 database.

Hotkey: Ctrl+S

3.4.4.2. Print

By selection of the *Print* menu point parameter settings can be printed in graphic and text form.

Note: The complete project data is printed out in text form

3.4.4.3. Print preview

By selection of the *Print preview* menu point the parameterisation which is to be printed is displayed in graphic or text format on the monitor.

3.4.4.4. Page setup

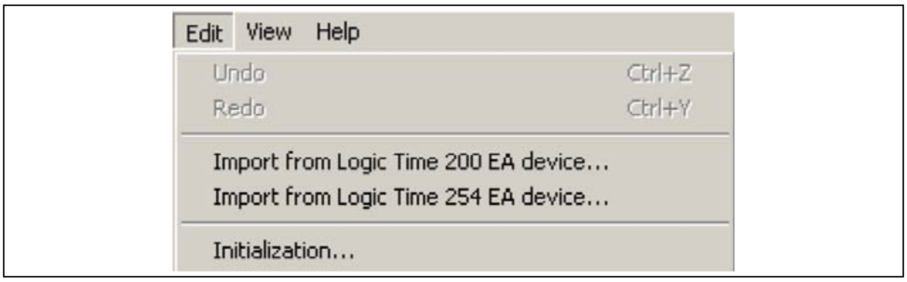
By selection on the *Page setup* menu point the page layout options can be set.

3.4.4.5. Exit

By selection of the *Exit* menu point you can close the application program and return to the *Properties* dialog window of the ETS3. You will be requested to save the data if necessary.

Hotkey: Ctrl+X

3.4.5.Edit selection menu



Edit
View
Help
Undo Ctrl+Z
Redo Ctrl+Y
Import from Logic Time 200 EA device...
Import from Logic Time 254 EA device...
Initialization...

Fig. 13: “Edit selection menu” user interface

Some menu points are context dependent and are only used for certain configurations. Only activated menu points are displayed with a grey background.

3.4.5.1. Undo

By selection of the menu point *Undo* the last action carried out will be undone.

Hotkey: Ctrl+Z

3.4.5.2. Redo

By selection of the menu point *Redo* the last action carried out will be deleted.

Hotkey: Ctrl+Y

3.4.5.3. Import from Logic Time 200EA/1.4 device

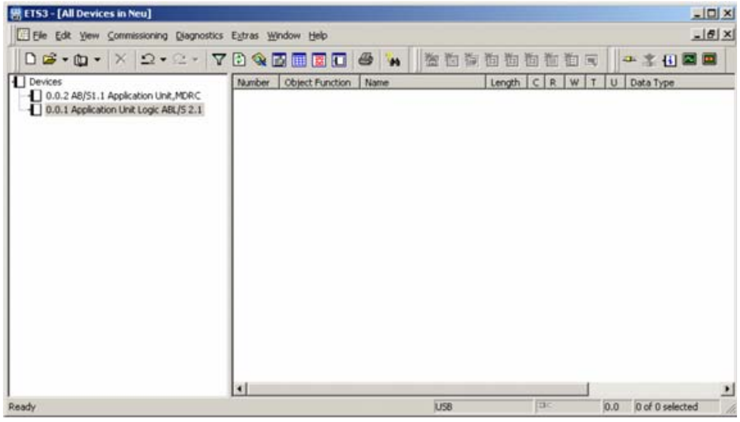

ETS3 - [All Devices in Neu]
File Edit View Commissioning Diagnostics Extras Window Help
Devices
0.0.2 AB/S1.1 Application Unit,MDRC
0.0.1 Application Unit Logic ABL/S 2.1
Number Object Function Name Length C R W T U Data Type
Ready
USB
0.0
0 of 0 selected

Fig. 14: User interface “AB/S 1.1 and ABL/S 2.1”

In order to import an application program *Logic Time 200EA/1.4* into the application logic module, it must be available in the ETS3 in the same project as the application module.

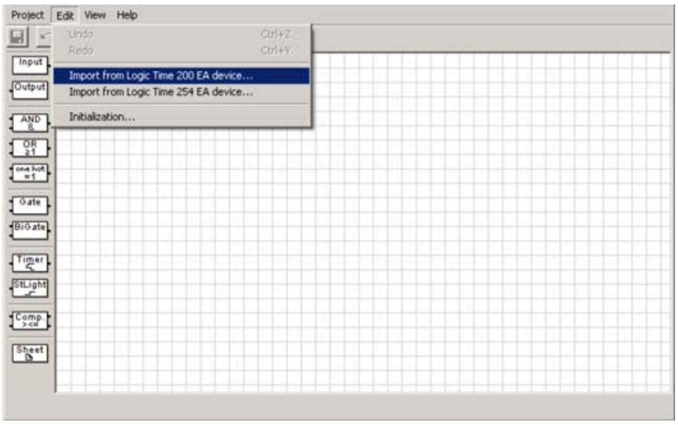
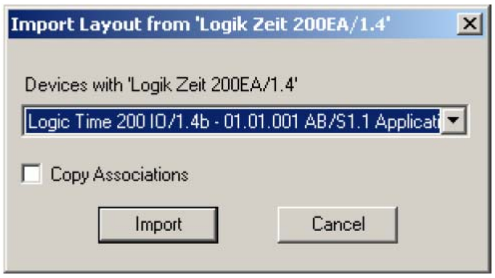


Fig. 15: User interface “Import from Logic Time 200 EA/1.4” device

Thereafter open the ABL/S and under Edit, add the *Import from Logic Time 200 EA device.*


Import Layout from 'Logik Zeit 200EA/1.4'

Devices with 'Logik Zeit 200EA/1.4'

Logic Time 200 IO/1.4b - 01.01.001 AB/S1.1 Applicat

□ Copy Associations

Fig. 16: User interface “Import Logic Time 200 EA/1.4 configuration”

Device with “Logic Time 200 EA/1.4”

Only select the application program to be imported from the selection list.

Copy Associations

If the checkbox is set under *Copy Associations*, the entire information e.g. group addresses of the application program is copied into the new application program.

Import buttons

The settings are copied and the parameter window closes via the button.

Cancel button

The function is interrupted and the parameter window closes via the button.

3.4.6.Import from an older application program

An older application program than the *Logic Time 200EA/1.4* must be processed first in the application module AB/S 1.1 with the application program *Logic Time 200EA/1.4.*

Note: The ETS3 is required.

Procedure:

An application program with the *Logic Time 200EA V1.1* should be converted to the *Logic Time 254EA.*

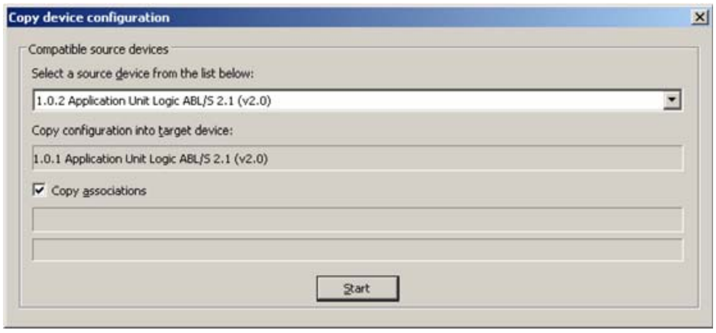
1. In the ETS2:

- Version *Logic Time 200EA V1.1* converts to *Logic Time 200EA V1.3.*
- Export the project and import it into the ETS3.

2. In the ETS3:

- Version *Logic Time 200EA V1.3* converts to *Logic Time 200EA V1.4*.
- Thereafter Version *Logic Time 200EA V1.4* converts to *Logic Time 254EA.*

3.4.6.1. Import from Logic Time 254EA device



Select a source device from the list below:
1.0.2 Application Unit Logic ABL/S 2.1 (v2.0)

Copy configuration into target device:
1.0.1 Application Unit Logic ABL/S 2.1 (v2.0)

☑ Copy associations

Fig. 17: User interface “Import from Logic Time 254 EA - successful”

By selection of the menu point Import from *Logic Time 254 EA device...* an existing application program which has been created in the application module can be imported into the current application module logic.

Select a source device from the list below:

The data is copied to the target device from this source device.

Copy configuration into target device:

This parameter is inactive. The target device is shown here.

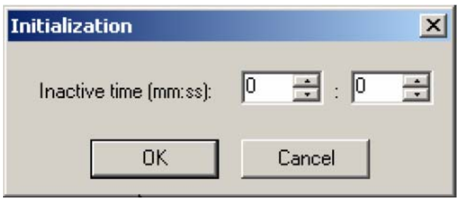
Copy associations

If the checkbox is set under *Copy associations*, the group addresses are also linked.

Start button

The settings are copied and the parameter window closes via the button.

3.4.6.2. Initialization



Initialization

Inactive time (mm:ss): 0 : 0

Fig. 18: User interface Initialization

Inactive time (mm:ss):

Options: 0…59 min
0…59 s

With this parameter the *Inactive time* is set. During this time no telegrams will be sent and received telegrams will not be evaluated. The display for minutes and seconds can be processed separately via the buttons *Up arrow* and *Down arrow*.

The following drawing indicates the sequence involved:

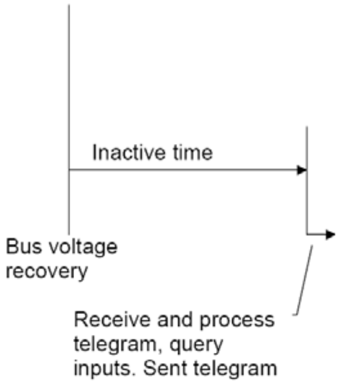


Fig. 19: Behaviour after voltage recovery

OK button

The settings are copied and the parameter window closes via the button.

Cancel button

The function is interrupted and the parameter window closes via the button.

3.4.7.View selection menu

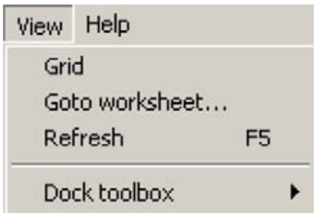


Fig. 20: User interface “View selection menu”

3.4.7.1. Grid

Options: activated = ticked
not activated = not ticked

By selection of the menu point *Grid* a grid is displayed on the user interface which simplifies alignment of the component elements.

3.4.7.2. Goto worksheet

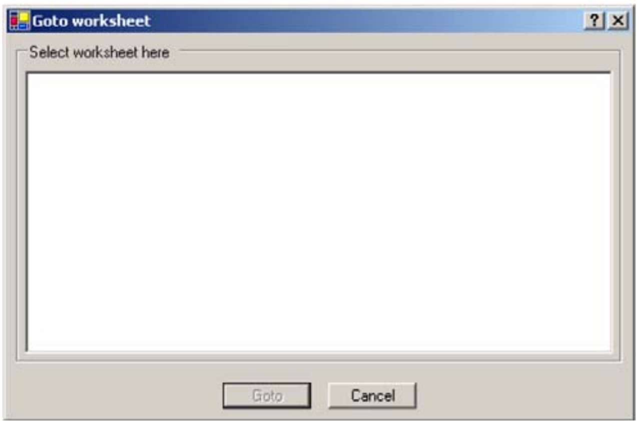


Fig. 21: User interface “Goto worksheet…”

By selection of the menu point *Goto worksheet...* a dialog window for selection of a worksheet opens.

More complex parameterisation and a smaller monitor size, can quickly lead to a very unclear view of the graphic parameterisation. Worksheet management is used in order to effectively design this. It describes a part of the graphical parameterisation and is implemented in the form of jump labels.

Goto button

The button is used to jump to the selected worksheet and the parameter window closes.

Cancel button

The function is interrupted and the parameter window closes via the button.

3.4.7.3. Refresh

By selection of the *Refresh* menu point a graphic representation can be reloaded and displayed.

Hotkey: F5

3.4.7.4. Dock toolbox

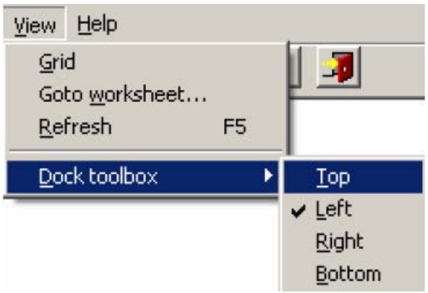


Fig. 22: User interface “Dock toolbox”

Options: activated = ticked
not activated = not ticked

By selection of the menu point *Dock toolbox on* the user can individually select if the toolbar is displayed at the top, left, right or bottom of the parameter window.

3.4.8.Help selection menu

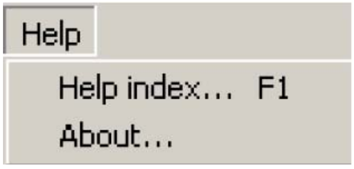


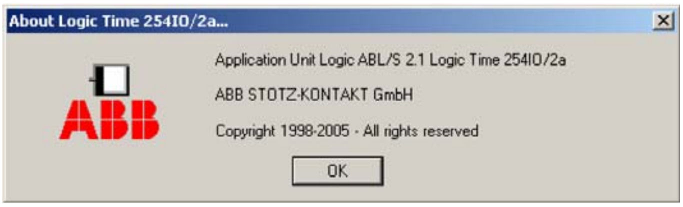
Fig. 23: User interface “Help selection menu”

3.4.8.1. Help index F1

By selection of the menu point *Help index...* the manual is displayed in PDF format.

Hotkey: F1

3.4.8.2. About



About Logic Time 25410/2a...

Application Unit Logic ABL/S 2.1 Logic Time 25410/2a

ABB STOTZ-KONTAKT GmbH

Copyright 1998-2005 - All rights reserved

Fig. 24: User interface “About…”

By selection of the menu point *About...* information concerning the application program and the manufacturer of the device are displayed.

OK button

The parameter window is closed via the button.

3.4.9.Toolbar

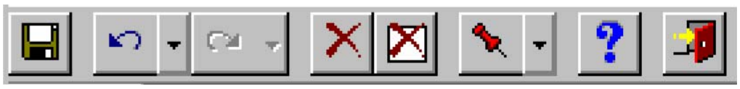


Fig. 25: User interface “Toolbar”

The toolbar contains buttons with the most important commands. The buttons are context dependent and are only used for certain configurations.

3.4.9.1. Save

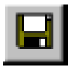


Fig. 26: Save icon

Saves the parameter data.

3.4.9.2. Undo last action

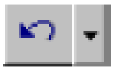


Fig. 27: Undo last action toolbar

The last action performed is undone.

3.4.9.3. Redo last action

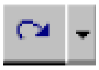


Fig. 28: Redo last action toolbar

The last action performed is restored.

3.4.9.4. Delete the selected element or link

Fig. 29: Delete the selected element or link icon

Deletes the selected objects or a selected link.

3.4.9.5. Deletes all objects and links

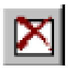


Fig. 30: Deletes all objects and links icon

Deletes all objects and links.

3.4.9.6. Define toolbar pin on position

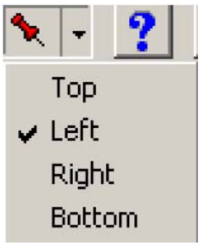


Fig. 31: Define pin on position of the toolbar icon

Options: Top/
Left/
Right/
Bottom
activated = checked
not activated = not checked

The pin on position of the toolbar can be set individually.

3.4.9.7. Display help themes

Fig. 32: Display help themes icon

Opens the Help for the application program. The manual is then displayed in PDF format.

Hotkey: F1

3.4.9.8. Exit

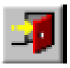


Fig. 33: Exit icon

Closes the application program. Requests when required to save the data.

3.4.9.9. Hotkey selection

• Ctrl + G	Print graphic
• Ctrl + S	Save
• Ctrl + T	Print text
• Ctrl + X	Exit
• Ctrl + Y	Redo
• Ctrl + Z	Undo
• Del	Delete
• F1	Help
• F5	Refresh

(Ctrl = control key)

3.4.10. Toolbar

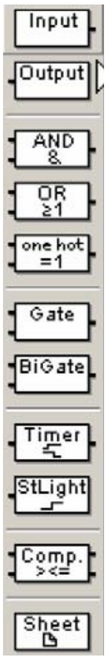


Fig. 34: Toolbar

The toolbar contains inputs, outputs, logical elements as well as the function worksheet.

3.4.10.1. Input

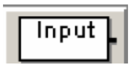


Fig. 35: Input

Corresponds with the communication objects in the ETS3 and can be linked to the logical elements. 254 communication objects are available for inputs and outputs. The communication object type can assume different forms.

3.4.10.2. Output

Fig. 36: Output

Corresponds with the communication objects in the ETS3 and can be linked to the logical elements. 254 communication objects are available for inputs and outputs. The communication object type can assume different forms.

3.4.10.3. AND gate

Fig. 37: AND gate

An AND gate only assumes the value 1 on the output if all inputs have the value 1.

3.4.10.4. OR gate

Fig. 38: OR gate

An OR gate output assumes the value 1 if at least one input has the value 1.

3.4.10.5. One hot gate

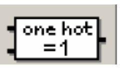


Fig. 39: One hot gate

An One hot gate output only assumes the value 1 if exactly one input has the value 1.

3.4.10.6. Gate

Fig. 40: Gate

A Gate has one input, one output and one control input. If an event occurs at the input and the gate is enabled, it can transfer the value at the input to the output. If the gate is disabled, there is no reaction.

3.4.10.7. Bidirectional gate

Fig. 41: Bidirectional gate

The Bidirectional gate features an input and an output, as well as a control input. The enabled gate can transfer a communication object value in both directions.

3.4.10.8. Timer

Fig. 42: Timer

A Timer has an input and an output. The input also has another filter. If an event occurs at the input, the timer can transfer the value at the input to the output with a time delay.

3.4.10.9. Staircaselight

Fig. 43: Staircaselight

The Staircaselight function consists of an input which is simultaneously an output. The input and output signal is reset after the adjustable time has timed out. If there is a new input signal within the time, the output signal remains at the existing value and the time is restarted.

3.4.10.10. Comparator

Fig. 44: Comparator

A Comparator comprises of two inputs, an output and an overflow output. It compares two input values with one another after an adjustable comparator operation. If the comparative condition is fulfilled the output sends a 1, if it is not fulfilled it sends a 0.

If the value range is exceeded during a comparison, a fault is indicated on the overflow output.

3.4.10.11. Worksheet

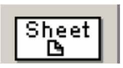


Fig. 45: Worksheet

A Worksheet is a text field. It describes a part of the graphical parameterisation and is implemented in the form of jump labels.

3.5. Working in the worksheet

The design of the logic functions is carried out in the worksheet and implemented in a graphic function chart based on DIN 40900.

The engineering design proceeds as follows:

- Place the input
- Implement the logic function
- Place the output
- Establish the connections
- Edit the parameters of the individual elements

3.5.1.Assigning new elements

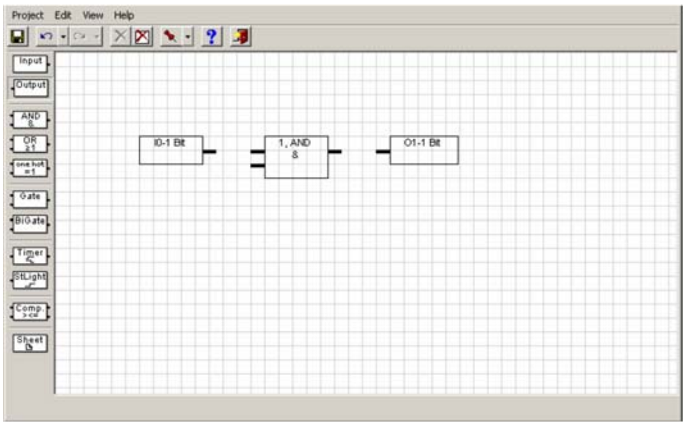


Fig. 46: Assigning new elements

With the help of the toolbar, on the left of the new elements, e.g. inputs, outputs and logical functions can be created. The corresponding element is selected with the left mouse button in the toolbar and placed with the pressed button on the worksheet. Several elements of the selected type can be added if the Ctrl key is held pressed during insertion.

Note: In order to align the elements the grid can be displayed on the View Menu.
Furthermore, the set print area should be observed when placing elements by calling up the *Print preview* and *Graphic preview.*

3.5.2.Selection of elements

The element can be selected in order to move or delete it. This is done by clicking on the element with the left mouse button. This is also the case when selecting a line of association.

3.5.3.Multiple selection of elements

Several elements can be selected at once, e.g. for moving or deleting. The elements are selected using the mouse. Press the left mouse button and drag out a rectangle, keeping the button pressed down. Or keep the Ctrl button pressed and click on each of the required elements with the left mouse button. Thus, several elements situated next to one another can be marked. The selected elements are then displayed in colour.

3.5.4.Moving elements in the worksheet

The visible area of the worksheet can be moved using the scrollbars on the right hand side and at the bottom of the window. Elements can also be moved to a position outside the visible area. Select the corresponding element with the left mouse button and move the mouse pointer in the required direction keeping the button pressed down. The visible area of the worksheet moves if the limits of the visible area are crossed over. If necessary the worksheet is enlarged.

Note: The set print area should be observed when placing elements by calling up the *Print preview* and *Graphic preview*.

3.5.5.Linking functions

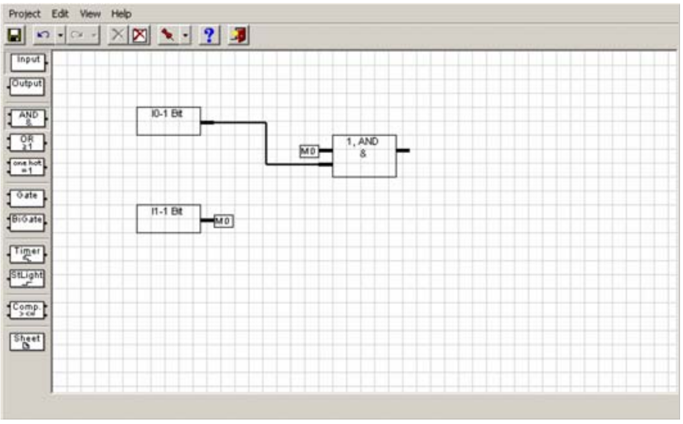


Fig. 47: Linking functions

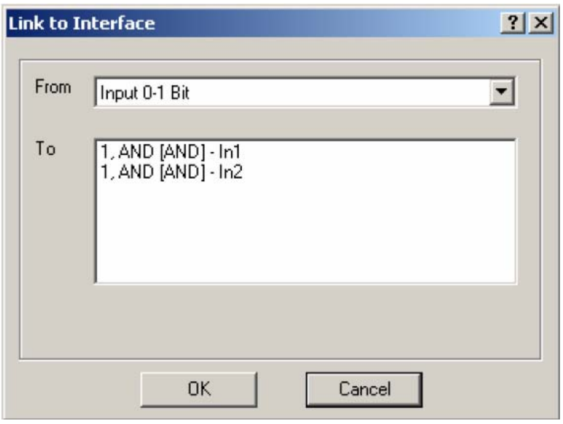
A connection is established when the starting point of the connection is selected with the left mouse button, the cursor is moved to the end point of the required connection while the button remains pressed and then the button on the mouse is released.

Links are shown as lines. If another line has to be crossed to represent a link, this is indicated by an auxiliary flag. The labelling of the flag is assigned automatically and cannot be changed.

Note: Only elements of the same type are connected to one another.

The parameter window *Link to Interface* becomes active with the following procedure:

- Select the element and open the context menu with the right mouse button. Then select the command *Link to Interface ….*



From: Input 0-1 Bit

To:
1, AND [AND] - In1
1, AND [AND] - In2

Fig. 48: Parameter window “Link to interface”

In order to simplify the creation of links for larger projects, e.g. if elements which are not all in the same visual area are to be linked, it is possible to link an element using the command *Link to Interface*.

OK button

The settings are copied and the parameter window closes via the button.

Cancel button

The function is interrupted and the parameter window closes via the button.

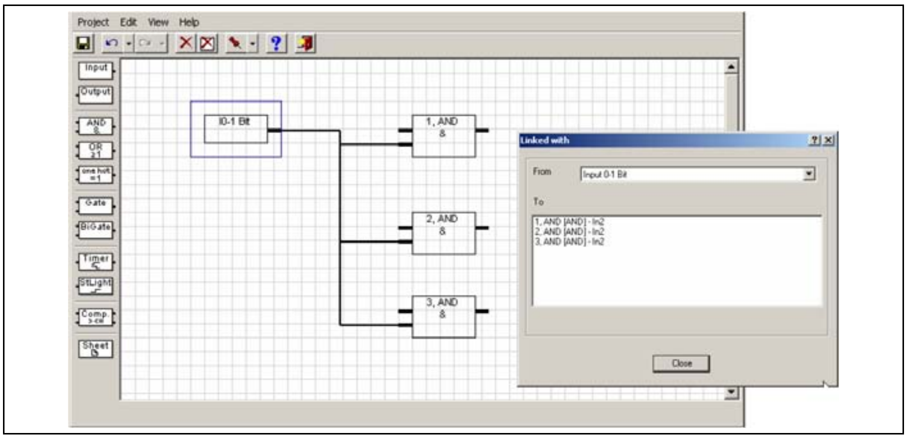
3.5.6.Deleting links

Links are deleted:

- By selecting the link and pressing the Del button
- By selecting the connection, pressing the right mouse button and selecting the *Delete* command.

Note: Once inputs, outputs and functions have been change in the worksheet, the links are automatically updated.

3.5.7.Displaying links



From: Input 0-1 Bit

To:

1, AND [AND] - In2
2, AND [AND] - In2
3, AND [AND] - In2

Fig. 49: Displaying links

The parameter window *Linked with...* is activated by the following procedure:

- Select the element and open the context menu with the right mouse button. Then select the *Linked with …* command.

Close button

The parameter window is closed via the button.

3.6. Functions

Note: The standard settings for the options are underlined, e.g. option: yes/no.

3.6.1.Inputs and outputs

The inputs and outputs in the application program correspond with the communication objects in the ETS3. They are visible in the ETS3 and can be processed there.

3.6.1.1. Input

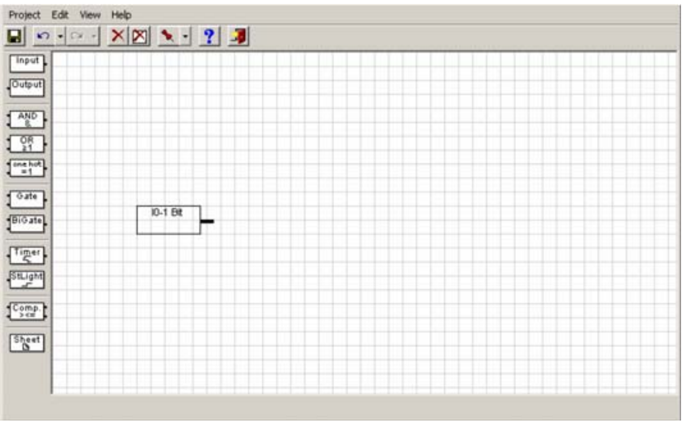


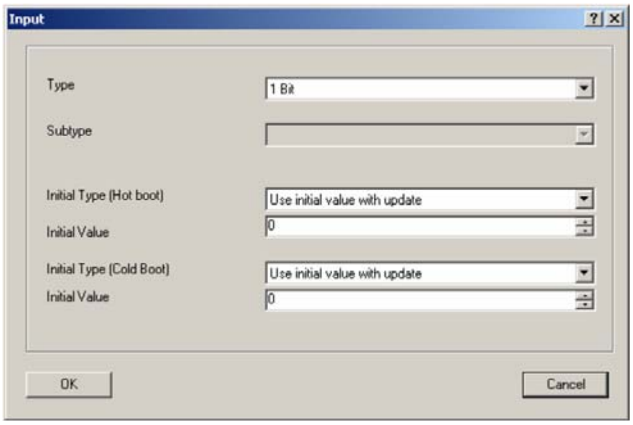
Fig. 50: Input

The connection for the input is on the right of the element.

Note: In the event of a bus voltage failure, the values of the input objects are stored for at least an hour. With a bus voltage failure that is longer than one hour in duration the values of the input objects are lost. A check is therefore made in the ABL/S 2.1 on bus voltage recovery before processing the values of the input objects to see whether data loss has occurred in the meantime.

The parameter window *Input* becomes active through the following procedure:

- Select the element with the right mouse button of the *Edit parameter*… context menu.


Input

Type: 1 Bit

Subtype:

Initial Type (Hot boot): Use initial value with update

Initial Value: 0

Initial Type (Cold Boot): Use initial value with update

Initial Value: 0

Fig. 51: Parameter window “Input”

In the *Input parameter* window the type, subtype and the behaviour with a bus voltage recovery are defined.

Note: If a group address is already defined for the element, the parameters *Type* and *Subtype* may not be changed.

Type

Options: 1 Bit
2 Bit
4 Bit
1 Byte
2 Byte
4 Byte

The type is assigned with this parameter.

The following parameter Subtype is only active in the parameter *Type* if the options *1 byte, 2 byte or 4 byte* are selected.

Subtype

Options: Floating comma (only active with 2 byte)
Signed
Unsigned

The subtype is assigned with this parameter.

On the graphic interface the

Floating comma is shown as (f),
Signed as (s) and
Unsigned with (u).

Initial Type (Hot boot)

Options: Use value before bus voltage dropped without update
Use initial value without update
Use value before bus voltage dropped with update
Use initial value with update
Ask bus for value

Under …*with update*… should be understood as a telegram with preset value simulated on the input. Thereafter the calculation of the input via the linked function to the output occurs.

The behaviour of the parameter can be compared with a device warm start.

What is a Hot boot?

If the saved values are still available, the Application Unit Logic leads to a so-called Hot boot with bus voltage recovery, i.e. it can still operate with the saved data, but can still used the predefined data or read the relevant data for them from other bus devices.

Without update… can be understood to mean that the input assumes the value before the reset or the preset value. With the next event (telegram) to any input which is associated with this function, the function is calculated with this value.

The following parameter *Initial Value* is only active if in the parameter *Initial Type (Hot boot)* the options Use *initial value without update or Use initial value with update* has been selected.

Initial Value

Different settings are possible in the options depending on the *Type* and *Subtype* which have been selected.

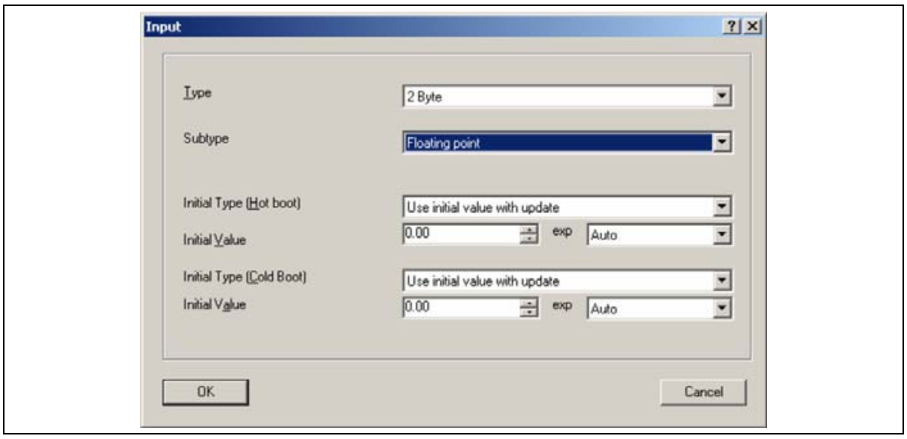
Options: 1 Bit = 0/1
2 Bit = 0…3
4 Bit = 0…15

1 Byte Unsigned = 0…255
1 Byte Signed = -128…0…127

2 Byte Floating point
2 Byte Unsigned = 0…65.535
2 Byte Signed = -32.768…0…32.767

4 Byte Unsigned = 0…4.294.967.295
4 Byte Signed = -2.147.483.648…0…2.147.483.647

With the selection *Floating point* in the parameter *Subtype* the parameter *Exponent (exp)* appears.



Input

Type: 2 Byte
Subtype: Floating point

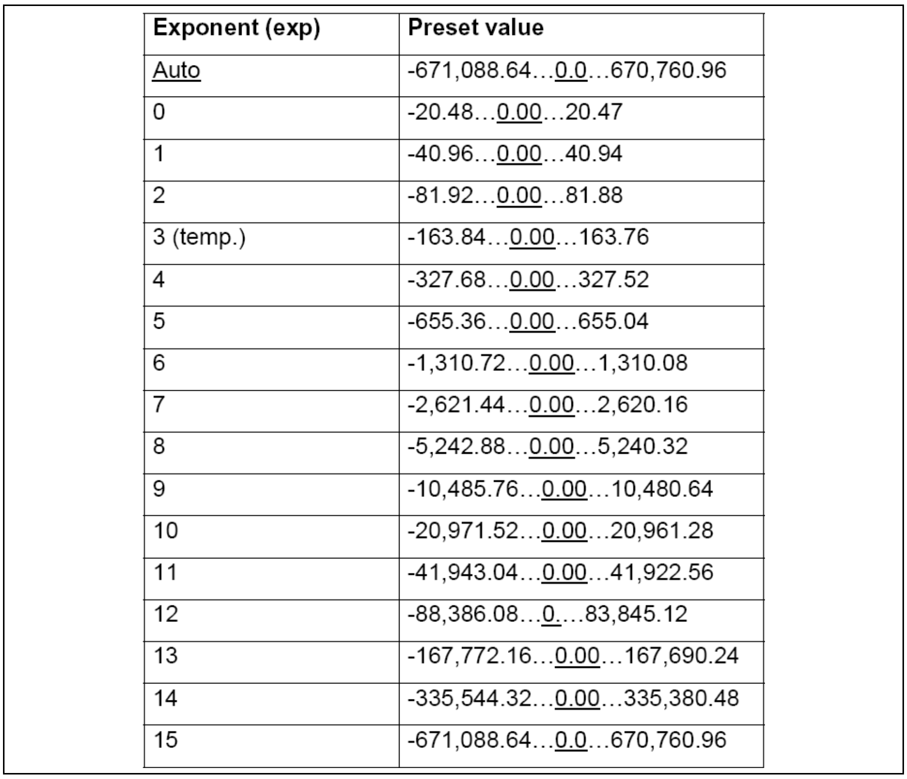
Initial Type (Hot boot): Use initial value with update
Initial Value: 0.00 exp Auto

Initial Type (Cold Boot): Use initial value with update
Initial Value: 0.00 exp Auto

Fig. 52: Parameter window “Input - Exponent”

exp / preset value

The preset value depends on the power.



Exponent (exp)	Preset value
Auto	-671,088.64…0.0…670,760.96
0	-20.48…0.00…20.47
1	-40.96…0.00…40.94
2	-81.92…0.00…81.88
3 (temp.)	-163.84…0.00…163.76
4	-327.68…0.00…327.52
5	-655.36…0.00…655.04
6	-1,310.72…0.00…1,310.08
7	-2,621.44…0.00…2,620.16
8	-5,242.88…0.00…5,240.32
9	-10,485.76…0.00…10,480.64
10	-20,971.52…0.00…20,961.28
11	-41,943.04…0.00…41,922.56
12	-88,386.08…0.…83,845.12
13	-167,772.16…0.00…167,690.24
14	-335,544.32…0.00…335,380.48
15	-671,088.64…0.0…670,760.96

Table 2: Exponent

Initial Type (Cold boot)

Options: Use initial value without update
Use initial value with update
Ask bus for value

Under …*with update*… should be understood as a telegram with preset value simulated on the input. Thereafter the calculation of the input via the linked function to the output occurs.

The behaviour of the parameter can be compared with a device cold start.

What is a Cold Boot?

In case of a data loss the Application Unit Logic undertakes a so-called *Cold Boot* with bus voltage recovery, i.e. it can no longer access saved data and must operate either with predefined data or must read the data relevant for itself from other bus devices.

Without update… can be understood to mean that the input assumes the value before the reset or the preset value. With the next event (telegram) to any input, which is associated with this function, the function is calculated with this value.

The following parameter *Initial Value* is only active if in the parameter *Initial Type (Cold boot)* the options *Use initial value without update or Use initial value with update* has been selected.

Initial Value

Different settings are possible in the options depending on the *Type* and *Subtype* which have been selected.

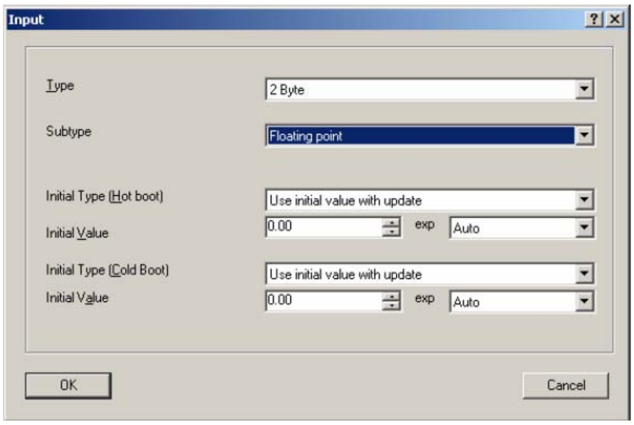
Options: 1 Bit = 0/1
2 Bit = 0…3
4 Bit = 0…15

1 Byte Unsigned = 0…255
1 Byte Signed = -128…0…127

2 Byte Floating point
2 Byte Unsigned = 0…65.535
2 Byte Signed = -32.768…0…32.767

4 Byte Unsigned = 0…4.294.967.295
4 Byte Signed = -2.147.483.648…0…2.147.483.647

With the selection *Floating point* in the parameter *Subtype* the parameter *Exponent (exp)* appears.



Input

Type: 2 Byte
Subtype: Floating point

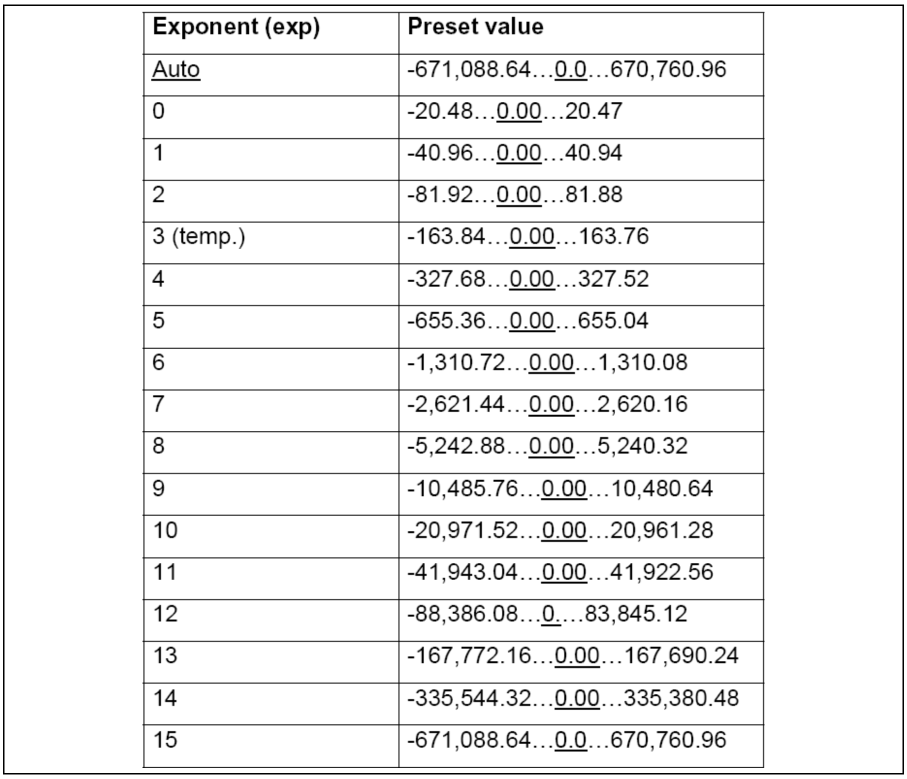
Initial Type (Hot boot): Use initial value with update
Initial Value: 0.00 exp Auto

Initial Type (Cold Boot): Use initial value with update
Initial Value: 0.00 exp Auto

Fig. 53: Parameter window “Input - Exponent”

exp / preset value

The preset value depends on the power.



Exponent (exp)	Preset value
Auto	-671,088.64…0.0…670,760.96
0	-20.48…0.00…20.47
1	-40.96…0.00…40.94
2	-81.92…0.00…81.88
3 (temp.)	-163.84…0.00…163.76
4	-327.68…0.00…327.52
5	-655.36…0.00…655.04
6	-1,310.72…0.00…1,310.08
7	-2,621.44…0.00…2,620.16
8	-5,242.88…0.00…5,240.32
9	-10,485.76…0.00…10,480.64
10	-20,971.52…0.00…20,961.28
11	-41,943.04…0.00…41,922.56
12	-88,386.08…0.…83,845.12
13	-167,772.16…0.00…167,690.24
14	-335,544.32…0.00…335,380.48
15	-671,088.64…0.0…670,760.96

Table 3: Exponent

OK button

The settings are copied and the parameter window closes via the button.

Cancel button

The function is interrupted and the parameter window closes via the button.

3.6.1.2. Output

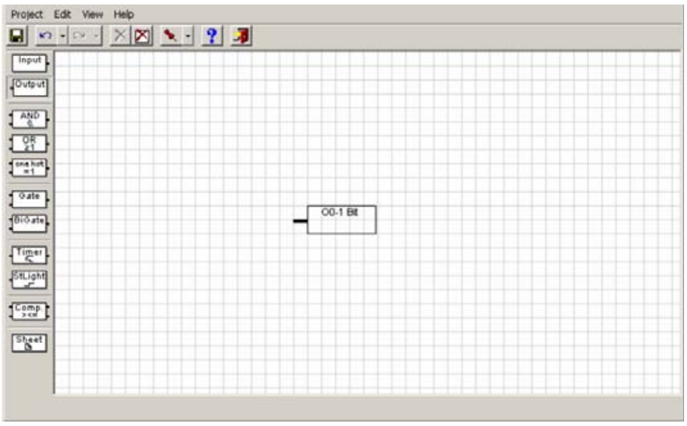
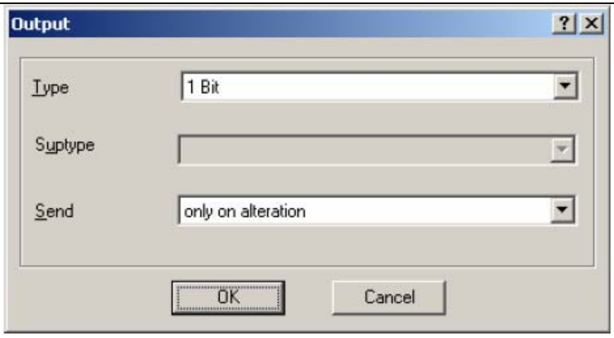


Fig. 54: Output

The connection for the output is on the left of the element.

The parameter window *Output* becomes active through the following procedure:

- Select the element with the right mouse button of the *Edit parameter…* context menu.



Type: 1 Bit

Suptype:

Send: only on alteration

Fig. 55: Parameter window “Output”

In the *Output* parameter window the type, subtype and sending of the output are defined.

Note: If a group address is already defined for the element, the parameters *Type* and *Subtype* may not be changed.

Type

Options: 1 Bit,
2 Bit
4 Bit
1 Byte
2 Byte
4 Byte

The *type* is assigned with this parameter.

The following parameter *Subtype* is only active in the parameter *Type* if the options *1 byte, 2 byte or 4 byte* are selected.

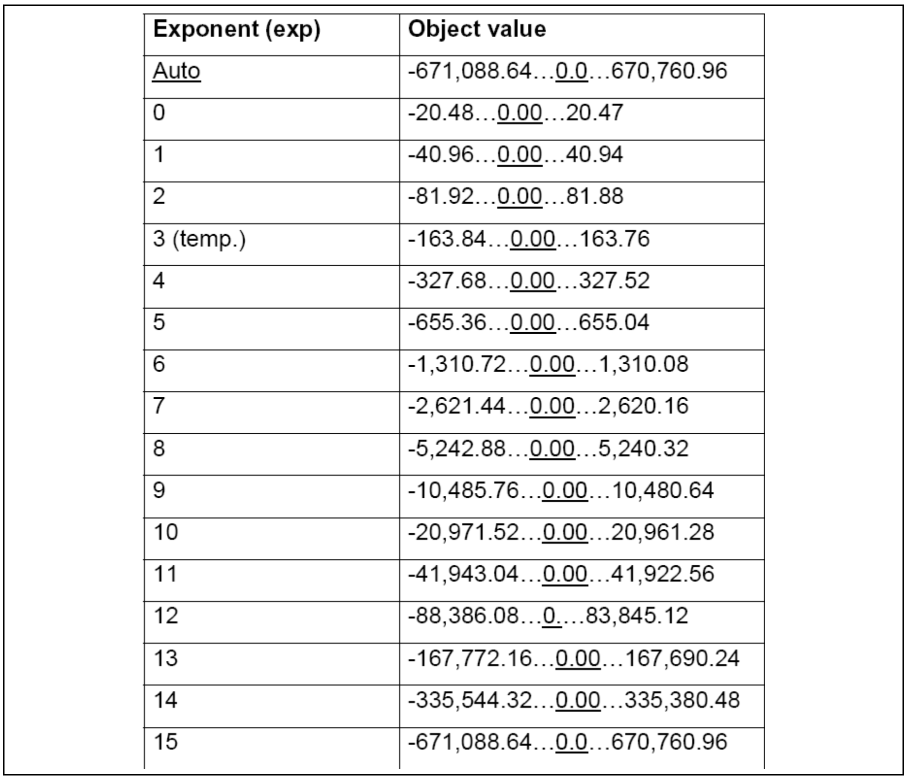
Subtype

Options: Floating point (only active with 2 byte)
Signed
Unsigned

The subtype is assigned with this parameter.
In the graphic interface the
Floating point is signified as (f),
with Signed as (s)
and Unsigned with (u).

1 Byte Unsigned =	0…255
1 Byte Signed =	-128…127…
2 Byte Floating point, see table below	
2 Byte Unsigned =	0…65.535
2 Byte Signed =	-32.768…32.767…
4 Byte Unsigned =	0…4.294.967.295
4 Byte Signed =	-2.147.483.648…2.147.483.647…

The output object type can appear as follows.



Exponent (exp)	Object value
Auto	-671,088.64…0.0…670,760.96
0	-20.48…0.00…20.47
1	-40.96…0.00…40.94
2	-81.92…0.00…81.88
3 (temp.)	-163.84…0.00…163.76
4	-327.68…0.00…327.52
5	-655.36…0.00…655.04
6	-1,310.72…0.00…1,310.08
7	-2,621.44…0.00…2,620.16
8	-5,242.88…0.00…5,240.32
9	-10,485.76…0.00…10,480.64
10	-20,971.52…0.00…20,961.28
11	-41,943.04…0.00…41,922.56
12	-88,386.08…0.…83,845.12
13	-167,772.16…0.00…167,690.24
14	-335,544.32…0.00…335,380.48
15	-671,088.64…0.0…670,760.96

Table 4: Exponent

Send

Options: only on alteration
on each calculation value

Option *only on alteration* = the output only sends a telegram if the object value changes.

Note: The output is only assigned with a new object value, if a telegram has been sent to an assigned input, which changes the object value (e.g. from 0 to 1).

Option *on each calculation value* = the output sends a telegram with every assignment of a new object value.

Note: A new object value is then always given to the output if a telegram was sent to an allocated input, regardless of whether the object value has changed (e.g. from 0 to 1) or has remained the same.
An exception exists if an inhibited gate or a non timed-out timer is located between an input and an output. In both these instances the output is not assigned a new object value.

OK button

The settings are copied and the parameter window closes via the button.

Cancel button

The function is interrupted and the parameter window closes via the button.

An example:

If on the signal path to an output a feedback is located, i.e. the output of a gate is connected with the input of a gate on the same signal path, then a new object value is continuously assigned to the output by this feedback. The output object then continually sends telegrams.

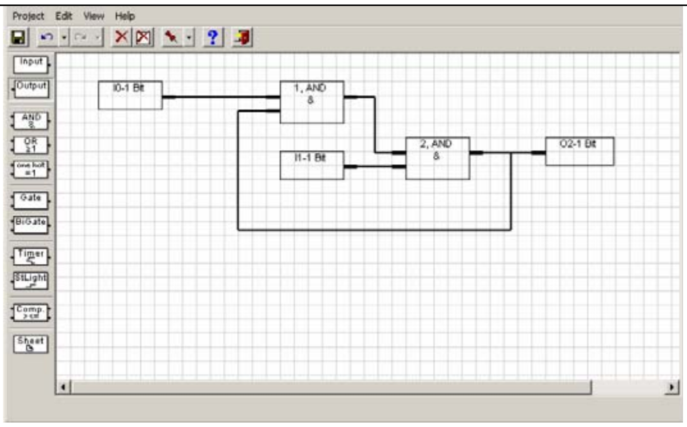
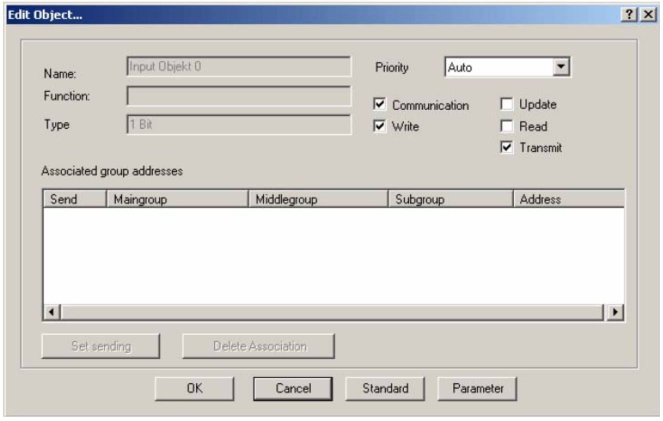


Fig. 56: Feedback example

3.6.1.3. Edit communication object

The parameter window *Edit Object...* is activated by the following procedure:

- By a double click on the input or output object
- Select the input or output object and open *Edit* in the context menu with the right mouse button



Edit Object...

Name: Input Objekt 0

Function:

Type 1 Bit

Priority Auto

☑ Communication
☑ Write
☐ Update
☐ Read
☑ Transmit

Associated group addresses

Send	Maingroup	Middlegroup	Subgroup	Address

Fig. 57: Parameter window “Edit Object”

Name

Options: Input Object X/
Output Object X

Here the display indicates if it is an input object or an output object. At the same time the number of the communication object is represented.

Function

Is empty

Type

Options: 1 Bit/
2 Bit/
4 Bit/
1 Byte/
2 Byte/
4 Byte

Here the bit to byte type concerned is represented.

Priority

Options: Auto
Normal
Alarm

The *Priority* can be set to the following values:

Option Auto = low priority for time uncritical functions

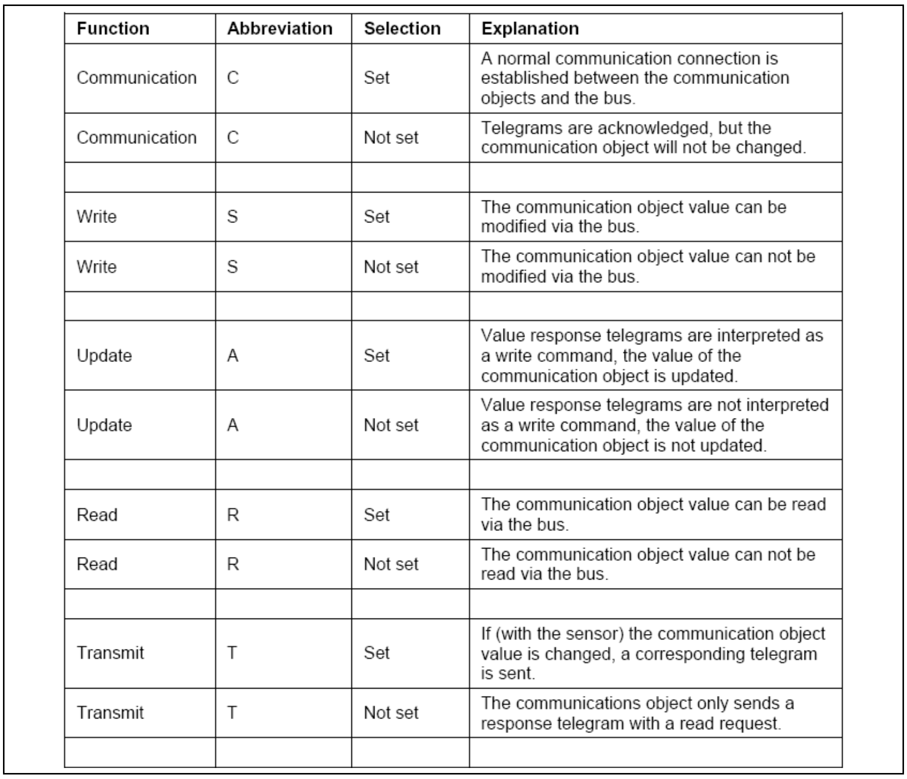
Option Normal = normal priority for manually actuated functions

Option Alarm = high priority for time-critical functions

Flag Communication, Write, Update, Read and Transmit

Options: activated = ticked
not activated = not ticked

The flags are set by activating the corresponding checkbox.



Function	Abbreviation	Selection	Explanation
Communication	C	Set	A normal communication connection is established between the communication objects and the bus.
Communication	C	Not set	Telegrams are acknowledged, but the communication object will not be changed.
Write	S	Set	The communication object value can be modified via the bus.
Write	S	Not set	The communication object value can not be modified via the bus.
Update	A	Set	Value response telegrams are interpreted as a write command, the value of the communication object is updated.
Update	A	Not set	Value response telegrams are not interpreted as a write command, the value of the communication object is not updated.
Read	R	Set	The communication object value can be read via the bus.
Read	R	Not set	The communication object value can not be read via the bus.
Transmit	T	Set	If (with the sensor) the communication object value is changed, a corresponding telegram is sent.
Transmit	T	Not set	The communications object only sends a response telegram with a read request.

Table 5: Communication flags

Associated group adresses

The table is divided into Send, Maingroup, Middlegroup, Subgroup and Address.

Send

Option yes/no

Maingroup

The name of the main group is displayed here.

Middlegroup

The name of the middle group is displayed here.

Subgroup

The name of the lower group is displayed here.

Address

The group address is displayed here

Set sending button

With this button a group address is marked as sent.

Note: The button becomes active as soon as a group address has been assigned.

Delete Association button

The assignment (link) between the group address and the communication object is deleted using this button

Note: The button becomes active as soon as a group address has been assigned.

OK button

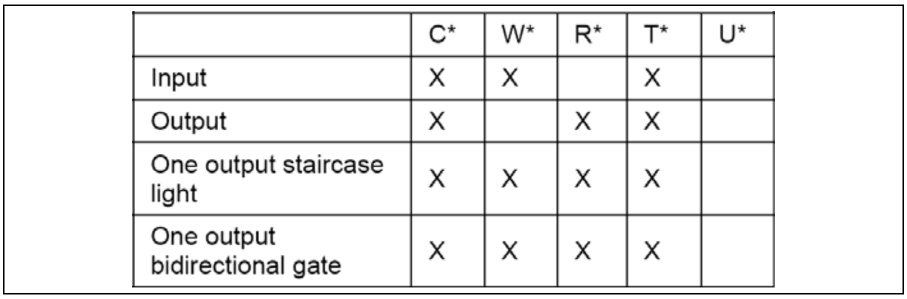
The settings are copied and the parameter window closes via the button.

Cancel button

The function is interrupted and the parameter window closes via the button.

Standard button

Using this button all flags are reset to the following standard settings.



	C*	W*	R*	T*	U*
Input	X	X		X	
Output	X		X	X	
One output staircase light	X	X	X	X	
One output bidirectional gate	X	X	X	X	

Table 6: Standard settings of the flags

* C = communication
W = write
R = read
T = transmit
U = update

Note: The standard flags should be changed in a special case.

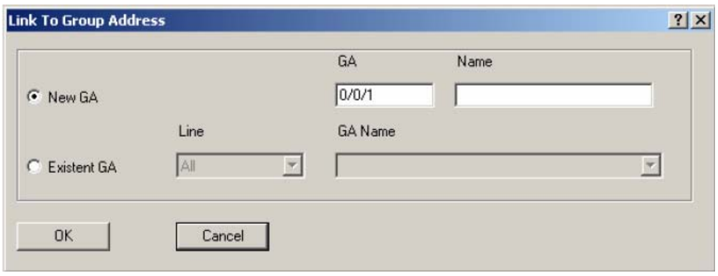
Parameter button

Via this button you can access the parameter window input or output.

3.6.1.4. Link with group addresses

The parameter window *Link To Group Address* becomes active with the following procedure:

- Select the element with the right mouse button of the *Link with group address…* context menu.



Link To Group Address

● New GA — GA: 0/0/1 — Name:

○ Existent GA — Line: All — GA Name:

Fig. 58: Parameter window “Link To Group Address”

Note: In the parameter window *Link To Group Address* a new group address can be created or group addresses which already exist can be selected.

New GA

Options: active = black point in the circle
not active = no black point in the circle

A new group address is created via the function.

GA

A new group address is entered here.

Note: The next free group address is automatically displayed here.

Name

The name of the group address is entered here.

Existent GA

Options: active = black point in the circle
not active = no black point in the circle

An *existing group address* is assigned via this function.

Line

Options: All
Unused

The selection of existing group addresses can be sorted according to *All*, *Line* or *Unused*. If *All* is selected, all the existing group addresses are available for selection.

Note: After selection of links between input and output objects with the right mouse button the assigned group addresses can be deleted under the *Edit Object* menu point.

OK button

The settings are copied and the parameter window closes via the button.

Cancel button

The function is interrupted and the parameter window closes via the button.

3.6.2.AND gate

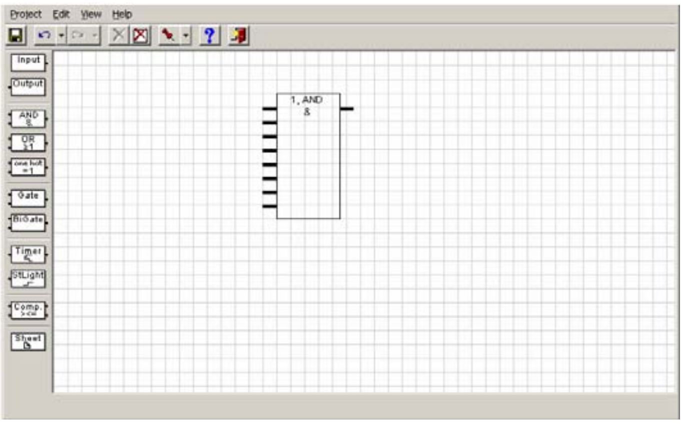


Fig. 59: AND gate

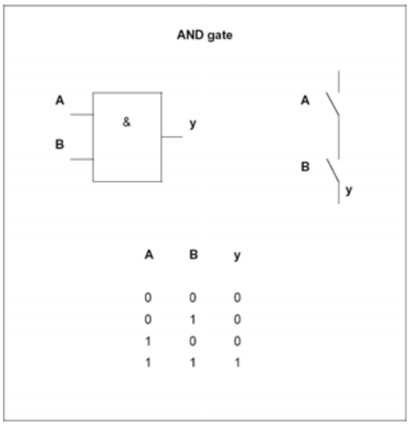
An AND gate has up to 8 inputs and one output.
When newly created the gate has 2 inputs and one output.
The inputs and outputs can be inverted.

The inputs of the AND gate can each be linked with an input object, or an output of another function. The output of a logical element can be linked with an output object or an input of another function. An output of a logical element can thus be linked with several inputs of logical elements.

The inputs are located on the left of the element.

The output is located on the right of the element.

Note: Only 1 bit objects are allowed for the inputs and the output of the AND gate.



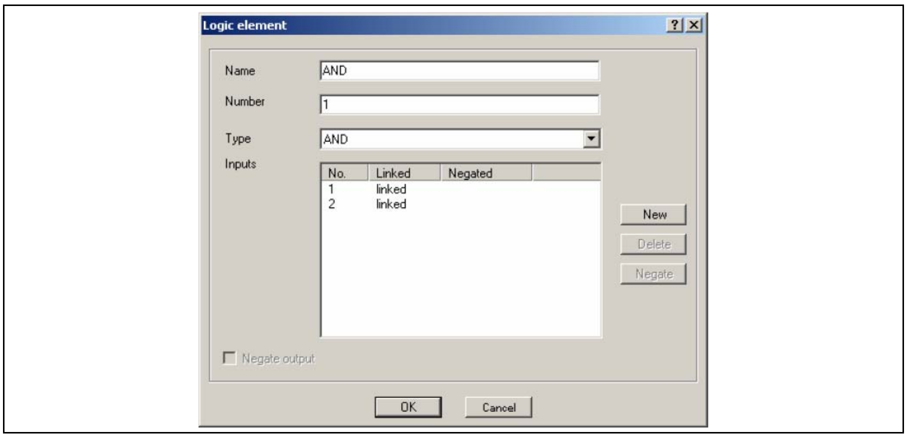
A	B	y
0	0	0
0	1	0
1	0	0
1	1	1

Fig. 60: AND gate truth table

3.6.2.1. Creating and deleting inputs

The parameter window *Logic element* becomes active through the following procedure:

- By a double click on the AND gate.
- Select the element and open the *Parameter* context menu with the right mouse button.



Logic element

Name: AND

Number: 1

Type: AND

Inputs

No.	Linked	Negated
1	linked	
2	linked	

□ Negate output

Fig. 61: Parameter window “AND”“

Name

Options: <u>AND</u>

The name can be chosen as required. A maximum of 36 characters can be used.

Number

Options: <u>1</u>…50

The number is successive. It can be freely assigned if it is not being used by another element.

Type

Options: <u>AND/</u>
OR/
one hot

The type can be chosen as required.

On the graphic interface the

AND	displayed as an &,
the OR	as a >=1 and
the one hot	as =1.

Inputs

In the columns the successive numbers display of an input is linked or an input is negated (inverted).

The meaning of the individual column is as follows:

No.

Options: 1…8

The number is automatically assigned; it indicates the corresponding number of the input.

Linked

Options: no text/
linked

On this column you display if an input is linked or not.

Negated

Options: no text/
negated

On this column you display if an input is negated or not.

New button

Further inputs can be added via the button.

Note: If inputs are added, the previous associations are preserved. A maximum of 8 inputs can be created.

Delete button

An input can be deleted via the button.

Note: The *Delete* button is only active if the input is not linked. If the number of inputs is reduced, only those inputs that have no associations are deleted.

Negate button

An input can be negated via the button. In the graphic user interface a negated input is signified by a circle.

Note: An output must be linked in order to negate it. The negation of a linked input can be set or deleted directly in the graphical user environment by a double click with the left mouse button on the respective unit.

Negate output

Options:activated = ticked
not activated = not ticked

The output can be negated by activating the checkbox. In the graphic user interface a negated output is signified by a circle.

Note:An output must be linked in order to negate it. The negation of a linked output can be set or deleted directly in the graphical user environment by a double click with the left mouse button on the respective output.

OK button

The settings are copied and the parameter window closes via the button.

Cancel button

The function is interrupted and the parameter window closes via the button.

3.6.3.OR gate

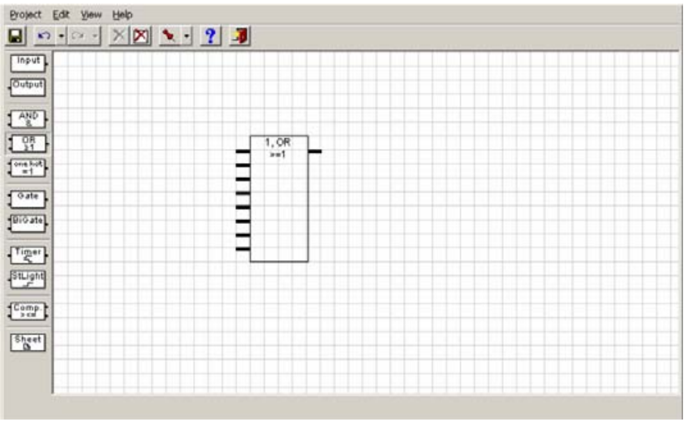


Fig. 62: OR gate

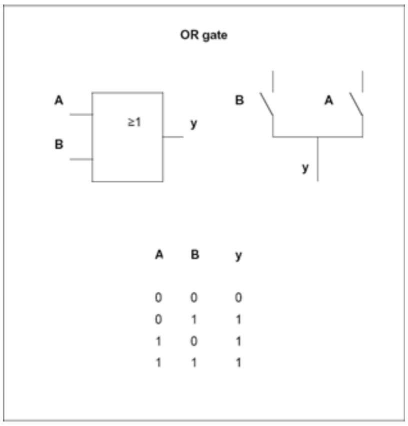
An OR gate has up to 8 inputs and one output.
When newly created the gates have 2 inputs and one output.
The inputs and outputs can be negated (inverted).

The inputs of the OR gate can each be linked with an input object, or an output of another function. The output of a logical element can be linked with an output object or an input of another function. An output of a logical element can thus be linked with several inputs of logical elements.

The inputs are located on the left of the element.

The output is located on the right of the element.

Note: Only 1 bit objects are allowed for the inputs and the output of the OR gate.



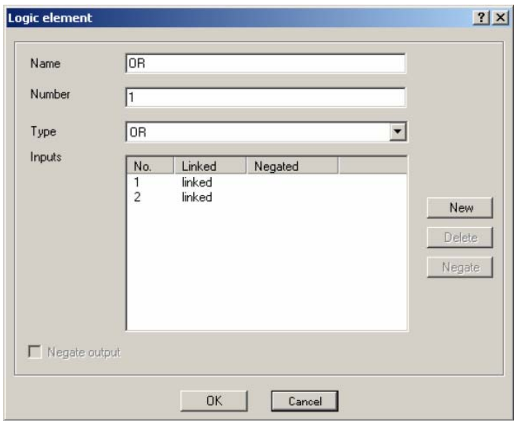
A	B	y
0	0	0
0	1	1
1	0	1
1	1	1

Fig. 63: OR gate truth table

3.6.3.1. Creating and deleting inputs

The parameter window *Logic element* becomes active through the following procedure:

- By a double click on the OR gate.
- Select the element and open the *Parameter* context menu with the right mouse button.



Name: OR
Number: 1
Type: OR
Inputs

No.	Linked	Negated
1	linked	
2	linked	

□ Negate output

Fig. 64: Parameter window “OR gate”

Name

Options: OR

The name can be chosen as required. A maximum of 36 characters can be used.

Number

Options: 1…50

The number is successive. It can be freely assigned if it is not being used by another element.

Type

Options: AND/
OR/
one hot

The type can be chosen as required.

On the graphic interface the

AND displayed as an &,
the OR as a >=1 and
the one hot as a =1.

Inputs

In the columns the successive numbers display of an input is linked or an input is inverted (negated).

The meaning of the individual column is as follows:

No.

Options: 1…8

The number is automatically assigned; it indicates the corresponding number of the input.

Linked

Options: no text/
linked

On this column you display if an input is linked or not.

Negated

Options: no text/
negated

On this column you display if an input is negated or not.

New button

Further inputs can be added via the button.

Note: If inputs are added, the previous associations are preserved. A maximum of 8 inputs can be created.

Delete button

An input can be deleted via the button.

Note: The *Delete* button is only active if the input is not linked. If the number of inputs is reduced, only those inputs that have no associations are deleted.

Negate button

An input can be negated via the button. In the graphic user interface a negated input is signified by a circle.

Note: An output must be linked in order to negate it. The negation of a linked input can be set or deleted directly in the graphical user environment by a double click with the left mouse button on the respective unit.

Negate output

Options: activated = ticked
not activated = not ticked

The output can be negated by activating the checkbox. In the graphic user interface a negated output is signified by a circle.

Note: An output must be linked in order to negate it. The negation of a linked output can be set or deleted directly in the graphical user environment by a double click with the left mouse button on the respective output.

OK button

The settings are copied and the parameter window closes via the button.

Cancel button

The function is interrupted and the parameter window closes via the button.

3.6.4.One hot gate

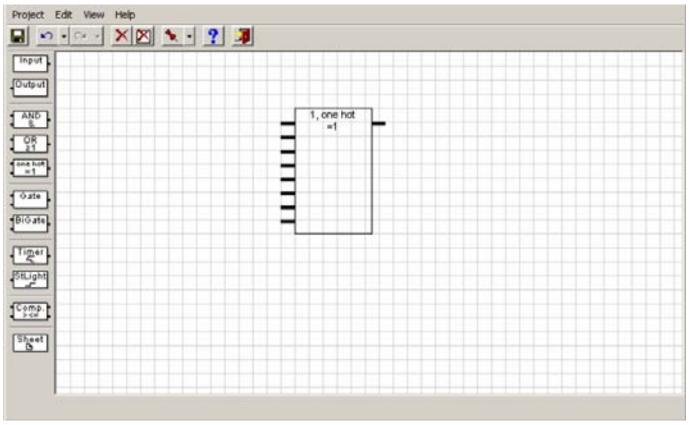


Fig. 65: One hot gate

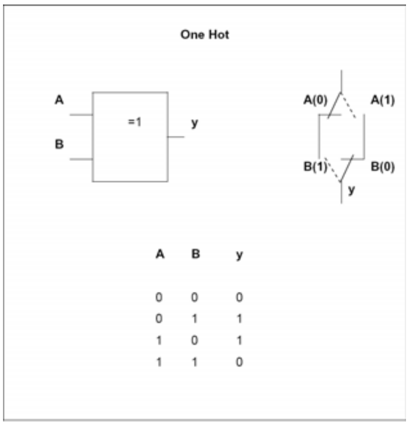
An One hot gate has up to 8 inputs and one output.
When newly created the gate has 2 inputs and one output.
The inputs and outputs can be negated (inverted).

The input of the One hot gate can each be linked with an input object, or an output of another function. The output of a logical element is linked with an output object or an input of another function. An output of a logical element can thus be linked with several inputs of logical elements.

The inputs are located on the left of the element.

The output is located on the right of the element.

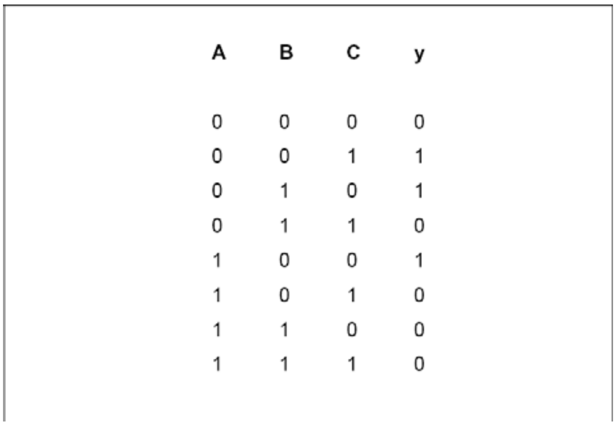
Note: Only 1 bit objects are allowed for the inputs and the output of the One hot gate.



A	B	y
0	0	0
0	1	1
1	0	1
1	1	0

Fig. 66: One hot – truth table 2 inputs

With 3 inputs or more the truth table appears as follows:



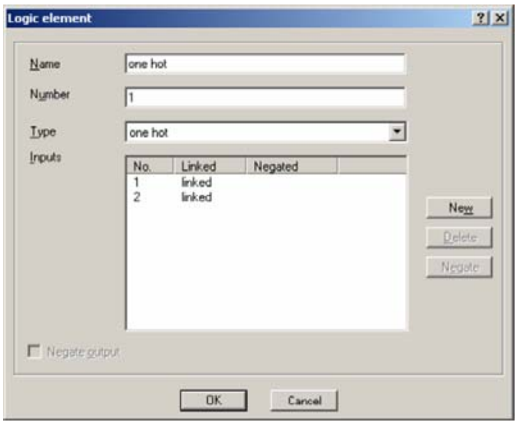
A	B	C	y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

Fig. 67: One hot – truth table 3 inputs

3.6.4.1. Creating and deleting inputs

The parameter window *Logic element* becomes active through the following procedure:

- By a double click on the One hot gate
- Select the element and open the *Parameter* context menu with the right mouse button.



Name: one hot

Number: 1

Type: one hot

Inputs

No.	Linked	Negated
1	linked	
2	linked	

☐ Negate output

Fig. 68: Parameter window One hot gate

Name

Options: one hot

The name can be chosen as required. A maximum of 36 characters can be used.

Number

Options: 1…50

The number is successive. It can be freely assigned if it is not being used by another element.

Type

Options: AND/
OR/
one hot

The type can be chosen as required.

On the graphic interface the

AND displayed as an &,

the OR as a >=1 and

the one hot as a =1.

Inputs

In the columns the successive numbers display of an input is linked or an input is inverted (negated).

The meaning of the individual column is as follows:

No.

Options: 1…8

The number is automatically assigned; it indicates the corresponding number of the input.

Linked

Options: no text/
linked

On this column you display if an input is linked or not.

Negated

Options: no text/
negated

On this column you display if an input is negated or not.

New button

Further inputs can be added via the button.

Note: If inputs are added, the previous associations are preserved. A maximum of 8 inputs can be created.

Delete button

An input can be deleted via the button.

Note: The *Delete* button is only active if the input is not linked. If the number of inputs is reduced, only those inputs that have no associations are deleted.

Negate button

An input can be negated via the button. In the graphic user interface a negated input is signified by a circle.

Note: An output must be linked in order to negate it. The negation of a linked input can be set or deleted directly in the graphical user environment by a double click with the left mouse button on the respective unit.

Negate output

Options: activated = ticked
not activated = not ticked

The output can be negated by activating the checkbox. In the graphic user interface a negated output is signified by a circle.

Note: An output must be linked in order to negate it. The negation of a linked output can be set or deleted directly in the graphical user environment by a double click with the left mouse button on the respective output.

OK button

The settings are copied and the parameter window closes via the button.

Cancel button

The function is interrupted and the parameter window closes via the button.

3.6.5.Unidirectional gate

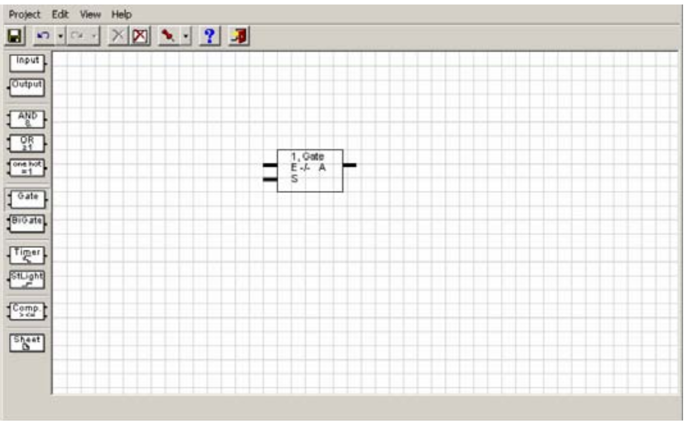


Fig. 69: Unidirectional gate

A unidirectional gate has one input, one output and one control input. The input also features a filter and save function. If an event occurs at the input and the gate is enabled, it can transfer the value at the input to the output. If the gate is disabled, there is no reaction.

The input is identified with an E and is located on the top left of the element.

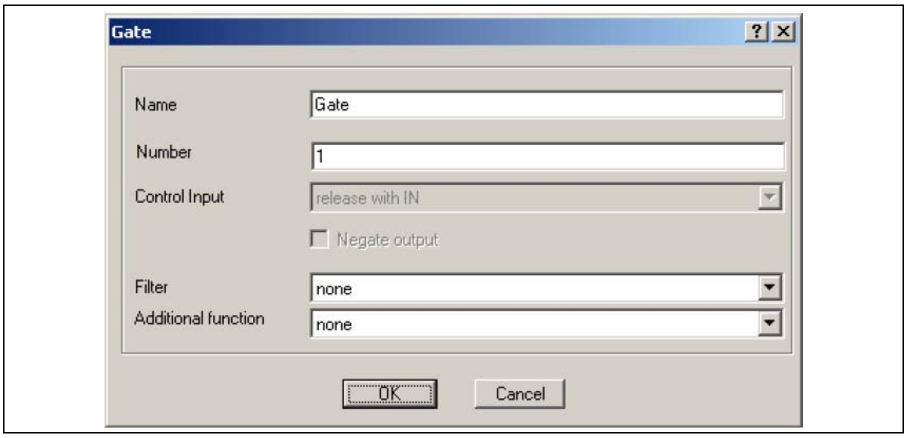
The control input is identified with an S and is located on the top left of the element.

The output is identified with an A and is found at the bottom right of the element.

Note: For input and output all communication object types are permissible, however they must be identical.
The input is linked with an input object or with an output of another function.
The output is linked with an output object or with the input of another function.

The parameter window *Gate* becomes active through the following procedure:

- By a double click on the element
- By opening of the context window with the right mouse button and the *Edit parameter...* command.



Gate

Name: Gate
Number: 1
Control Input: release with IN
☐ Negate output
Filter: none
Additional function: none

Fig. 70: Parameter window “Bidirectional gate”

Name

Options: Gate

The name can be chosen as required. A maximum of 36 characters can be used.

Number

Options: 1…50

The number is successive. It can be freely assigned if it is not being used by another element.

Control Input

Options: release with IN/
release with OUT

On the graphic user interface the option *release with OUT* is identified by a circle. The option *release with IN* is not marked. The control input is marked with an S.

Note: In order to set the control input it must be connected. The control input can be connected with an input object or the output of another function, only 1 bit objects are permissible.

Negate output

Options activated = ticked
not activated = not ticked

The output can be negated by activating the checkbox. In the graphic user interface a negated output is signified by a circle.

Note: An output must be linked in order to negate it. The negation of a linked output can be set or deleted directly in the graphical user environment by a double click with the left mouse button on the respective output.

Filter

Options: none/
passthrough object value 0/
passthrough object value 1

On the graphic user interface the option none is represented by a dash, the option *passthrough object value 0* and the option *passthrough object value 1.*

Option *none* (-) = no filter function active.

Option *passthrough object value 0* = communication objects with value 0 are filtered out.

Option *passthrough object value 1* = communication objects with value 1 are filtered out.

Note: The filter can only be activated if the input is assigned to a 1 bit object. The filter function affects the input.

Where is the filter function located in the process chain?

The filter function is situated before the gate, i.e. the input signal initially goes through a filter and then through a gate function.

Additional function

Options: none/
Save during blocking phase/
Save value with send trigger

If *none* is selected the first received communication object received after the gate is enabled is transferred on to the output of the gate. In the graphic user interface the option *none* is identified by a dash.

If *Save is selected during the blocking phase,* then the communication object value which was last present on the input before enable is transferred onto the output. On the graphic user interface the option *Save during blocking phase* is identified by an S.

If you select *Save value with send trigger*, the gate output sends with every enable command on the control input, even if the gate is already enabled and the communication object value on the input has not changed. On the graphic user interface the option *Save value with send trigger* is identified by a T.

Note: In order for this function to be effective, the assigned output must be parameterised accordingly (output sends with every new assignment of a new object value).

OK button

The settings are copied and the parameter window closes via the button.

Cancel button

The function is interrupted and the parameter window closes via the button.

3.6.6.Bidirectional Gate

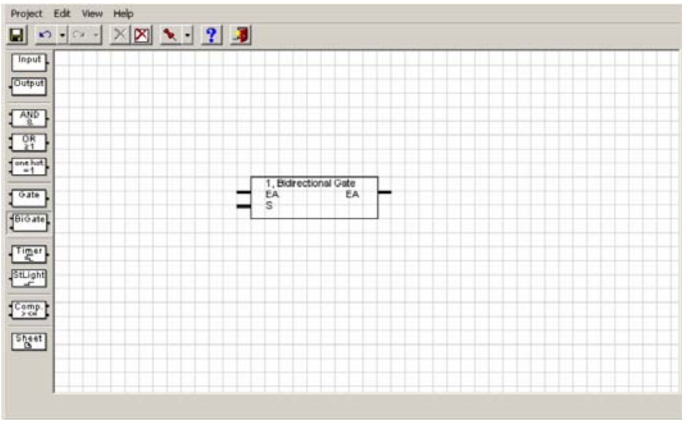


Fig. 71: Bidirectional Gate

The bidirectional gate features an input and an output, as well as a control input. No filtering is possible with a bidirectional gate. The enabled gate can transfer a communication object value in both directions. If an input or output object is linked with an input or output of the bidirectional gate, this automatically becomes an input **and** output.

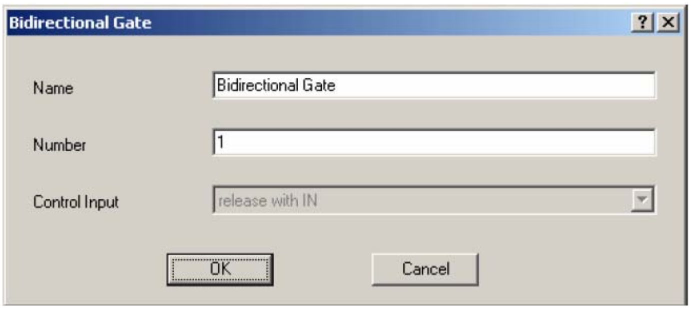
The input and output are identified with the designation EA and can for found at the top left and top right on the element.

The control input is identified with an C and is located on the top left of the element.

Note: For input and output all communication object types are permissible, however they must be identical.
The input can only be linked to an input/output object. Not with an output or another function.
The output can only be linked to an input/output object. Not with an input or another function.

The parameter window *Bidirectional Gate* becomes active through the following procedure:

- By a double click on the element
- By opening of the context window with the right mouse button and the *Edit parameter...* command.


Bidirectional Gate

Name: Bidirectional Gate

Number: 1

Control Input: release with IN

Fig. 72: Parameter window “Bidirectional Gate”

Name

Options: Bidirectional Gate

The name can be chosen as required. A maximum of 36 characters can be used.

Number

Options: 1…50

The number is successive. It can be freely assigned if it is not being used by another element.

Control Input

Options: release with IN /
release with OUT

On the graphic user interface the option *release with OUT* is identified by a circle. The option *release with IN* is not marked. The control input is marked with an S.

Note: In order to set the control input it must be connected.
The control input can be connected with an input object or the output of another function, only 1 bit objects are permissible.

OK button

The settings are copied and the parameter window closes via the button.

Cancel button

The function is interrupted and the parameter window closes via the button.

3.6.7.Timer

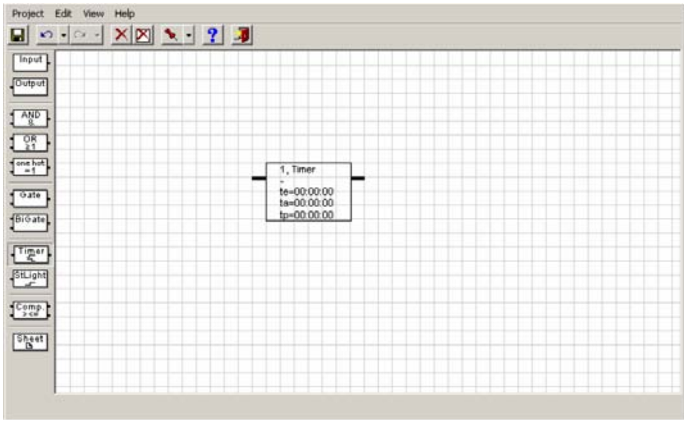


Fig. 73: Timer

A timer has an input and an output. The input also has another filter. If an event occurs at the input, the timer can transfer the value at the input to the output with a time delay.

The input of the timer is located on the left of the element.

The output of the timer is located on the right of the element.

Note: The switch on/off telegrams can be re-triggered!

What is the initial state?

With bus voltage recovery all ongoing delays and pulse duration's are stopped. The timer waits for the result on an input.

Note: The timer is connected with an input object or the output of another function, only 1 bit objects are permissible. The output is linked with an output object or with the input of another function.

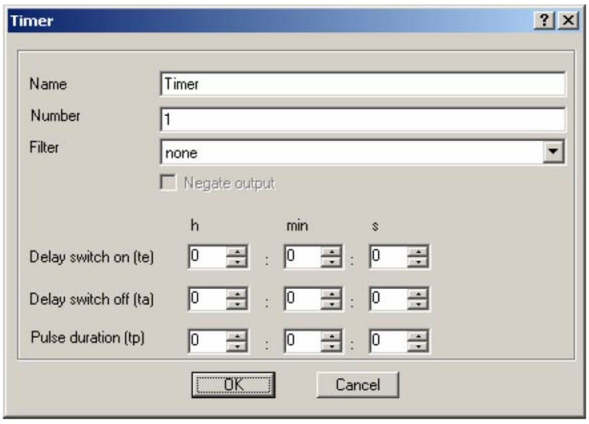
With bus voltage recovery, after a bus voltage failure the timer immediately assumes the output state, unless a time has been sent in the menu point *Initialisation*, in the menu bar under *Edit*.
In this case the element will wait for the set time until the output state of the timer is restored.

If for example an ON delay is set at the timer and telegrams with the value "1" are sent to the assigned input with a shorter interval than the delay time, then the beginning of the delay is restarted with each new telegram i.e. the timer operation can never elapse.

If a telegram which has been filtered out due to the settings arrives at the input of a timer, and if the assigned output has been set so that it sends a telegram each time a new object value is calculated, then the output does not send a telegram in this case. An "effective" filter can thus also be compared with a disabled gate or a timer operation that has not elapsed.

The parameter window *Timer* is opened through the following procedure:

- By a double click on the element
- By opening of the context window with the right mouse button and the *Edit parameter...* command.



Timer

Name: Timer
Number: 1
Filter: none
□ Negate output

	h	min	s
Delay switch on (te)	0	0	0
Delay switch off (ta)	0	0	0
Pulse duration (tp)	0	0	0

Fig. 74: Parameter window "Timer"

Name

Options: <u>Timer</u>

The name can be chosen as required. A maximum of 36 characters can be used.

Number

Options: <u>1</u>…30

The number is successive. It can be freely assigned if it is not being used by another element.

Filter

Options: none/
passthrough object value 0/
passthrough object value 1

On the graphic user interface the option none is represented by a dash, the option *passthrough object value 0* and the option *passthrough object value 1.*

Option *none* (-) = no filter function active.

Option *passthrough object value 0* = communication objects with value 0 are filtered out.

Option *passthrough object value 1* = communication objects with value 1 are filtered out.

Note The filter can only be activated if the input is assigned to a 1 bit object. The filter function affects the input.

Where is the filter function located in the process chain?

The filter function is situated before the timer, i.e. the input signals initially go through a filter and then through a time function.

Negated output

Options: activated = ticked
not activated = not ticked

The output can be negated by activating the checkbox. In the graphic user interface a negated output is signified by a circle.

Note: An output must be linked in order to negate it. The negation of a linked output can be set or deleted directly in the graphical user environment by a double click with the left mouse button on the respective output.

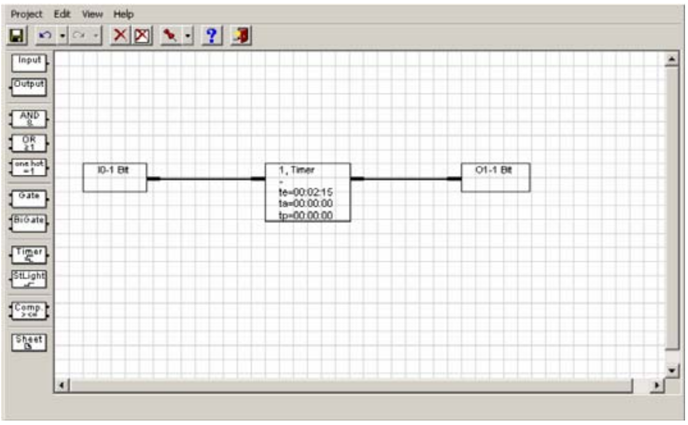
Delay switch on (t_e)

Options: Hours 0…18/
Minutes 0…59/
Seconds 0…59

The maximum adjustable on delay is 18:12:15.

On the graphic user interface the *Delay switch on* is identified by a t_e.

How does the Delay switch on function (te)?



I0-1 Bit

1, Timer
te=00:02:15
ta=00:00:00
tp=00:00:00

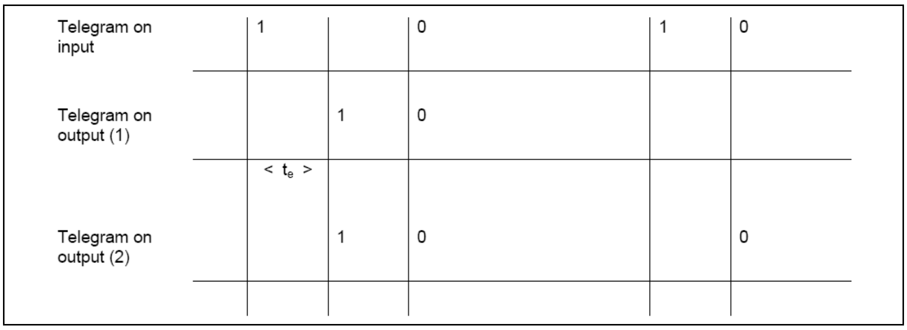
O1-1 Bit

Fig. 75: Delay switch on

A switch on signal present on the input of a timer is only transferred to the output after the set time te has timed out. A switch off signal is sent without delay by the output.

Note: If the output only sends *if the object value changes* or *with every assignment of a new object value*, depends on the parameterisation of the send criteria on the output.

An example:



Telegram on input	1		0	1	0
Telegram on output (1)		1	0		
	< t_e >				
Telegram on output (2)		1	0		0

Fig. 76: Diagram "Delay switch on"

(1) With the setting *output sends if object value changes*

(2) With setting *output sends with every assignment of a new object value*

Delay switch off (ta)

Options: Hours 0…18/
Minutes 0…59/
Seconds 0…59

The maximum adjustable off delay is 18:12:15.

On the graphic user interface the *Delay* switch off is identified by a ta.

How does the Delay switch off function (t_a)?

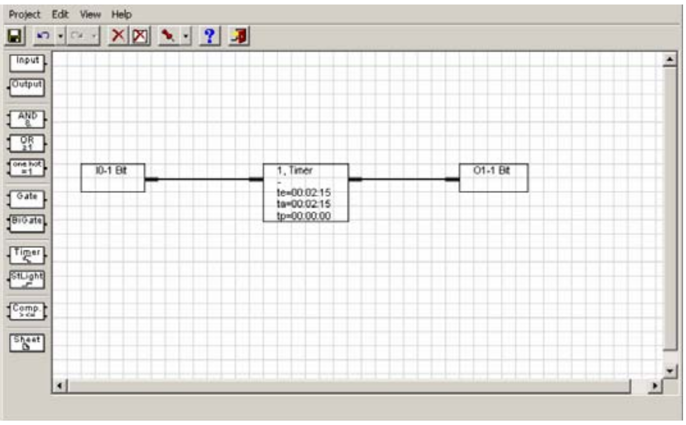
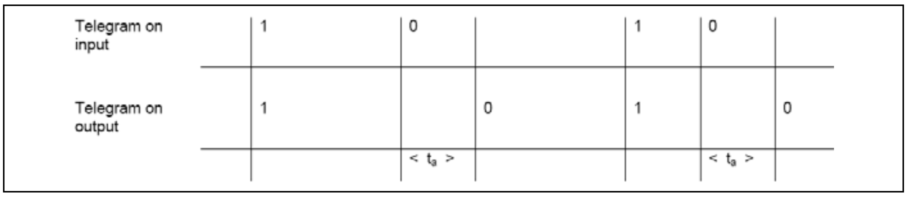


Fig. 77: Delay switch off

A switch off signal present on the input of a timer is only transferred to the output after the set time t_a has timed out. A switch on signal is sent without delay by the output.

Note: If the output only sends *if the object value changes* or *with every assignment of a new object value*, depends on the parameterisation of the send criteria on the output.

An example:



Telegram on input	1	0		1	0	
Telegram on output	1		0	1		0
		< t_a >			< t_a >	

Fig. 78: Diagram "Delay switch off"

How does the Delay switch on and off (t_e and t_a) function?

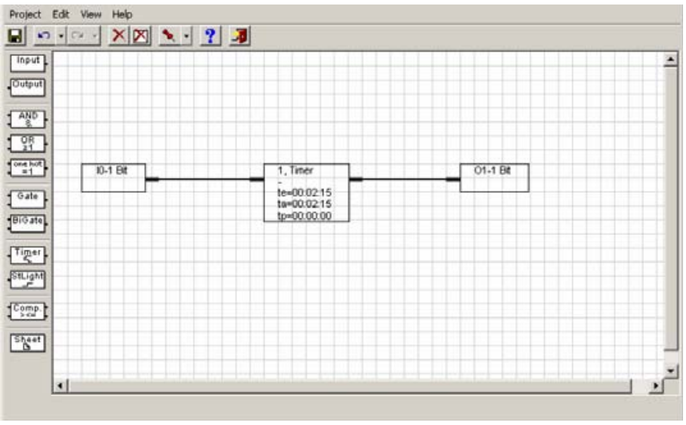


Fig. 79: Delay switch on and off

A switch on signal present on the input of a timer is only transferred to the output after the set time t_e has timed out. A switch off signal present on the input of a timer is only transferred to the output after the set time t_a has timed out.

Note: If the output only sends *if the object value changes* or *with every assignment of a new object value*, depends on the parameterisation of the send criteria on the output.

An example.

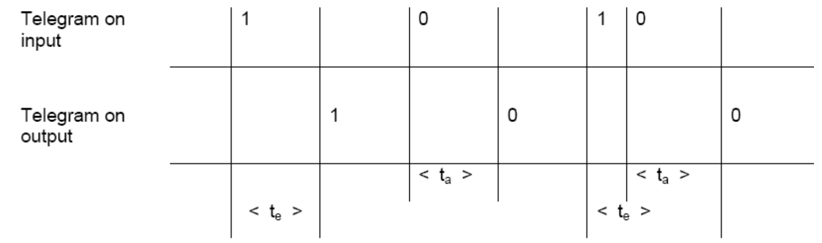


Fig. 80: Diagram “Delay switch on and off”

Pulse duration (t_p)

Options: Hours 0…18/
Minutes 0…59/
Seconds 0…59

The maximum adjustable pulse duration is 18:12:15.

On the graphic user interface the *pulse duration* is designated by a t_p.

How does the pulse duration function (t_p)?

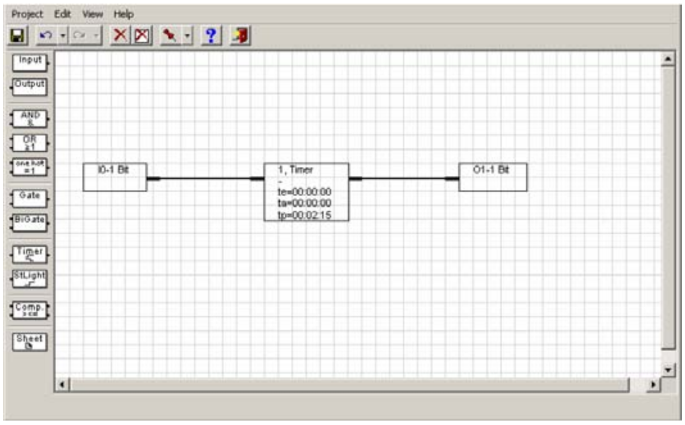


Fig. 81: Pulse duration

With the *pulse duration* function a switch on signal is reset (from 1 to 0) after a defined *pulse duration* time.

Note: If the output only sends *if the object value changes* or *with every assignment of a new object value*, depends on the parameterisation of the send criteria on the output.

An example:

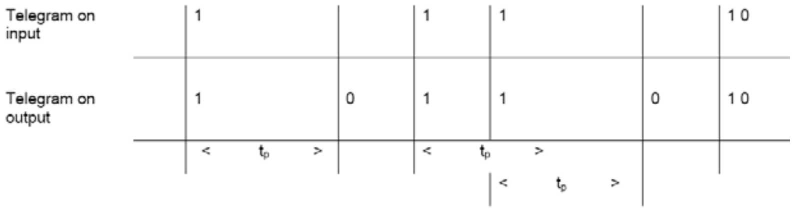


Fig. 82: Diagram "Pulse duration"

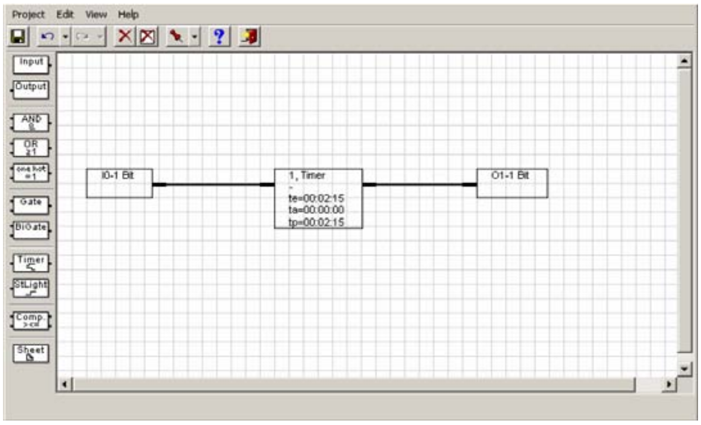
OK button

The settings are copied and the parameter window closes via the button.

Cancel button

The function is interrupted and the parameter window closes via the button.

How does the switch on delay and pulse duration function (t_e and t_p)?



I0-1 Bit

1, Timer
-
te=00:02:15
ta=00:00:00
tp=00:02:15

O1-1 Bit

Fig. 83: Switch on delay and pulse duration

A switch on signal present on the input of a timer is only transferred to the output after the set time t_e has timed out. After the *pulse duration* has timed out the switch on signal is reset (from 1 to 0).

If a new switch on signal is received within the *pulse duration time*, the set *pulse duration* remains unchanged and is not extended.

Note: If the output only sends *if the object value changes* or *with every assignment of a new object value*, depends on the parameterisation of the send criteria on the output.

An example:

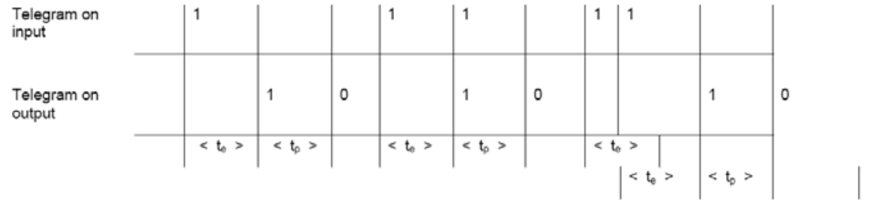


Fig. 84: Diagram "Switch on delay and pulse duration"

OK button

The settings are copied and the parameter window closes via the button.

Cancel button

The function is interrupted and the parameter window closes via the button.

3.6.8.Staircaselight

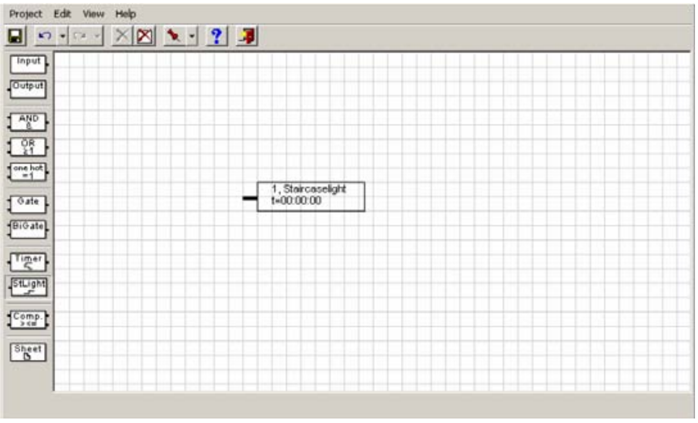


Fig. 85: Staircaselight

The *Staircaselight* function *consists of an input which is simultaneously an output.*

The input and output of the Staircaselight can be found on the left of the element.

Note: The Staircaselight is linked to an input object, only 1 bit objects are permissible.

In the event of a bus voltage failure, the stairlight function immediately reverts to its initial state i.e. the outputs go to zero signal regardless of the signals applied at the input or the time that has already elapsed.

The behaviour described beforehand is dependent on the linked input and its parameters.

How does the Staircaselight function operate?

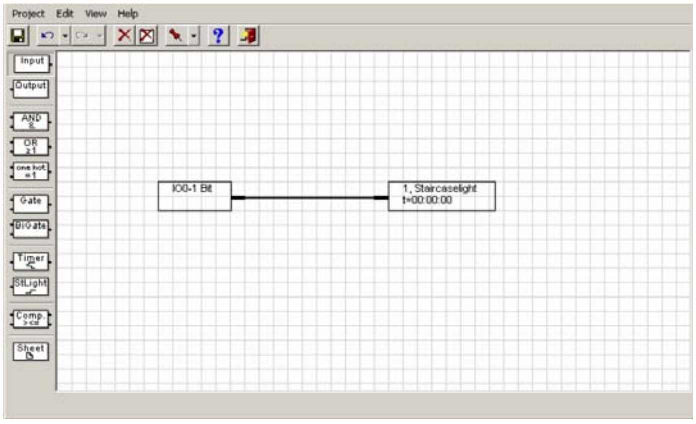


Fig. 86: Staircase lighting function

With the *Staircaselight* function the input signal (which is simultaneously the output signal) is reset after the autoreset time t is reset (e.g. from 1 to 0).If there is a new input signal within the autoreset time, the output signal remains at the existing value and the autoreset time is restarted.

An example:

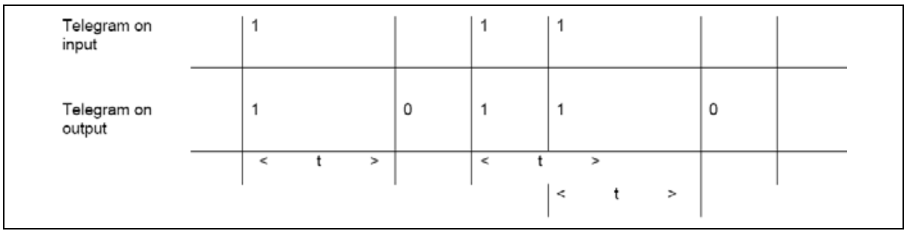
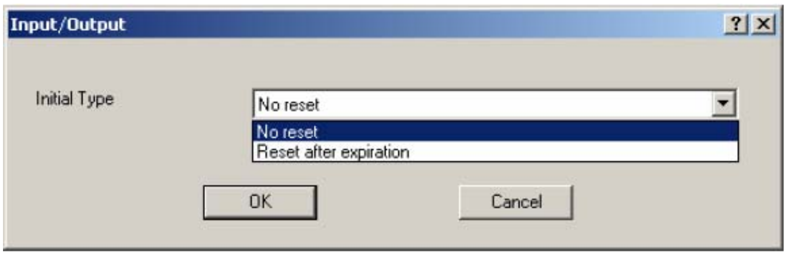


Fig. 87: Diagram "Staircaselight function"

The parameter window *Input/Output* becomes active through the following procedure:

- By a double click on the element
- By opening of the context window with the right mouse button and the *Edit parameter...* command.



Input/Output

Initial Type: No reset

Fig. 88: Input/Output function

With a bus voltage recovery the switched "on" output is

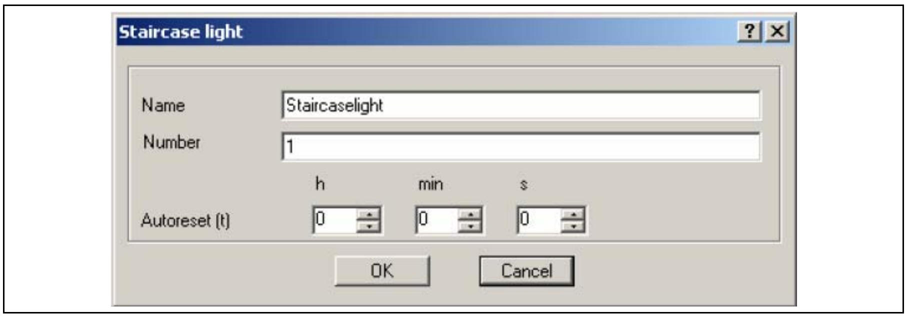
Options: *No reset*
Reset after expiration

Option *No reset* = with a bus voltage recovery the staircase lighting function is not reset.

Option *Reset after expiration* = with a bus voltage recovery the Staircaselight function is reset after the set time has timed out.

The parameter window *Staircaselight* becomes active through the following procedure:

- By a double click on the element
- By opening of the context window with the right mouse button and the *Edit parameter...* command.



Staircase light

Name: Staircaselight

Number: 1

Autoreset (t): h 0, min 0, s 0

Fig. 89: Parameter window "Staircaselight"

Name

Options: Staircaselight

The name can be chosen as required. A maximum of 36 characters can be used.

Number

Options: 1…30

The number is successive. It can be freely assigned if it is not being used by another element.

Autoreset (t)

Options: Hours 0…18/
Minutes 0…59/
Seconds 0…59

The maximum adjustable autoreset duration is 18:12:15.

On the graphic user interface the *autoreset time* is designated by a tp.

OK button

The settings are copied and the parameter window closes via the button.

Cancel button

The function is interrupted and the parameter window closes via the button.

3.6.9.Comparator

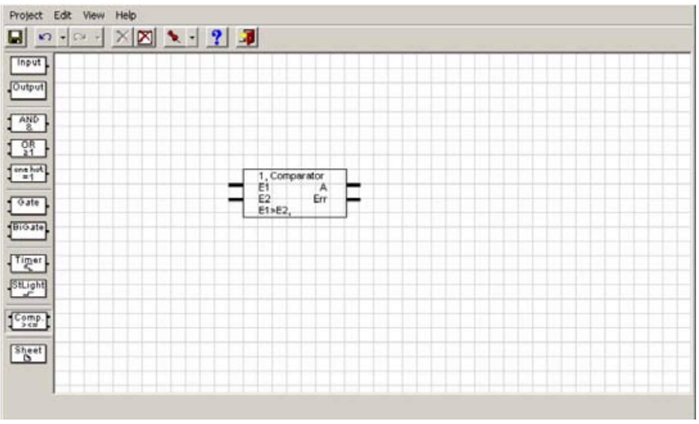


Fig. 90: Comparator

A *Comparator* consists of two inputs, an output and an overflow output (Err output). It compares two input values with one another after an adjustable comparator operator. If the comparative condition is fulfilled the output sends a 1, if it is not fulfilled it sends a 0.

Usually input 1 is used for measured values and input 2 for reference values. If the value range is exceeded during a comparison, a fault is indicated on the overflow output (Err output).

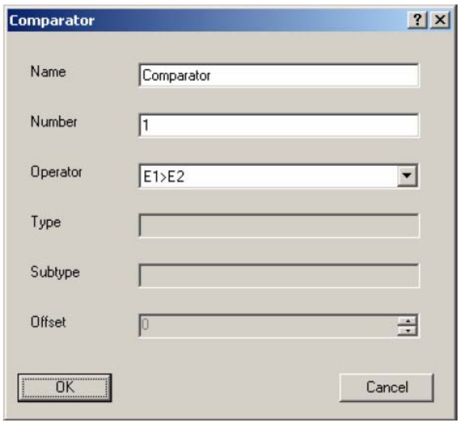
The inputs of the comparator are identified with E1 and E2 and are located on the left of the element.

The outputs of the comparator are identified with A and Err and are located on the right of the element.

Note: The two input values must be from the same data type, e.g. 1 byte, 2 byte or 4 byte.

The parameter window *Comparator* becomes active through the following procedure:

- By a double click on the element
- By opening of the context window with the right mouse button and the *Edit parameter..*. command.



Comparator

Name: Comparator

Number: 1

Operator: E1>E2

Type:

Subtype:

Offset: 0

Fig. 91: Parameter window "Comparator"

Name

Options: Comparator

The name can be chosen as required. A maximum of 36 characters can be used.

Number

Options: 1…10

The number is successive. It can be freely assigned if it is not being used by another element.

Operator

Options: E1 = E2/
E1 > E2/
E1 >= E2/
E1 <= E2/
E1 < E2

E1 = E2	=> input 1 (E1) is equal to input 2 (E2)
E1 > E2	=> input 1 (E1) is greater than input 2 (E2)
E1 >= E2	=> input 1 (E1) is greater than or equal to input 2 (E2)
E1 <= E2	=> input 1 (E1) is less than or equal to input 2 (E2)
E1 < E2	=> input 1 (E1) is less than input 2 (E2)

Note: The selected operator is shown on the graphic user interface at the very bottom.

Both parameters *Type* and *Subtype* remain inactive. *Type* and *subtype* in the *comparator* are oriented according to the input object types.

Type

Options: 1 Byte
2 Byte
4 Byte

The type of input object is displayed in the parameter.

Note: The selected type is shown on the graphic user interface at the very bottom.

Subtype

Options: Floating point
Signed
Unsigned

The subtype of input object is displayed in the parameter.

The following data types are possible:

1 Byte Unsigned	= 0…255>
1 Byte Signed	= -128…127…
2 Byte Floating point	= see table below
2 Byte Unsigned	= 0…65.535
2 Byte Signed	= -32.768…32.767…
4 Byte Unsigned	= 0…4.294.967.295
4 Byte Signed	= -2.147.483.648…2.147.483.647…

Offset

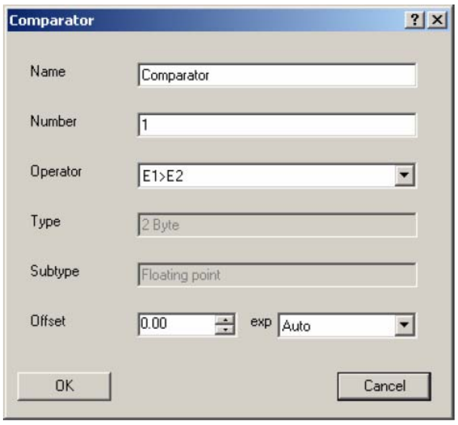
Options:	0…255	with 1 Byte Unsigned
	-128…0…127	with 1 Byte Signed,
	0 (see table)	with 2 Byte Floating point
	0…65.535	with 2 Bytes Unsigned
	-32.768…0…32.767	with 2 Bytes Signed
	0…4.294.967.295	with 4 Bytes Unsigned
	-2.147.483.648…0… 2.147.483.647	with 4 Bytes Signed

The function *Offset* functions as soon as an input of the comparator is linked.

For certain applications, e.g. room temperature control, it is necessary to apply an *Offset* according to the following formula:

Value on E2 + offset = input value

If an input is linked to a 2 byte floating point value the additional parameter power (exp.) appears.



Comparator

Name: Comparator

Number: 1

Operator: E1>E2

Type: 2 Byte

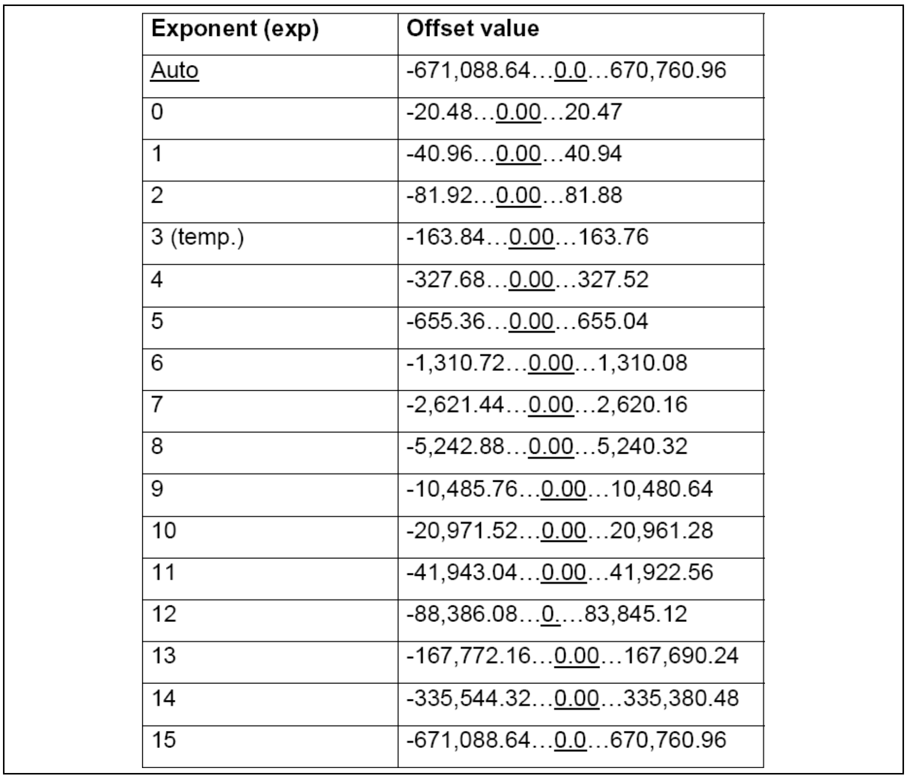
Subtype: Floating point

Offset: 0.00 exp: Auto

Fig. 92: Parameter window “Comparator - Exponent”

exp

The adjustable *Offset* depends on the exponent (exp).



Exponent (exp)	Offset value
Auto	-671,088.64…0.0…670,760.96
0	-20.48…0.00…20.47
1	-40.96…0.00…40.94
2	-81.92…0.00…81.88
3 (temp.)	-163.84…0.00…163.76
4	-327.68…0.00…327.52
5	-655.36…0.00…655.04
6	-1,310.72…0.00…1,310.08
7	-2,621.44…0.00…2,620.16
8	-5,242.88…0.00…5,240.32
9	-10,485.76…0.00…10,480.64
10	-20,971.52…0.00…20,961.28
11	-41,943.04…0.00…41,922.56
12	-88,386.08…0.…83,845.12
13	-167,772.16…0.00…167,690.24
14	-335,544.32…0.00…335,380.48
15	-671,088.64…0.0…670,760.96

Table 7: Exponent

OK button

The settings are copied and the parameter window closes via the button.

Cancel button

The function is interrupted and the parameter window closes via the button.

How does the Err output function?

Err output

If during the calculation of the input value by the *Offset* a value range is exceeded, a fault message in the form of a 1 is sent by the overflow output (Err output).

For certain application cases, e.g. room temperature control, it is necessary to apply an *Offset* according to the following formula:

Value on E2 + offset = input value

Note: The offset is added exclusively to the input value E2.

If the calculated input value exceeds the set value range, the *Overflow output* (Err output) sends an error message.

Example: Max. value range, e.g. 2 byte unsigned (0…65.535) input value on E2 = 65,535 plus an offset of 1 results in the value range being exceeded, i.e. the Err output sends an ON telegram.

Note: Both inputs of the comparator must be linked, otherwise it will not function.

3.6.10. Worksheet

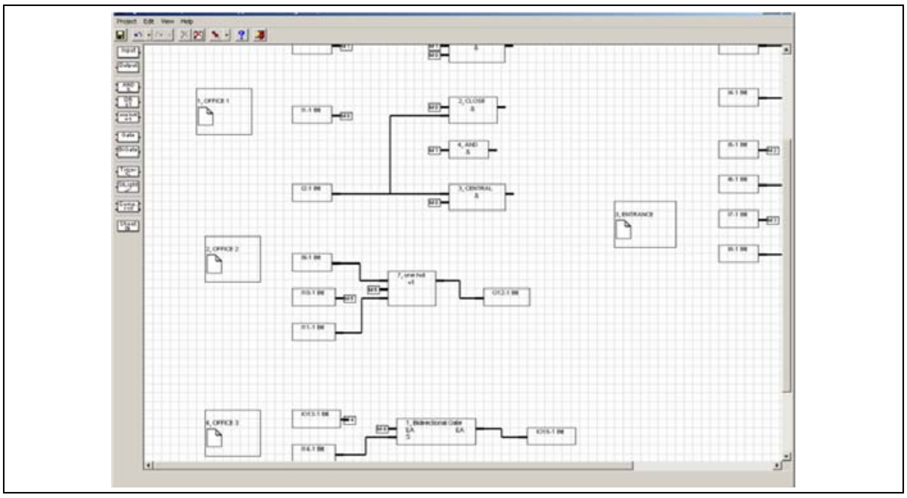
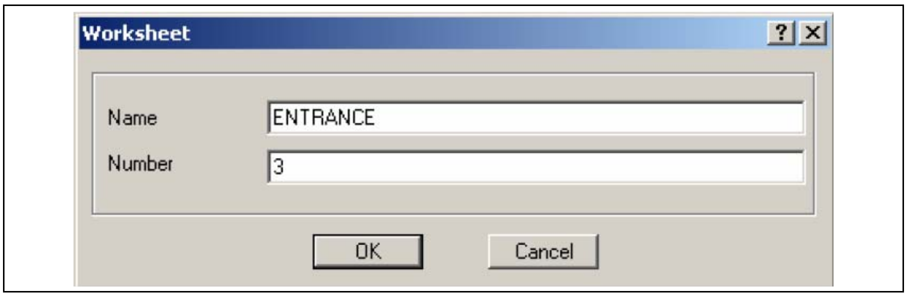


Fig. 93: Display of worksheets

More complex parameterisation and a smaller the monitor size can quickly lead to a very unclear view of the graphic parameterisation. Worksheet management is used in order to effectively design this. It describes a part of the graphical parameterisation and is implemented in the form of jump labels.

Note: Worksheets can be applied, edited and deleted.



Name: ENTRANCE

Number: 3

Fig. 94: Worksheet

Name

Options: ENTRANCE

The name can be chosen as required. A maximum of 36 characters can be used.

Note: The text field in the worksheet describes a partial area of the graphic parameterisation and is implemented as a marker.

Number

Options: 1…2…3…200

The number is successive. It can be freely assigned if it is not being used by another element.

OK button

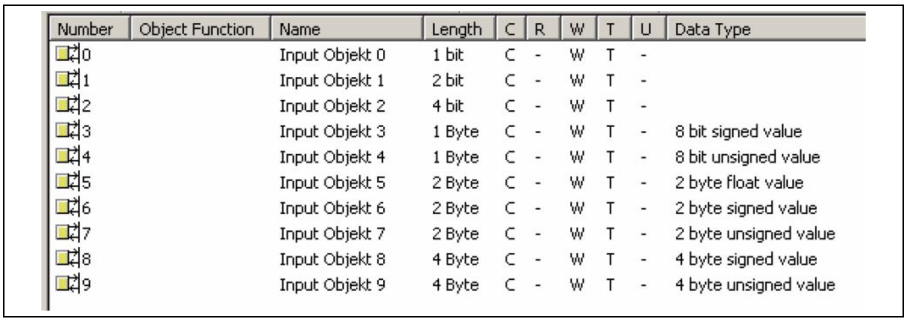
The settings are copied and the parameter window closes via the button.

Cancel button

The function is interrupted and the parameter window closes via the button.

3.7. Communication object

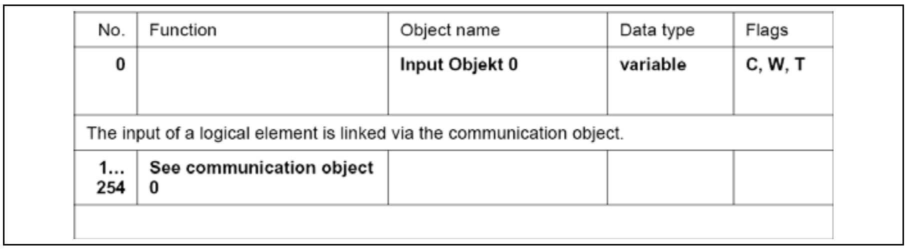
3.7.1.Input



Number	Object Function	Name	Length	C	R	W	T	U	Data Type
0		Input Objekt 0	1 bit	C	-	W	T	-	
1		Input Objekt 1	2 bit	C	-	W	T	-	
2		Input Objekt 2	4 bit	C	-	W	T	-	
3		Input Objekt 3	1 Byte	C	-	W	T	-	8 bit signed value
4		Input Objekt 4	1 Byte	C	-	W	T	-	8 bit unsigned value
5		Input Objekt 5	2 Byte	C	-	W	T	-	2 byte float value
6		Input Objekt 6	2 Byte	C	-	W	T	-	2 byte signed value
7		Input Objekt 7	2 Byte	C	-	W	T	-	2 byte unsigned value
8		Input Objekt 8	4 Byte	C	-	W	T	-	4 byte signed value
9		Input Objekt 9	4 Byte	C	-	W	T	-	4 byte unsigned value

Fig. 95: Communication objects for inputs

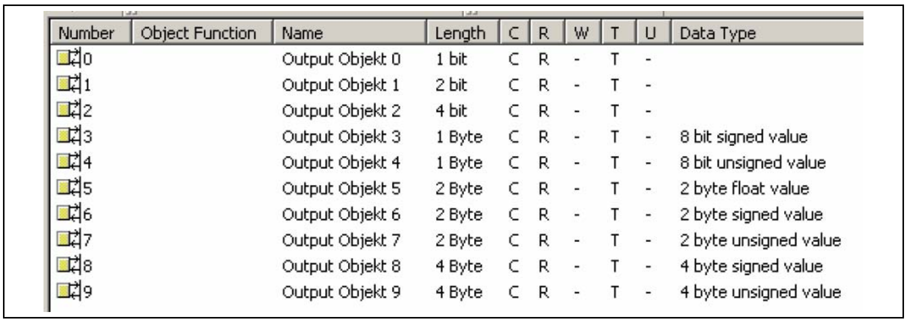
Note: In the graphics all possible data types for an input are represented.



No.	Function	Object name	Data type	Flags
0		**Input Objekt 0**	**variable**	**C, W, T**
The input of a logical element is linked via the communication object.				
1… 254	**See communication object 0**			

Table 8: Communication object for the input

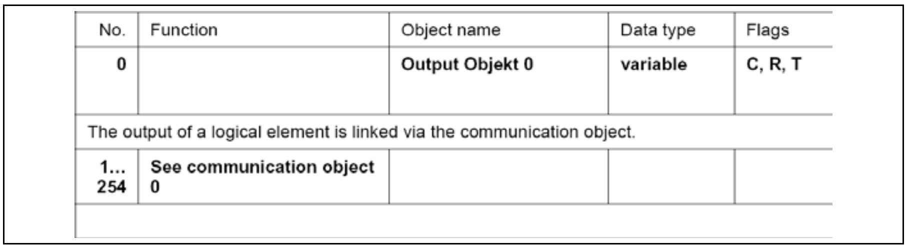
3.7.2.Output



Number	Object Function	Name	Length	C	R	W	T	U	Data Type
0		Output Objekt 0	1 bit	C	R	-	T	-	
1		Output Objekt 1	2 bit	C	R	-	T	-	
2		Output Objekt 2	4 bit	C	R	-	T	-	
3		Output Objekt 3	1 Byte	C	R	-	T	-	8 bit signed value
4		Output Objekt 4	1 Byte	C	R	-	T	-	8 bit unsigned value
5		Output Objekt 5	2 Byte	C	R	-	T	-	2 byte float value
6		Output Objekt 6	2 Byte	C	R	-	T	-	2 byte signed value
7		Output Objekt 7	2 Byte	C	R	-	T	-	2 byte unsigned value
8		Output Objekt 8	4 Byte	C	R	-	T	-	4 byte signed value
9		Output Objekt 9	4 Byte	C	R	-	T	-	4 byte unsigned value

Fig. 96: Communication objects for outputs

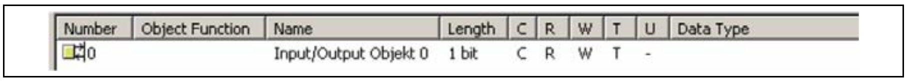
Note: In the graphics all possible data types for an output are represented.



No.	Function	Object name	Data type	Flags
0		**Output Objekt 0**	**variable**	**C, R, T**
The output of a logical element is linked via the communication object.				
1… 254	**See communication object 0**			

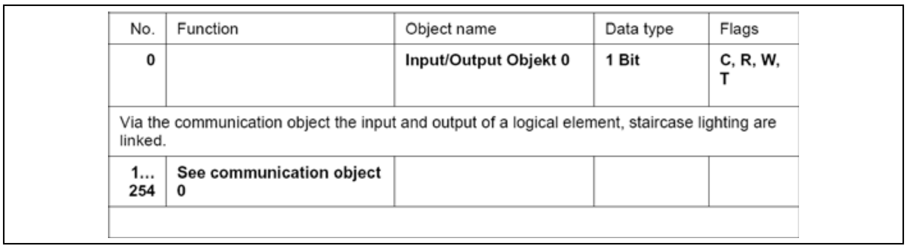
Table 9: Communication object for outputs

3.7.3.Input and output



Number	Object Function	Name	Length	C	R	W	T	U	Data Type
0		Input/Output Objekt 0	1 bit	C	R	W	T	-	

Fig. 97: Communication objects for input and output



No.	Function	Object name	Data type	Flags
0		**Input/Output Objekt 0**	**1 Bit**	**C, R, W, T**
Via the communication object the input and output of a logical element, staircase lighting are linked.				
1… 254	**See communication object 0**			

Table 10: Communication object for input and output

4. Planning and application

In this section you will find some tips for practical use of the Application Unit Logic time.

4.1. Help

Help provides the necessary information for parameterisation of the application program Logic Time 254 EA/2.

Help is accessed as follows:

- Press the **Help** button in the dialog window,
- In the Worksheet press the right mouse button to select **Help**,
- Press **F1** on the keyboard or
- Press the **Help** button in the toolbar.

You can best work with the application program and its help as follows:

First of all start the ETS3 and open the project which you require. Minimise the ETS3 main window and push it to the top right.

After the start of the application program, minimise the window and move it to the right edge of the screen so that about a third of the screen width remains free. Now call up Help and minimise and position the window that opens in the free area on the left of the monitor.

Now you can see the corresponding help page to every processing step on the left hand side if you access Help to obtain information. You can obtain further explanation about these terms by clicking on the green marked and underlined terms. You can return to the original Help page with Back.

4.2. Transient intermediate states (Glitches)

From hard-wired programmed controls it is well known that internal propagation times of the gates can cause the occurrence of transients, i.e. temporary transient states. These so-called *glitches* do not become apparent as they only last a few milliseconds and normally all memory devices and outputs have a delay of a few milliseconds.

Transient states can also occur on the ABL/S 2.1. This becomes noticeable for example when several telegrams are sent successively from a stable output state on the bus until a stable output state is re-established.

These glitches only become noticeable as the bus device that should react to these telegrams also reacts.

4.2.1.Criteria for the occurrence of glitches

Example of a functional sequence where a transient intermediate state arises:

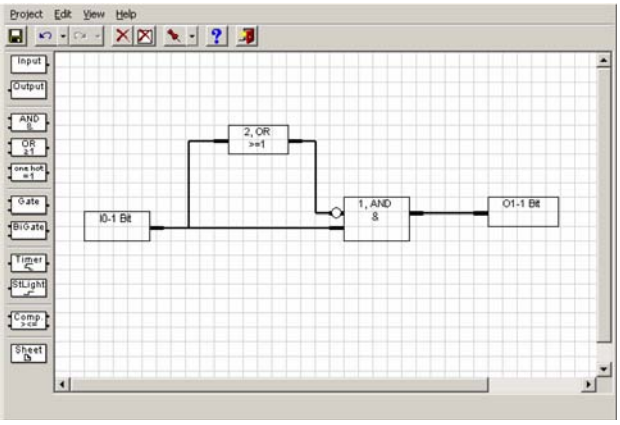
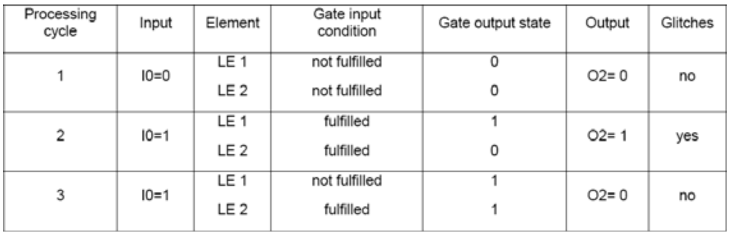


Fig. 98: Glitches



Processing cycle	Input	Element	Gate input condition	Gate output state	Output	Glitches
1	I0=0	LE 1 LE 2	not fulfilled not fulfilled	0 0	O2= 0	no
2	I0=1	LE 1 LE 2	fulfilled fulfilled	1 0	O2= 1	yes
3	I0=1	LE 1 LE 2	not fulfilled fulfilled	1 1	O2= 0	no

Table 11: Example, Glitches

All logical connections should be analysed to see if glitches could occur due to different operating (propagation) times of the logical elements.

The following should be noted:

If a signal change occurs at the input of a functional sequence, the current input state is then determined for all the elements of the functional sequence and then set accordingly for the output.

This process is undertaken in successive cycles.

The processing of elements in a cycle is carried out in the order of their numbering. The inputs and outputs are recalculated first of all, followed by the logical elements, gates and timers.

At the point of processing gate 1 the output value of gate 2 is still 0, thus the input condition for gate 1 is still fulfilled and send with a 1 telegram. During the next cycle, the true value is determined and a “0” telegram is sent.

4.2.2.Criteria for the occurrence of glitches

Example of a functional sequence where no transient intermediate state arises:

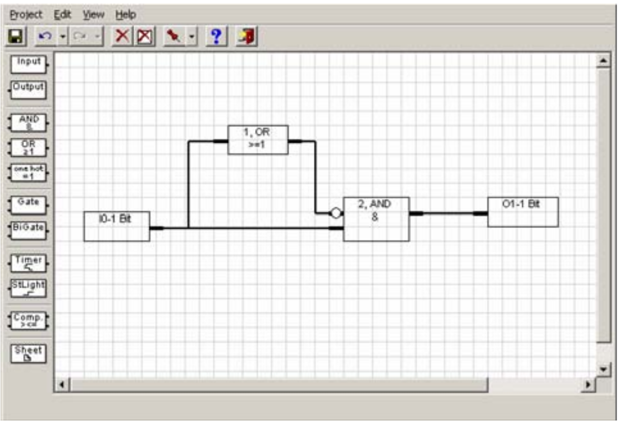
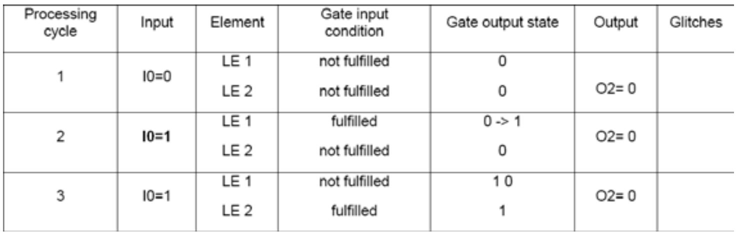


Fig. 99: No glitches

Example of a functional sequence where a transient intermediate state cannot arise:



Processing cycle	Input	Element	Gate input condition	Gate output state	Output	Glitches
1	I0=0	LE 1 LE 2	not fulfilled not fulfilled	0 0	O2= 0	
2	**I0=1**	LE 1 LE 2	fulfilled not fulfilled	0 -> 1 0	O2= 0	
3	I0=1	LE 1 LE 2	not fulfilled fulfilled	1 0 1	O2= 0	

Table 12: Example, no glitches

Note: The output of logical element 2 (also “1”) is first of all determined in processing cycle 2. The true value is determined in the same cycle, output O2 remains “0”.

4.3. Reaction time

The application program *Logic Time 254 EA/2* of the application unit ABL/S 2.1 processes the logical functions sequentially, similar to a PLC.

The state of the input and outputs is then determined in one processing cycle. Finally the elements are processed internally according to the parameters that have been assigned.

The processing of elements in a cycle is carried out in the order of their numbering. The logical elements are calculated first of all, followed by the gates and then the timers.

Finally the newly calculated states are sent.

The timer for a processing cycle – for processing the elements in ascending numerical order – is typically 100 ms.

4.4. Reaction on bus voltage failure

With a bus voltage failure the Application Unit Logic saves the communication object values of the inputs and outputs for at least one hour. If the bus voltage failure lasts considerably longer than an hour, the stored communication object values are lost.

Safe states after a bus voltage failure can be achieved by the selection of the behaviour after bus recovery parameter setting.

4.5. Behaviour after voltage recovery

In order to avoid unwanted operational performance after a bus voltage failure, particularly in larger installations, the installation of an uninterruptible power supply (UPS) should be provided for the bus voltage and for the interrogation voltage of the contacts.

4.5.1.Cold start

Start behaviour:

- The inactive waiting period times out.
- All RAM data is deleted.
- The communication objects are initialised in accordance with the parameters *After bus voltage recovery* …*with data loss* and *preset value.*
- If inputs are to be queried, they are queried every second.
- All logic functions are processed.

4.5.2.Warm start and bus reset

Start behaviour:

- The inactive waiting period times out.
- Gate states and all timer states are deleted.
- The communication objects are initialised in accordance with the parameters *After bus voltage recovery …without data loss* and *preset value.*
- If inputs are to be queried, they are queried every second.
- All logic functions are processed.

5. Appendix

A.1 Scope of delivery

The Application Unit Logic is supplied with the following parts. Please check the items received using the following list.

- 1 pc. ABL/S 2.1, Application Unit Logic, MDRC
- 1 pc. installation and operating instructions
- 1 pc. bus connection terminal (red/black)

A.2 Directory of drawings

A.3 Directory of tables